Seeing the Wind from a Falling Leaf

Zhiyuan Gao[1]* **Jiageng Mao**[1]* **Hong-Xing Yu**[2] **Haozhe Lou**[1] **Emily Yue-Ting Jia**[1]
Jernej Barbic[1] **Jiajun Wu**[2] **Yue Wang**[1]
[1]University of Southern California [2]Stanford University
{gaozhiyu, jiagengm, haozhelo, eyjia, jnb, yue.w}@usc.edu
{koven, jiajunwu}@cs.stanford.edu

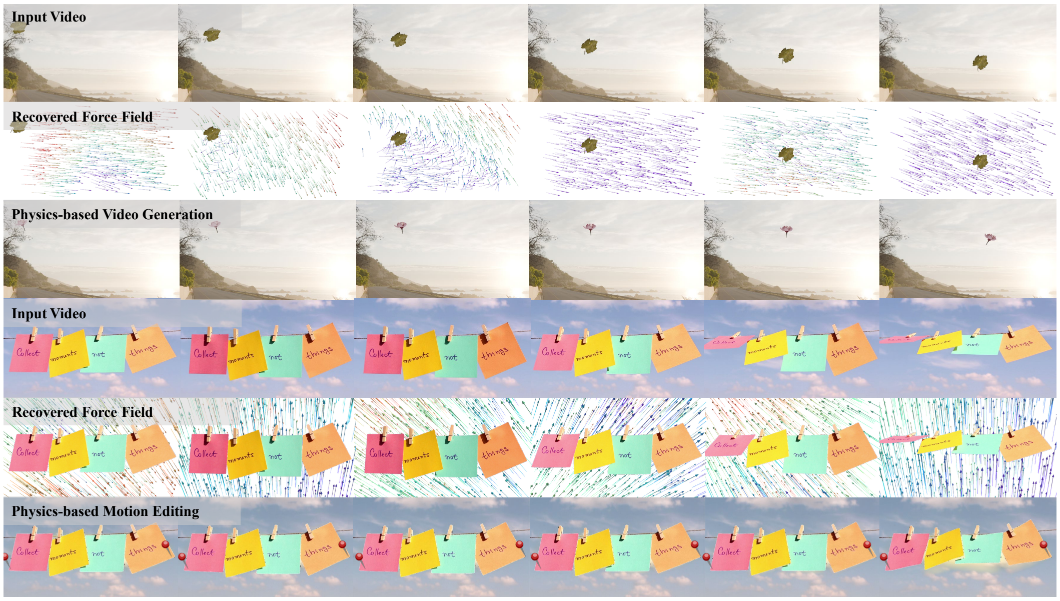

Figure 1: We propose an end-to-end differentiable framework capable of estimating invisible forces directly from video data, mimicking the human ability to perceive unseen physical effects through vision alone. This approach enables applications such as physics-based video generation, where new objects can be seamlessly introduced into a scene and simulated within the same force field. Force strength: from low to high (best viewed in colors).

Abstract

A longstanding goal in computer vision is to model motions from videos, while the representations behind motions, i.e. the invisible physical interactions that cause objects to deform and move, remain largely unexplored. In this paper, we study how to recover the invisible forces from visual observations, e.g., estimating the wind field by observing a leaf falling to the ground. Our key innovation is an end-to-end differentiable inverse graphics framework, which jointly models object geometry, physical properties, and interactions directly from videos. Through backpropagation, our approach enables the recovery of force representations from object motions. We validate our method on both synthetic and real-world scenarios, and the results demonstrate its ability to infer plausible force fields from videos. Furthermore, we show the potential applications of our approach, including physics-based video generation and editing. We hope our approach sheds light on understanding and modeling the physical process behind pixels, bridging the gap

*Equal contribution.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

between vision and physics. Please check more video results in our project page https://chaoren2357.github.io/seeingthewind/.

> *"Who has seen the wind? Neither I nor you: But when the leaves hang trembling, the wind is passing through."* – *Christina Rossetti*

1 Introduction

Watching leaves swirl and glide through the autumn breeze, we can almost sense the wind gently guiding them in a natural choreography. Similarly, as cherry blossom petals drift in spring, it feels as though the air cradles them, orchestrating their delicate descent. Although we cannot directly see the wind, humans can seamlessly infer these *invisible* physical interactions from visible cues in their surroundings, such as those captured in videos. While this intuitive physics capability has long existed in human vision, it remains underexplored in computer vision. In this paper, we bridge the gap by introducing a differentiable framework to revealing invisible forces from visual data.

The key challenge of this problem lies in extracting insights about an unseen target—dynamic forces—while relying exclusively on visible inputs. To address this, it is essential to understand how videos, as visible cues, connect to the underlying invisible dynamics. Consider a video of a leaf falling: external forces like wind, apply to a leaf with known shape, appearance, and physical properties, producing a motion that aligns with physical laws and is captured visually. By developing an end-to-end differentiable model of this physical process, we can learn and predict these invisible forces, such as wind, based on video evidence alone.

To this end, we propose a differentiable inverse graphics framework, which models objects' inherent properties (geometry, appearance, and physical properties), invisible force representations, and physical processes from video inputs. For object modeling, we leverage 3D Gaussians [1] as representations for shape and appearance, which can be easily obtained from videos. To model objects' physical properties, we propose a novel approach that leverages commonsense about physical properties in vision-language models and attaches the knowledge to 3D Gaussians. For force representations, we adopt the Eulerian perspective and introduce a novel causal tri-plane representation, which models the spatio-temporal continuity and intrinsic causality of forces with high fidelity. For physical processes, we implement a differentiable physics simulator for deformable objects to animate object motions based on object properties and forces. We note that our object representation (Gaussians as Lagrangian elements) and the force representation (causal tri-plane as grids) perfectly fit into the formulations of the material point method [2], allowing us to accurately model the physical process. Together, these components form a differentiable framework that bridges perception and physics, so that we can estimate forces from video object motions via backpropagation.

While our framework accurately models the physical process, recovering force representations from object motions in videos remains highly challenging. Unlike system identification approaches which estimate only a few physical parameters, forces are omni-directional and can present throughout the 3D space. Estimating such dense and complex force representations poses great challenges to optimization. Moreover, time integration in the physics simulator leads to unstable backpropagation, with gradients often exploding as they accumulate over time. To address the challenges, we propose a novel 4D sparse tracking objective, where we represent object motions as the movements of sparse keypoints in the spatio-temporal space, and the movements of the Lagrangian elements, i.e., the 3D Gaussians, are further controlled by their neighboring keypoints via barycentric interpolation. With this objective, we greatly reduce the complexity of the prediction space and facilitate the estimating of force representations.

We evaluated our estimated force representations on both synthetic and real-world scenarios. The results demonstrate our method's ability to recover invisible forces from videos. Moreover, we show that with the estimated force representation, we can generate novel and physically plausible object motions by changing object types, physical properties, or boundary conditions, which enables realistic physics-based video generation and editing.

