PhysChoreo: Physics-Controllable Video Generation with Part-Aware Semantic Grounding

Haoze Zhang[1], Tianyu Huang[1], Zichen Wan[1], Xiaowei Jin[1], Hongzhi Zhang[1], Hui Li[1], and Wangmeng Zuo[1⋆]

[1]Harbin Institute of Technology, China
kakaka0521@outlook.com
Project: https://kakaka0521.github.io/physchoreo_web

Abstract. While recent video generation models have achieved significant visual fidelity, they often suffer from the lack of explicit physical controllability and plausibility. To address this, some recent studies attempted to guide video generation with physics-based rendering. However, these methods face inherent challenges in accurately modeling complex physical properties and effectively controlling the resulting physical behavior over extended temporal sequences. In this work, we introduce PhysChoreo, a framework that can generate videos with diverse controllability and physical realism from a single image. Our method consists of two stages: first, it estimates the static initial physical properties of all objects in the image through part-aware physical property reconstruction. Then, through temporally instructed and physically editable simulation, it synthesizes high-quality videos with rich dynamic behaviors and physical realism. Experimental results show that PhysChoreo can generate videos with rich behaviors and physical realism, outperforming state-of-the-art methods on multiple evaluation metrics.

Keywords: physics-based video generation · controllable generation · physical property reconstruction

1 Introduction

Recent advances in video generation [2, 4, 10, 46, 53] have significantly improved visual fidelity and spatiotemporal consistency, yet the physical realism of generated content remains limited [16, 47]. Contemporary models primarily scale up by learning from vast amounts of data to capture physical phenomena, rather than understanding the underlying principles that govern how objects respond to forces and material constraints. As a consequence, generated videos often fail to accurately present real-world behavior, particularly in complex or even counterfactual scenarios. These limitations reveal that current models highly rely on simplistic imitation of motion patterns, lacking causal reasoning about physical rules.

⋆ Corresponding author. Email: wmzuo@hit.edu.cn

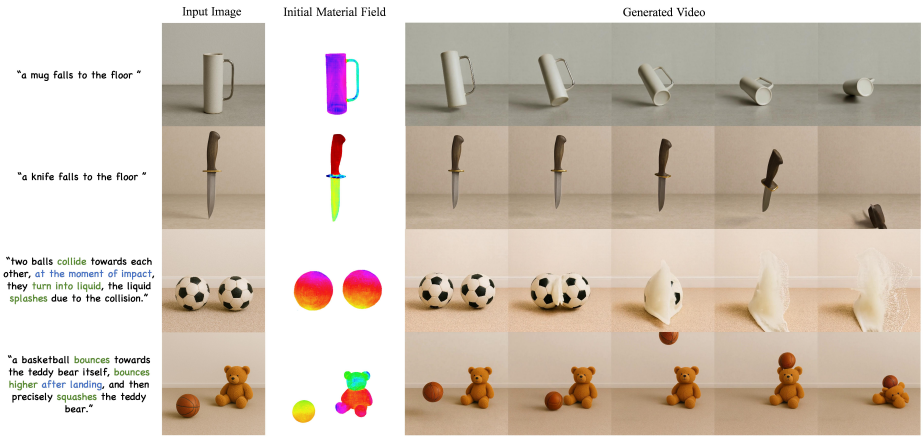

Fig. 1: We propose PhysChoreo, a framework for controllable image-to-video generation. PhysChoreo reconstructs the material field of objects from a single image and generates physically realistic and dynamically rich videos. In (a) and (b), physically realistic dynamics are generated based on reconstructed physical properties. In (c) and (d), by controlling physical properties during generation, more cinematic videos can be produced while maintaining physical realism.

To produce physically plausible outcomes, recent works [14, 22, 41, 49, 55] have attempted to incorporate physical simulation into the generation procedure. Specifically, these methods employ finite element simulation to compute the motion state of the target objects, which is then used as the condition to render or generate corresponding dynamic effects. Since the process of physical simulation is deterministic, the key challenges become: (1) modeling the physics fields of the target scene, and (2) controlling the generated content with simulated results. However, existing methods have not effectively addressed these two issues. On the one hand, physics optimization approaches [14, 22, 55] typically impose coarse prediction on single objects, which fails to meet the modeling requirements of real-world simulation. On the other hand, simulators [15, 32, 49] used in these methods are not as flexible in control as video models, resulting in limited simulation modes.