To conclude, we summarize our contributions as follows:

- We identify an important problem in physics understanding from videos: recovering invisible forces from object motions. To tackle this problem, we propose a novel inverse-graphics framework that jointly models object properties, forces, and physical processes, enabling the estimation of underlying forces directly from video observations.

- We introduce a novel sparse tracking objective, which effectively handles the optimization challenges in differentiable physics and enables robust estimation of forces from visual inputs.
- We demonstrate our method's ability to recover forces from motion, and showcase its potential for generating physically plausible motions and enabling physics-based video generation and editing.

2 Related Works

Intuitive Physics. Understanding the physical world is a fundamental aspect of human intelligence. Researchers have long sought to bring this intuitive physics understanding ability to machine intelligence. Galileo [3] and the following works [4, 5] integrated deep learning with physics simulation to estimate physical object properties from visual observations. More recent approaches performed system identification by leveraging differentiable physics [6–9], neural fields [10–12], 3D Gaussian splatting [13, 14], vision-language models [15, 16], and video generation [17–20], enabling more accurate estimation of physical object properties. However, these methods primarily focus on a single physical parameter, such as mass, friction, and Young's modulus. In contrast, estimating forces is significantly more challenging, as they are vectors that can exist throughout the 3D space. In this paper, we propose a novel framework that successfully recovers force fields from visual inputs.

Differentiable Physics. Differentiable physics simulators [21–29] have been widely used to bridge perception and physics by enabling the backpropagation of particle motion gradients to physical parameters. However, using gradients from physics simulators to optimize physical properties can be notoriously difficult, as the inherent discontinuous behavior and the time integration of physics simulation often lead to vanishing or exploding gradients. To handle this challenge, we propose an optimization scheme with a novel sparse tracking objective, which greatly stabilizes the estimation process and enables robust recovery of high-dimensional forces.

Force Estimation. Researchers explored modeling contact forces for robotic manipulation [30–34] and human-object interactions [35–40]. However, most approaches rely on controlled robotic environments with tactile sensors or require strong priors on hand and object shapes, as well as physical properties, to estimate forces. In contrast, our method operates on natural videos with minimal assumptions about object properties, enabling force estimation in unconstrained scenarios.

Physics-based Generation. Researchers explored reconstructing physically interactive scenes [41, 42] and generating physically plausible videos [43–46]. Most approaches rely on physics simulators or physics-informed neural networks [47] to animate motion, but they typically require manually specified forces and environmental conditions. Beyond these, interactive editing methods [48, 49] drive visual changes by optimizing displacement fields in the image or feature space under generative priors; such formulations specify apparent motion without estimating underlying physical forces. An alternative approach learns 3D velocity fields directly from videos [50, 51], producing smooth trajectories yet lacking explicit force representations, which makes parameter-aware edits (e.g., changing mass) less principled. In contrast, our approach automatically recovers forces and physical conditions from natural videos and applies them to novel objects, enabling physics-driven video generation without manual parameter tuning.

3 Method

We study recovering invisible forces from videos. Our inverse graphics framework first models object properties (Section 3.1), force representations (Section 3.2), and physical processes (Section 3.3) from videos. To optimize force representations, we introduce a sparse tracking objective (Section 3.4). An overview of our method is in Figure 2.

3.1 Object Modeling

Capturing the essence of dynamic objects requires modeling both their shape for accurate physical interactions and their appearance for visual fidelity. This necessitates a representation that seamlessly integrates precise Lagrangian shape modeling with photorealistic rendering. To this end, we adopt 3D Gaussians [1] as the representation for shape and appearance. Specifically, an object in a video is

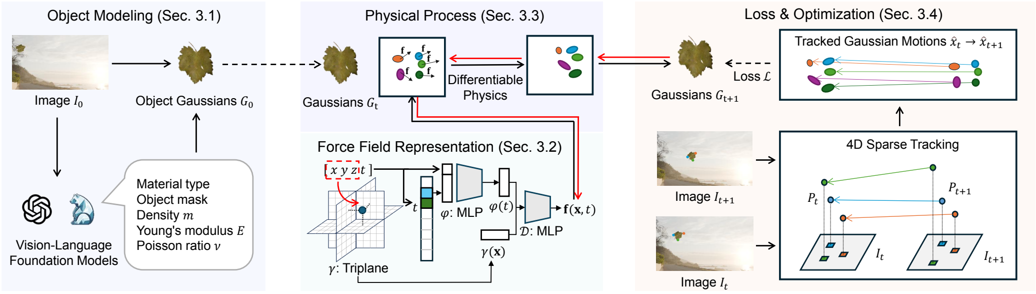

Figure 2: We propose a differentiable inverse graphics framework to recover invisible forces from videos by integrating object modeling, physics simulation, and optimization. Objects are represented with 3D Gaussians and assigned physical properties via Vision-Language Models. Forces are modeled as a causal tri-plane, and object motions are simulated using a differentiable physics simulator. A sparse tracking objective enables robust differentiable force recovery from videos.

represented by a set of Gaussian kernels G in the 3D space. Each Gaussian kernel G is parameterized by

$$G = \{\mathbf{x}, \mathbf{v}, \mathbf{\Sigma}, \sigma, SH, \mathbf{D}, m, E, \nu\}, \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^3$ and $\mathbf{v} \in \mathbb{R}^3$ are the spatial location and velocity of a Gaussian kernel respectively. The covariance matrix $\mathbf{\Sigma}$ represents the shape, and opacity σ and spherical harmonics SH represent the appearance of a Gaussian. Moreover, we attach each Gaussian with its physical properties: a deformation gradient $\mathbf{D}$, the mass m, Young's modulus E, and the Poisson ratio ν, which we will discuss later.

Since objects in a video undergo motion and deformation due to external forces, their corresponding 3D Gaussians also evolve over time. Let t denote a timestep in the video. The Gaussians at time t is then defined as

$$G^t = \{\mathbf{x}^t, \mathbf{v}^t, \mathbf{\Sigma}^t, \sigma, SH, \mathbf{D}^t, m, E, \nu\}, \tag{2}$$

where the spatial position $\mathbf{x}^t$, velocity $\mathbf{v}^t$, covariance $\mathbf{\Sigma}^t$, and deformation gradient $\mathbf{D}^t$ change over time t.

We initialize the Gaussians $\{G^0\}$ at $t = 0$ only using the first frame of the video. Specifically, we use pixel-aligned point clouds that are extracted from the first image I^0 via a pretrained metric-depth model [52] to initialize the Gaussian positions $\{\mathbf{x}^0\}$, and we optimize $\{\mathbf{\Sigma}^0, \sigma, SH\}$ via Gaussian splatting on I^0. Notably, although point clouds from a single image can be incomplete, we can still obtain robust force estimates thanks to the proposed sparse tracking objective, which will be discussed later. We have also explored multiview object reconstruction in our experiments. For $\mathbf{v}^0$ and $\mathbf{D}^0$, we initialized them as $\mathbf{0}$ and $\mathbf{I}$ respectively.

To model the physical interactions between objects and forces, we also need to know the objects' physical properties from videos. To this end, we introduce a simple but effective approach that leverages commonsense knowledge from vision-language models to assign the physical properties $\{m, E, \nu\}$ to each 3D Gaussian. Specifically, given the first image I^0, we first query a vision-language model [53] to infer the object types and provide an estimate of the physical properties $\{m, E, \nu\}$ from commonsense knowledge. Then, we query a grounded segmentation model [54] to generate object segmentation masks based on the object types. Finally, the pixel-aligned Gaussians $\{G^0\}$ that are inside the object masks are assigned with the corresponding estimated physical properties $\{m, E, \nu\}$. For common objects, the estimated physical properties from the vision-language model are quite robust. Hence, even without accurate system identification, our framework could provide a robust force estimate with the commonsense physical properties(see Experiments 4.3).

Leveraging foundation models [52–54], our framework automatically recovers objects' geometry, appearance, and physical properties from videos without manual effort. The recovered object Gaussians serve as a unified representation for modeling physical interactions in dynamic videos.

3.2 Force Representations

Properly modeling force representations is essential to our framework. For point contact forces, we can directly define force vectors on Gaussian particles. However, for forces that are distributed throughout 3D space, *e.g.*, wind, we adopt the Eulerian perspective and introduce a causal tri-plane to represent forces in 3D space. This representation is based on the observations that forces are spatially continuous and causally dependent over time. Specifically, we define the force $\mathbf{f}$ at the position $\mathbf{x}$ and the time t as

$$\mathbf{f}(\mathbf{x}, t) = \mathcal{D}(\gamma(\mathbf{x}) + \varphi(t; \varphi(t-1))), \tag{3}$$

where $\mathcal{D}(\cdot)$ is a feature decoder and $\gamma(\cdot)$ represents the tri-plane feature map from [55]. $\varphi(\cdot)$ is a small MLP that encodes the time t, initialized using the learned weights from the previous timestep $t-1$, *i.e.*, $\varphi(t-1)$. Compared to other 4D representations [56–58], this representation disentangles space and time, leading to superior computational efficiency. Additionally, the recursive dependency of $\varphi(t)$ on $\varphi(t-1)$ enables accurate modeling of evolving force dynamics over time.

3.3 Physical Process

With the object Gaussians and force representations, we are ready to simulate object motion following physical laws. To this end, we implemented a differentiable physics simulator for deformable objects using the Material Point Method (MPM) [2]. Empirically, we found that our object representation, where Gaussians act as Lagrangian elements, and our force representation, modeled as a causal tri-plane on a grid, naturally align with the formulations of [2], enabling accurate modeling of the physical process.

In detail, the forward physical process $F_{physics}$ takes the object Gaussians $\{G^t\}$ at time t and the force field $\mathbf{f}(\mathbf{x}, t)$, and outputs $\{G^{t+1}\}$ at the next timestep $t+1$:

$$\{G^{t+1}\} = F_{physics}(\{G^t\}, \mathbf{f}(\mathbf{x}, t)). \tag{4}$$

The physical process relies on multiple sub-steps δt to update motions from t to $t+1$ incrementally. In the following section, we introduce the computational flow in a sub-step δt. For simplicity, we consider a single Gaussian G^t and omit the particle-to-grid, grid computation, and grid-to-particle process in MPM, focusing solely on the core physics principles and update formulas.

For a Gaussian $G^t : \{\mathbf{x}^t, \mathbf{v}^t, \mathbf{\Sigma}^t, \sigma, SH, \mathbf{D}^t, m, E, \nu\}$ at the time t, we first characterize the object deformations by updating the deformation gradient $\mathbf{D}^t$:

$$\mathbf{D}^t = (\mathbf{I} + \nabla \mathbf{v}^{t-\delta t} \delta t)\mathbf{D}^{t-\delta t}, \tag{5}$$

where $\nabla \mathbf{v}^{t-\delta t}$ is the velocity gradient and $\mathbf{I}$ is the identity matrix. Then, we update the Gaussian velocity $\mathbf{v}^t$ by incorporating both external and internal forces:

$$\mathbf{v}^t = \mathbf{v}^{t-\delta t} + \delta t \frac{\mathbf{f}(\mathbf{x}^t, t)}{\mathbf{m}} + \delta t \frac{\mathbf{f}_{\mathrm{i}}(\mathbf{x}^t, t, E, \nu, \mathbf{D}^t)}{\mathbf{m}}, \tag{6}$$

where $\mathbf{f}(\mathbf{x}^t, t)$ is the external force by querying the casual tri-plane $\mathbf{f}(\mathbf{x}, t)$ at the Gaussian position $\mathbf{x}^t$, $\mathbf{f}_{\mathrm{i}}(\mathbf{x}^t, t, E, \nu, \mathbf{D}^t)$ represents internal forces $\mathbf{f}_{\mathrm{i}}$ determined by the constitutive model. $\mathbf{m}$ is the mass matrix derived from the mass m of the Gaussian. Note that external forces are computed on particles before the particle-to-grid step, while internal forces are calculated on the corresponding grid during the grid-to-particle step. External forces are applied directly to particles because the scene volume is much larger than the occupied regions. Acting on particles instead of grid nodes avoids empty cells and yields finer, less noisy fields aligned with the moving mass.

Next, we update the position $\mathbf{x}^t$ of the Gaussian following standard time integration:

$$\mathbf{x}^t = \mathbf{x}^{t-\delta t} + \delta t \mathbf{v}^t. \tag{7}$$

Finally, the covariance matrix $\mathbf{\Sigma}$ is calculated based on the deformation gradient:

$$\mathbf{\Sigma}^t = \mathbf{D}^t \mathbf{\Sigma}^0 (\mathbf{D}^t)^T. \tag{8}$$

Through multiple sub-step updates, we can evolve the Gaussian state from G^t to G^{t+1}. Notably, the computational process is fully differentiable. Hence, given the per-Gaussian motion $\hat{\mathbf{x}}^t \to \hat{\mathbf{x}}^{t+1}$

extracted from adjacent video frames, we leverage $\hat{\mathbf{x}}^t \rightarrow \hat{\mathbf{x}}^{t+1}$ as the motion tracking target to optimize the force field $\mathbf{f}(\mathbf{x}, t)$ via backpropagation.

$$\begin{aligned} & \min_{\mathbf{f}(\mathbf{x},t)} |\hat{\mathbf{x}}^{t+1} - \mathbf{x}^{t+1}|, \\ \text{s.t.} \quad & (\hat{\mathbf{x}}^t, \mathbf{x}^{t+1}) \in (G^t, G^{t+1}), \\ & G^{t+1} = F_{physics}(G^t, \mathbf{f}(\hat{\mathbf{x}}^t, t)). \end{aligned} \tag{9}$$

In the following sections, we will discuss how to establish per-Gaussian motion $\hat{\mathbf{x}}^t \rightarrow \hat{\mathbf{x}}^{t+1}$ from videos, and how to optimize the force field $\mathbf{f}(\mathbf{x}, t)$ robustly.

3.4 Recovering Forces from Videos

To optimize the force field $\mathbf{f}(\mathbf{x}, t)$, it is essential to track per-Gaussian motions $\hat{\mathbf{x}}^t \rightarrow \hat{\mathbf{x}}^{t+1}$ from videos as the optimization target. A straightforward approach is to use a photometric loss, *i.e.* comparing the pixel differences in adjacent frames $|I^t - I^{t+1}|$ by projecting the Gaussians onto the image plane: $I^t = \pi(\{G^t\})$, where π is the projection function. Nevertheless, we found that photometric loss alone fails to provide sufficient motion constraints, often resulting in vanishing gradients during optimization. Alternatively, we can extract dense 3D scene flows from videos using off-the-shelf depth and optical flow prediction or 4D reconstruction [59–61]. Nevertheless, the accuracy of these pre-trained models is limited, resulting in noisy dense 3D flows that significantly hinder the optimization process. We found the key to robust optimization is to reduce the target space and adopt more reliable motion estimates. To this end, for objects with bending-only deformations (e.g., paper folding) or small deformations, we introduce a novel 4D sparse-tracking objective. Specifically, we adopt a more reliable point-tracking algorithm [62] that provides sparse, pixel-level estimates of object keypoint motions $\mathbf{p}^t \rightarrow \mathbf{p}^{t+1}$, where $\mathbf{p} \in \mathbb{R}^{N\times 2}$ is N keypoint pixel coordinates. Next, we want to establish the keypoint correspondences in 3D: $\mathbf{P}^t \rightarrow \mathbf{P}^{t+1}$, where we use $\mathbf{P} \in \mathbb{R}^{N\times 3}$ and $\mathbf{p} \in \mathbb{R}^{N\times 2}$ to denote the associated keypoints in the 3D and pixel space respectively. To obtain $\mathbf{P}^t \rightarrow \mathbf{P}^{t+1}$ from $\mathbf{p}^t \rightarrow \mathbf{p}^{t+1}$, we first estimate the 3D keypoint locations $\mathbf{P}^0$ in the first frame by unprojecting $\mathbf{p}^0$ into 3D with depth estimates d:

$$\mathbf{P}^0 = \pi^{-1}(\mathbf{p}^0, d), \tag{10}$$

where d comes from a metric-depth model [52] and π^{-1} is an inverse-projection function. Then, for each adjacent frames, we obtain $\mathbf{P}^t \rightarrow \mathbf{P}^{t+1}$ from $\mathbf{P}^t$ and $\mathbf{p}^{t+1}$ by optimizing the following objective:

$$\min_{\mathbf{P}^t \rightarrow \mathbf{P}^{t+1}} |\pi(\mathbf{P}^{t+1}) - \mathbf{p}^{t+1}| + \lambda \mathcal{L}_{arap}, \tag{11}$$

where the as-rigid-as-possible loss $\mathcal{L}_{arap}$ is represented as

$$\mathcal{L}_{arap} = \Sigma_{i,j\in\mathbf{P}} |(\mathbf{P}_i^{t+1} - \mathbf{P}_j^{t+1}) - (\mathbf{P}_i^t - \mathbf{P}_j^t)|. \tag{12}$$

By minimizing the re-projection errors while keeping the object skeleton as rigid as possible, we obtain a robust estimate of the 3D keypoint motions $\mathbf{P}^t \rightarrow \mathbf{P}^{t+1}$. Notably, without reliance on per-frame depth estimation, our method circumvents the inconsistent video depth estimation problem and enables more robust 3D motion estimates.

Next, we leverage the sparse keypoint motions $\mathbf{P}^t \rightarrow \mathbf{P}^{t+1}$ to control the per-Gaussian motions $\hat{\mathbf{x}}^t \rightarrow \hat{\mathbf{x}}^{t+1}$ via

$$\hat{\mathbf{x}} = \alpha_i \mathbf{P}_i + \alpha_j \mathbf{P}_j + \alpha_k \mathbf{P}_k, \tag{13}$$

where the Equation 13 is the barycentric interpolation, $P_{i,j,k}$ are the 3-nearest neighbors of $\hat{\mathbf{x}}$, and the coefficients $\alpha_{i,j,k}$ are computed in the first frame and fixed in the following frames. The sparse keypoints $\mathbf{P}$ characterize the object skeletons and control the fine-grained Gaussian positions $\hat{\mathbf{x}}$. Compared to the 3D scene flow approach that directly tracks each Gaussian's motion, estimating sparse keypoint motions $\mathbf{P}^t \rightarrow \mathbf{P}^{t+1}$ reduces the prediction space and demonstrates superior robustness and accuracy, allowing us to obtain high-quality $\hat{\mathbf{x}}^t \rightarrow \hat{\mathbf{x}}^{t+1}$ for optimizing the force field.

Finally, we optimize the force $\mathbf{f}(\mathbf{x}, t)$ using the motion-tracking loss $\mathcal{L}_{motion}$ in Equation 9, with the estimated Gaussian motions $\hat{\mathbf{x}}^t \rightarrow \hat{\mathbf{x}}^{t+1}$. Moreover, we add two regularization terms $\mathcal{L}_{space}$ and $\mathcal{L}_{time}$ for spatial and temporal smoothness, respectively:

$$\mathcal{L} = \mathcal{L}_{motion} + \lambda_1 \mathcal{L}_{space} + \lambda_2 \mathcal{L}_{time}, \tag{14}$$

where $\mathcal{L}_{space}$ follows [56] and penalize the total variation in space, and $\mathcal{L}_{time}$ encourages temporal smoothness by penalizing the parameter differences of the time encoder $\varphi(\cdot)$:

$$\mathcal{L}_{time} = |\varphi_\theta^{t+1} - \varphi_\theta^t|, \tag{15}$$

where φ_θ^t is the parameters of the time encoder φ at the time t in Equation 3. With the losses in Equation 14, we are able to recover the force $\mathbf{f}(\mathbf{x}, t)$ by tracking the Gaussian motions $\hat{\mathbf{x}}^t \to \hat{\mathbf{x}}^{t+1}$ extracted from videos.

4 Experiments

In this section, we conduct comprehensive experiments to investigate the following key questions:

- Can our method successfully recover forces from both synthetic and real-world videos? (Section 4.2)
- How do the proposed components, *i.e.*, force representation and loss function affect the final performance, and how robust is the proposed VLM framework to variations in object physical properties?(Section 4.3)
- How can our method be applied to physics-based video generation and editing? (Section 4.4)

4.1 Experimental Setup

We conduct experiments on both real-world and synthetic data. For real-world data, we leverage Internet videos to verify the physical plausibility of recovered forces by visualizing the force field. In addition, we conduct real physical experiments with a force gauge to measure the actual forces, and we evaluate our recovered forces via re-simulation. For synthetic data, we use synthetic objects in [63, 18] to evaluate the numerical accuracy of recovered forces. We leverage objects from 3 distinct material types, *i.e.*, elastic, elastoplastic, viscoplastic, 6-8 unique force fields, and 2 different camera viewpoints to build the synthetic scenarios in the physics simulator [24]. We use rendered videos as inputs to our system to estimate the forces and compare them with the ground truth forces in the simulator. For numerical comparisons in synthetic scenarios, we adopt image reconstruction metrics, *i.e.*, PSNR, SSIM [64], and LPIPS [65], to compare the re-simulated videos with the recovered forces and the original input videos, to demonstrate the accuracy of recovered forces to match the ground truth object motions in simulation. Moreover, we compare the recovered forces with the ground truth forces using two metrics: average magnitude error (reported as percentages) and direction error (measured in degrees).

4.2 Force Recovery

In-the-Wild Videos. We evaluate our method on real Internet videos to demonstrate its ability to recover plausible force fields from natural object motions. Figure 4 presents qualitative results on various scenes. Our method successfully infers the underlying forces by observing object deformations and trajectories over time. The visualized forces dynamically adapt to object motion, demonstrating a physically plausible force field that varies over time.

Controlled Real-World Physics Experiments. Since the ground truth forces in real Internet videos are unknown, we conduct controlled real-world experiments to further validate our method. Using a force gauge, we apply known forces to an object while capturing its motion on video. We then use our method to recover the force field and reapply it to the object in simulation. As shown in Figure 3, our method successfully reconstructs the object's motion and deformation, closely aligning with real-world observations. The experimental results demonstrate the accuracy of our recovered forces in real-world scenarios.

Synthetic Scenarios. To evaluate the numerical accuracy of recovered forces, we build synthetic scenarios in the physics simulator to obtain the “ground truth" forces. As shown in Table 1, our method successfully recovers the original force fields with low numerical errors. These quantitative results provide strong evidence that our approach accurately estimates the underlying force dynamics and generalizes well across various objects and physical properties.

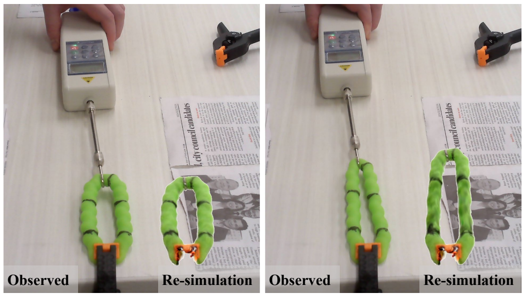

Figure 3: Comparison of observed data (left in each frame) and re-simulated results (right in each frame) for two different frames in the real-world experiment.

Material type	Object	PSNR	SSIM	LPIPS	Mag. Error (%)	Dir. Error (°)
Elastic	Lego	33.70	0.98	0.01	19.53	7.02
	Ficus	25.92	0.94	0.03	23.97	11.55
	Sunflower	34.08	0.99	0.01	14.38	7.85
Elastoplastic	Toy	41.35	0.99	0.00	29.19	8.11
	Chair	40.10	0.99	0.00	33.31	23.40
Viscoplastic	Hotdog	30.63	0.96	0.02	15.09	11.63

Table 1: Force recovery in synthetic scenarios.

Material Type	Method	PSNR	SSIM	LPIPS	Mag. Error (%)	Dir. Error (°)
Elastic	Point	20.57	0.94	0.04	95.91	76.48
	K-Planes	26.25	0.96	0.03	18.03	39.83
	Ours	**39.79**	**0.99**	**0.01**	**5.14**	**4.38**
Elastoplastic	Point	31.40	0.98	0.01	98.36	**26.8**
	K-planes	30.14	0.98	0.01	87.81	61.42
	Ours	**39.93**	**0.99**	0.01	**75.06**	45.50
Viscoplastic	Point	17.49	0.95	0.12	98.30	92.02
	K-planes	**41.73**	0.99	0.03	89.04	33.09
	Ours	39.00	**0.99**	**0.01**	**21.44**	**7.50**

Table 2: Quantitative comparison of force representations.

Loss functions	PSNR	SSIM	LPIPS	Mag. Error(%)	Dir. Error (°)
Image Loss	37.24	0.99	0.01	86.74	50.23
Flow+Depth Loss	**41.54**	0.99	0.01	27.90	16.07
Ours	39.79	**0.99**	**0.01**	**5.14**	**4.38**

Table 3: Quantitative comparison of loss functions.

Material Type	#Samples	Type F1	ρ MAPE (%)	E log-MAPE(%)
Elastic	6	1	2.38	5.31
Elastoplastic	6	1	10.80	3.36
Viscoplastic	5	1	0	13.98
Overall	17	1	4.65	7.17

Table 4: VLM performance on the daily-item dataset.

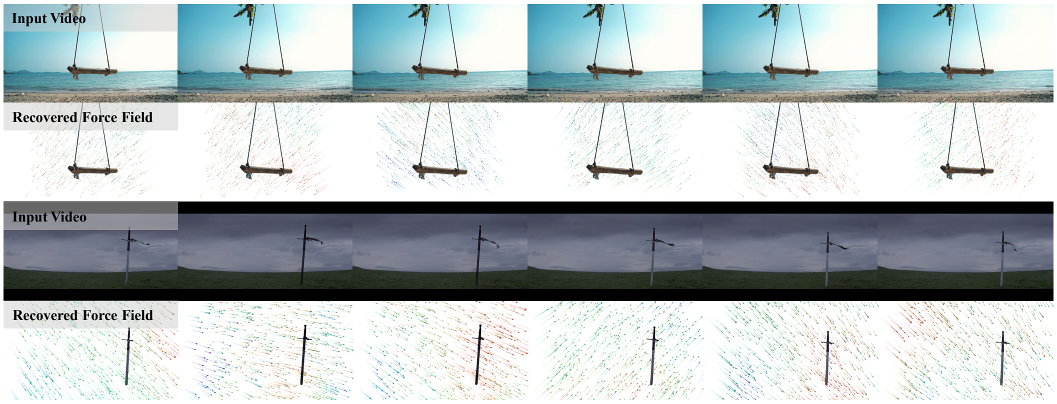

Figure 4: Our method estimates invisible force fields from real-world videos, producing physically plausible motion interpretations.

4.3 Empirical Study

Force Representations. To evaluate the effectiveness of our force representation in Section 3.2, we compare our causal tri-plane with other 4D representations such as K-planes [56] and point contact forces. The results in Table 2 demonstrate that our causal tri-plane force representation performs better than other 4D representations. This is mainly because our force representation accurately models the spatial continuity and temporal dependence of forces.

Loss Functions. To evaluate the effectiveness of our loss functions in Section 3.4, we compare our sparse tracking loss with the image reconstruction loss and the dense 3D scene flow loss derived from depth and flow estimation (Flow+Depth loss). The results in Table 3 demonstrate that our sparse tracking loss shows better performance than others, especially in force accuracy. This is because sparse tracking provides more robust motion estimates that can be leveraged as more accurate signals to optimize the forces.

VLM Material-Property Estimation. To evaluate the robustness of leveraging a vision-language model for physical parameter estimation, we measure the estimation errors by utilizing the GPT-4o-Vision to infer material type and material properties(density ρ and Young's modulus E) from a single image and comparing with the ground truth values. A small benchmark of 17 everyday objects

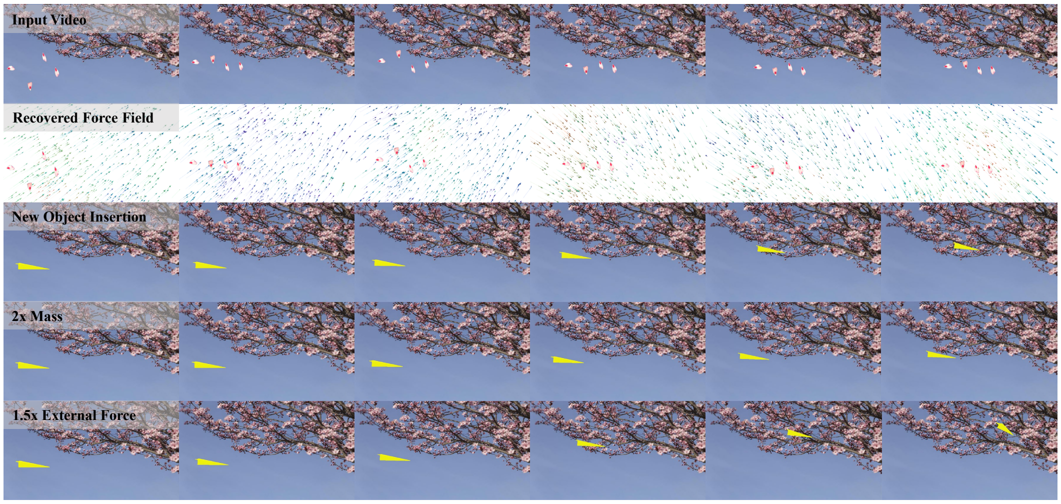

Figure 5: Our method recovers force fields from input videos and enables the insertion of novel objects while maintaining physically plausible motion. We demonstrate new object insertion, and modifications of physical conditions (e.g. mass and external force), showcasing the model's ability to generate physically plausible videos.

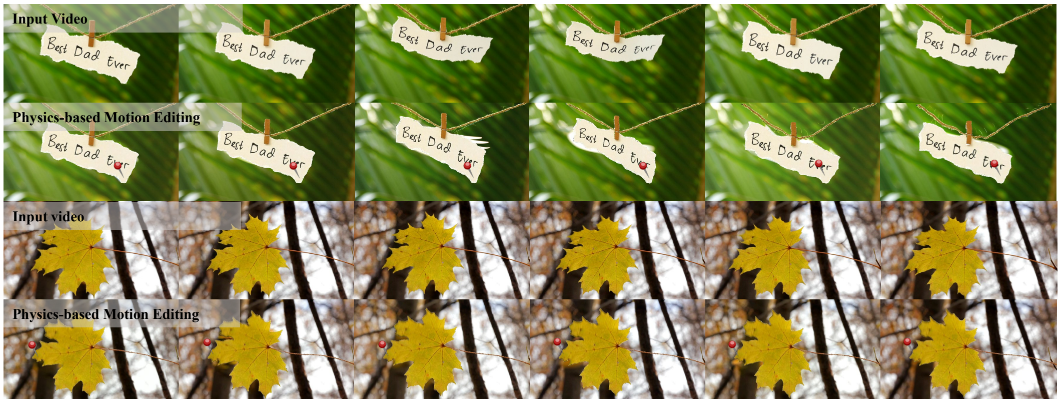

Figure 6: Our method allows modifying object motion by adjusting external constraints while preserving physical realism. We demonstrate how altering boundary conditions (e.g., fixing parts of an object) influences motion under the same estimated force field. These results highlight the flexibility of our approach for controllable, physics-based video editing.

is collected, each made of a single, well-documented engineering material. Ground-truth ρ and E values are taken from MatWeb database[66]. Results are summarised in Table 4. The model achieves an overall F1 of 1.0 for material classification, an 4.65% mean-absolute-percentage error (MAPE) on density, and a log-MAPE of 7.17%, which indicates that the VLM can have a robust initial estimate of the physical parameters.

Moving Cameras For non-stationary cameras we adopt 4D reconstruction pipelines to supply camera poses and trajectories, which we consume without retraining. Qualitative videos for dynamic-camera sequences are best viewed on the project page.

4.4 Physics-based Generation

With the recovered force field, we demonstrate the potential of our approach for physics-based video generation and editing. Figure 5 shows the results of physics-based video generation. Our framework enables the replacement of novel objects within the same force field, generating physically plausible motions for novel objects via physics simulation. This flexibility also allows us to modify the object's physical properties and force strengths to create distinct object motions that adhere to physical laws. Compared to other video generation methods, our approach produces more controllable and physically-accurate videos. Figure 6 shows the results of physics-based video editing, where we can modify the boundary conditions in the scenes, *e.g.*, fixing a point, to generate different physically plausible object motions in the same video. These results highlight the versatility of our framework in generating and editing videos while maintaining physical consistency.

We qualitatively compare against interactive editing / motion-driven methods [48, 49] and velocity-field learning baselines [50, 51] on matched inputs. These approaches optimize displacements or velocities under strong priors and thus yield kinematically plausible results, but they do not estimate identifiable physical forces and do not enforce Newtonian consistency when physical parameters are edited (e.g., doubling mass). Consequently, the resulting motion may continue to "match" an appearance prior yet diverge from the dynamics implied by the edited parameters. By explicitly recovering time-varying forces inside a differentiable simulator, our method preserves dynamical consistency under parameter edits and non-uniform fields. Qualitative videos are best viewed on the project page.

5 Conclusion

In this work, we introduced a differentiable inverse graphics framework to recover invisible forces from video object motions, bridging vision and physics. By modeling object properties, forces, and physical processes, our method enables robust force estimation via backpropagation. Experiments in both real-world and synthetic data have demonstrated accurate force recovery and controllable physics-based video generation and editing of our approach.

Limitations. Our framework is primarily applied to objects with small deformation or bending-only deformation. Fluids or other object types that require different, differentiable physical processes are out of the scope of this paper, which is left for future works.

As in our demo, we model foreground physics and composite over a static background, as off-the-shelf per-frame inpainting can introduce temporal flicker that obscures our contribution.

Acknowledgments and Disclosure of Funding

This work is in part supported by NSF RI #2211258 and #2338203, ONR MURI N00014-22-1-2740, and the Okawa Research Grant. The USC Geometry, Vision, and Learning Lab acknowledges generous supports from Toyota Research Institute, Dolby, Google DeepMind, Capital One, Nvidia, and Qualcomm. Yue Wang is also supported by a Powell Research Award.

References

[1] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

[2] Chenfanfu Jiang, Craig Schroeder, Joseph Teran, Alexey Stomakhin, and Andrew Selle. The material point method for simulating continuum materials. In *ACM SIGGRAPH 2016 courses*, pages 1–52, 2016.

[3] Jiajun Wu, Ilker Yildirim, Joseph J Lim, Bill Freeman, and Josh Tenenbaum. Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. *Advances in neural information processing systems*, 28, 2015.

[4] Jiajun Wu, Joseph J Lim, Hongyi Zhang, Joshua B Tenenbaum, and William T Freeman. Physics 101: Learning physical object properties from unlabeled videos. In *BMVC*, volume 2, page 7, 2016.

[5] Jiajun Wu, Erika Lu, Pushmeet Kohli, Bill Freeman, and Josh Tenenbaum. Learning to see physics via visual de-animation. *Advances in neural information processing systems*, 30, 2017.

[6] Jennifer Cardona, Michael Howland, and John Dabiri. Seeing the wind: Visual wind speed prediction with a coupled convolutional and recurrent neural network. *Advances in Neural Information Processing Systems*, 32, 2019.

[7] David Hahn, Pol Banzet, James M. Bern, and Stelian Coros. Real2sim: visco-elastic parameter estimation from dynamic motion. *ACM Trans. Graph.*, 38(6), November 2019.

[8] Krishna Murthy Jatavallabhula, Miles Macklin, Florian Golemo, Vikram Voleti, Linda Petrini, Martin Weiss, Breandan Considine, Jérôme Parent-Lévesque, Kevin Xie, Kenny Erleben, et al. gradsim: Differentiable simulation for system identification and visuomotor control. *arXiv preprint arXiv:2104.02646*, 2021.

[9] Mingyu Ding, Zhenfang Chen, Tao Du, Ping Luo, Josh Tenenbaum, and Chuang Gan. Dynamic visual reasoning by learning differentiable physics models from video and language. *Advances In Neural Information Processing Systems*, 34:887–899, 2021.

[10] Xuan Li, Yi-Ling Qiao, Peter Yichen Chen, Krishna Murthy Jatavallabhula, Ming Lin, Chenfanfu Jiang, and Chuang Gan. Pac-nerf: Physics augmented continuum neural radiance fields for geometry-agnostic system identification. *arXiv preprint arXiv:2303.05512*, 2023.

[11] Ri-Zhao Qiu, Ge Yang, Weijia Zeng, and Xiaolong Wang. Feature splatting: Language-driven physics-based scene synthesis and editing, 2024.

[12] Albert J Zhai, Yuan Shen, Emily Y Chen, Gloria X Wang, Xinlei Wang, Sheng Wang, Kaiyu Guan, and Shenlong Wang. Physical property understanding from language-embedded feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28296–28305, 2024.

[13] Junhao Cai, Yuji Yang, Weihao Yuan, Yisheng He, Zilong Dong, Liefeng Bo, Hui Cheng, and Qifeng Chen. Gaussian-informed continuum for physical property identification and simulation. *arXiv preprint arXiv:2406.14927*, 2024.

[14] Licheng Zhong, Hong-Xing Yu, Jiajun Wu, and Yunzhu Li. Reconstruction and simulation of elastic objects with spring-mass 3d gaussians. In *European Conference on Computer Vision*, pages 407–423. Springer, 2025.

[15] Xingrui Wang, Wufei Ma, Angtian Wang, Shuo Chen, Adam Kortylewski, and Alan Yuille. Compositional 4d dynamic scenes understanding with physics priors for video question answering. *arXiv preprint arXiv:2406.00622*, 2024.

[16] Wei Chow, Jiageng Mao, Boyi Li, Daniel Seita, Vitor Guizilini, and Yue Wang. Physbench: Benchmarking and enhancing vision-language models for physical world understanding. *arXiv preprint arXiv:2501.16411*, 2025.

[17] 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 *European Conference on Computer Vision*, pages 388–406. Springer, 2025.

[18] Fangfu Liu, Hanyang Wang, Shunyu Yao, Shengjun Zhang, Jie Zhou, and Yueqi Duan. Physics3d: Learning physical properties of 3d gaussians via video diffusion. *arXiv preprint arXiv:2406.04338*, 2024.

[19] Tianyu Huang, Yihan Zeng, Hui Li, Wangmeng Zuo, and Rynson WH Lau. Dreamphysics: Learning physical properties of dynamic 3d gaussians with video diffusion priors. *arXiv preprint arXiv:2406.01476*, 2024.

[20] Quentin Garrido, Nicolas Ballas, Mahmoud Assran, Adrien Bardes, Laurent Najman, Michael Rabbat, Emmanuel Dupoux, and Yann LeCun. Intuitive physics understanding emerges from self-supervised pretraining on natural videos. *arXiv preprint arXiv:2502.11831*, 2025.

[21] Tao Du, Kui Wu, Pingchuan Ma, Sebastien Wah, Andrew Spielberg, Daniela Rus, and Wojciech Matusik. Diffpd: Differentiable projective dynamics. *ACM Trans. Graph.*, 41(2), nov 2021.

[22] William Gilpin. Generative learning for nonlinear dynamics. *Nature Reviews Physics*, 6(3):194–206, 2024.

[23] Eric Heiden, David Millard, Erwin Coumans, Yizhou Sheng, and Gaurav S Sukhatme. Neuralsim: Augmenting differentiable simulators with neural networks. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9474–9481. IEEE, 2021.

[24] 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 Transactions on Graphics (TOG)*, 37(4):1–14, 2018.

[25] Yuanming Hu, Tzu-Mao Li, Luke Anderson, Jonathan Ragan-Kelley, and Frédo Durand. Taichi: a language for high-performance computation on spatially sparse data structures. *ACM Transactions on Graphics (TOG)*, 38(6):201, 2019.

[26] Yuanming Hu, Luke Anderson, Tzu-Mao Li, Qi Sun, Nathan Carr, Jonathan Ragan-Kelley, and Frédo Durand. Difftaichi: Differentiable programming for physical simulation. *ICLR*, 2020.

[27] Yuanming Hu, Jiafeng Liu, Xuanda Yang, Mingkuan Xu, Ye Kuang, Weiwei Xu, Qiang Dai, William T. Freeman, and Frédo Durand. Quantaichi: A compiler for quantized simulations. *ACM Transactions on Graphics (TOG)*, 40(4), 2021.

[28] Junior Rojas, Eftychios Sifakis, and Ladislav Kavan. Differentiable implicit soft-body physics. *arXiv preprint arXiv:2102.05791*, 2021.

[29] Zizhou Huang, Davi Colli Tozoni, Arvi Gjoka, Zachary Ferguson, Teseo Schneider, Daniele Panozzo, and Denis Zorin. Differentiable solver for time-dependent deformation problems with contact. *ACM Transactions on Graphics*, 43(3):1–30, May 2024.

[30] Barza Nisar, Philipp Foehn, Davide Falanga, and Davide Scaramuzza. Vimo: Simultaneous visual inertial model-based odometry and force estimation. *IEEE Robotics and Automation Letters*, 4(3):2785–2792, 2019.

[31] Xingyu Chen, Jialei Shi, Helge Wurdemann, and Thomas George Thuruthel. Vision-based tip force estimation on a soft continuum robot. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7621–7627. IEEE, 2024.

[32] Wenzhen Yuan, Rui Li, Mandayam A Srinivasan, and Edward H Adelson. Measurement of shear and slip with a gelsight tactile sensor. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 304–311. IEEE, 2015.

[33] Eric Heiden, Miles Macklin, Yashraj Narang, Dieter Fox, Animesh Garg, and Fabio Ramos. Disect: A differentiable simulation engine for autonomous robotic cutting. *arXiv preprint arXiv:2105.12244*, 2021.

[34] Haozhe Lou, Yurong Liu, Yike Pan, Yiran Geng, Jianteng Chen, Wenlong Ma, Chenglong Li, Lin Wang, Hengzhen Feng, Lu Shi, et al. Robo-gs: A physics consistent spatial-temporal model for robotic arm with hybrid representation. *arXiv preprint arXiv:2408.14873*, 2024.

[35] Hyun Soo Park, Jianbo Shi, et al. Force from motion: decoding physical sensation in a first person video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3834–3842, 2016.

[36] Zongmian Li, Jiri Sedlar, Justin Carpentier, Ivan Laptev, Nicolas Mansard, and Josef Sivic. Estimating 3d motion and forces of person-object interactions from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8640–8649, 2019.

[37] Davis Rempe, Leonidas J Guibas, Aaron Hertzmann, Bryan Russell, Ruben Villegas, and Jimei Yang. Contact and human dynamics from monocular video. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 71–87. Springer, 2020.

[38] Kiana Ehsani, Shubham Tulsiani, Saurabh Gupta, Ali Farhadi, and Abhinav Gupta. Use the force, luke! learning to predict physical forces by simulating effects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 224–233, 2020.

[39] Yufei Zhang, Jeffrey O Kephart, Zijun Cui, and Qiang Ji. Physpt: Physics-aware pretrained transformer for estimating human dynamics from monocular videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2305–2317, 2024.

[40] Bin Wang, Longhua Wu, KangKang Yin, Uri M Ascher, Libin Liu, and Hui Huang. Deformation capture and modeling of soft objects. *ACM Trans. Graph.*, 34(4):94–1, 2015.

[41] Gengshan Yang, Shuo Yang, John Z Zhang, Zachary Manchester, and Deva Ramanan. Ppr: Physically plausible reconstruction from monocular videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3914–3924, 2023.

[42] Yandan Yang, Baoxiong Jia, Peiyuan Zhi, and Siyuan Huang. Physcene: Physically interactable 3d scene synthesis for embodied ai. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16262–16272, 2024.

[43] Katerina Fragkiadaki, Pulkit Agrawal, Sergey Levine, and Jitendra Malik. Learning visual predictive models of physics for playing billiards. *arXiv preprint arXiv:1511.07404*, 2015.

[44] Zhengqi Li, Richard Tucker, Noah Snavely, and Aleksander Holynski. Generative image dynamics, 2024.

[45] Shaowei Liu, Zhongzheng Ren, Saurabh Gupta, and Shenlong Wang. Physgen: Rigid-body physics-grounded image-to-video generation. In *European Conference on Computer Vision*, pages 360–378. Springer, 2024.

[46] Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4389–4398, 2024.

[47] Chayan Banerjee, Kien Nguyen, Clinton Fookes, and Karniadakis George. Physics-informed computer vision: A review and perspectives. *ACM Computing Surveys*, 57(1):1–38, 2024.

[48] Xingang Pan, Ayush Tewari, Thomas Leimkühler, Lingjie Liu, Abhimitra Meka, and Christian Theobalt. Drag your gan: Interactive point-based manipulation on the generative image manifold. In *ACM SIGGRAPH 2023 conference proceedings*, pages 1–11, 2023.

[49] Daniel Geng, Charles Herrmann, Junhwa Hur, Forrester Cole, Serena Zhang, Tobias Pfaff, Tatiana Lopez-Guevara, Carl Doersch, Yusuf Aytar, Michael Rubinstein, Chen Sun, Oliver Wang, Andrew Owens, and Deqing Sun. Motion prompting: Controlling video generation with motion trajectories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[50] Jinxi Li, Ziyang Song, and Bo Yang. Nvfi: Neural velocity fields for 3d physics learning from dynamic videos. *Advances in Neural Information Processing Systems*, 36:34723–34751, 2023.

[51] Jinxi Li, Ziyang Song, Siyuan Zhou, and Bo Yang. Freegave: 3d physics learning from dynamic videos by gaussian velocity. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12433–12443, 2025.

[52] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10106–10116, 2024.

[53] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[54] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.

[55] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16123–16133, 2022.

[56] Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12479–12488, 2023.

[57] Ang Cao and Justin Johnson. Hexplane: A fast representation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 130–141, 2023.

[58] Haotong Lin, Qianqian Wang, Ruojin Cai, Sida Peng, Hadar Averbuch-Elor, Xiaowei Zhou, and Noah Snavely. Neural scene chronology. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20752–20761, 2023.

[59] Qianqian Wang, Vickie Ye, Hang Gao, Weijia Zeng, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4d reconstruction from a single video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9660–9672, 2025.

[60] Quankai Gao, Qiangeng Xu, Zhe Cao, Ben Mildenhall, Wenchao Ma, Le Chen, Danhang Tang, and Ulrich Neumann. Gaussianflow: Splatting gaussian dynamics for 4d content creation, 2024.

[61] Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast and robust structure and motion from casual dynamic videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 10486–10496, 2025.

[62] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. In *European Conference on Computer Vision*, pages 18–35. Springer, 2024.

[63] 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. *Communications of the ACM*, 65(1):99–106, 2021.

[64] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[65] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric, 2018.

[66] MatWeb, LLC. Matweb: Online materials information resource, 2025. Accessed: 2025-05-13.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our papers main contribution has been clearly declared in the abstract and introduction

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We include the limitation part in conclusion section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We present a comprehensive formulation of our proposed framework, with detailed descriptions provided in each subsection.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: A detailed description of our method is provided, and a demonstration code will be released upon acceptance.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The code is not included for now. A demonstration code will be released upon acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We include them in implementation details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We perform the experiments multiple times and compute the average. We also have 2-sigma error results in appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide sufficient information on the computer resources for experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We follow the NeurIPS Code of Ethics

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discussed the potential positive social impact of our work, as we outlined some of its applications in the main text. As for negative social impacts, since this work represents a relatively new and early-stage direction, we have not yet identified significant negative implications.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The codebase we use has been properly cited in the paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The demo dataset will be released with well documentation.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: We clearly state the usage of VLM in our method.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.