We address these limitations with **PhysChoreo**, a physics-based video generation framework that consists of *part-aware semantic physics* prediction and *physics-editable control*. The key idea for *part-aware semantic physics* is to align textual semantics with detailed 3D part structures. For each part in an object, both the material model and calibrated continuous physical quantities are estimated with our pre-trained predictor. Consequently, a part-level physics supervision strategy is proposed, which tightly couples semantics, geometry, and physics at fine granularity, enabling our predictor to reconcile physical interpretability and controllability. Beyond static physics estimation, we integrate physics-based simulation [11–13] with *physics-editable control* directly injected into the generative process. Our control allows the model to modulate external

forces and material attributes over time, so that objects respond in a physically plausible manner throughout a sequence. This formulation converts high-level instructions into temporally consistent physical changes, making counterfactual interventions such as liquefying upon collision, collapsing upon landing, counter-intuitive bounces, both natural and verifiable.

To support part-level physics modeling, we construct a fine-grained *text–part–physics* dataset that covers common materials and canonical part relations across a variety of objects. For each instance, the dataset provides a global textual description and part-level phrase prompts that capture functional or material cues. Both point-level and part-level labels are annotated, including discrete material categories together with key continuous physical quantities, *i.e.*, Young's modulus E, Poisson's ratio ν, and density ρ. This resource serves as a unified benchmark for training and evaluating per-part physics estimation.

We conduct extensive experiments to evaluate the prediction of physical properties and the quality of physics-controllable video generation. For *part-level physics* prediction, the proposed method advances the state-of-the-art in continuous parameter regression, maintaining robust generalization across different categories and scenarios. For *physics-controllable* video generation, we used more complex and diverse instructions to generate videos with various visual effects, and systematically evaluate the physical realism, instruction following, and visual quality of the generated videos, showing that PhysChoreo can faithfully realize instruction-driven changes while maintaining high visual fidelity and temporal coherence.

Our contributions can be summarized as:

- We release a text–part–physics dataset with consistent semantic, geometric, and physical annotations, establishing a reusable benchmark for training and evaluation.
- We propose to predict *part-level* semantic physics modeling via soft assignment and hierarchical cross-attention, which is supervised by multiple physics constraints.
- We propose *physics-editable control* in dynamics simulation, which formalizes editing as temporally continuous intervention, enabling flexible, fine-grained modulation and interaction.
- We introduce PhysChoreo, a physics-based video generation framework that reconstructs physical properties and generates diverse controllable dynamic sequences.

2 Related Work

Physical Property Estimation. Existing methods can be broadly divided into two categories: runtime optimization and direct estimation. Runtime optimization methods [14,22,55] gradually optimize physical properties through iterative processes guided by rendered videos and diffusion models. However, these methods can't supervise fine-grained physical modeling and incur significant time overhead, making them difficult to use for direct video generation. Among direct

estimation methods, NeRF2Physics [54] and PUGS [38] use multi-view images to enable VLMs to infer physical properties. However, they both rely on generated 2D features and cannot be applied to general 3D representations. Pixie [18] injects features in 3D through multi-view images and performs direct estimation, but it also cannot be directly applied to general 3D representations and can only learn static features rather than their distributions.

Physics-Based Video Generation. Existing methods utilize physical simulators to generate physics-based videos. Some methods, such as PhysGaussian [49] and subsequent work [14, 22, 27, 30, 55], reconstruct scene representations from multi-view images, simulate these representations, and then render videos. However, these methods rely on high-quality 3D reconstruction. Recent methods [3, 25, 41] consider generating dynamics using physical simulators from a single image and synthesizing them, while others [7, 20, 48] consider using dynamics as guidance to assist video models in generation. PhysCtrl [44] directly uses diffusion models to generate dynamics and then employs video models for generation. However, these methods depend on manual physics parameter settings and lack fine-grained temporal control, resulting in short and monotonous generated dynamics.

Controllable Generative Video Model. Generative video models are trained on large-scale data to acquire the ability to generate videos and can produce high-quality videos [4–6, 50, 53], but they often lack controllability. Recent methods have explored various approaches for controllable video generation, where optical flow [21, 33], motion [9, 19], and others can serve as control conditions to guide generation. However, they still lack physical plausibility and controllability. A key aspect of our work is generating rich dynamics to guide the generation process, aiming to achieve physically realistic videos.

3 Method

In this section, we introduce **PhysChoreo**, a novel framework for physics-controllable video generation. Given 3D objects reconstructed from a single image, PhysChoreo conducts a part-aware physics prediction via soft assignment and hierarchical cross-attention (See Sec. 3.1). A part-level physics supervision is then proposed to facilitate the training of the physics predictor in terms of local smoothness and prompt guidance (See Sec. 3.2). Finally, the predicted physics field drives controllable simulators and video model to produce editable, physically consistent videos (See Sec. 3.3).

3.1 Part-Aware Physics Reconstruction

Given an input image I, our intention is to reconstruct the corresponding scene and assign reasonable physical properties to all objects in the scene. To this end, we segment all the instances in I and then reconstruct a dense triangular mesh for each instance with InstantMesh [51]. We obtain the point cloud representation

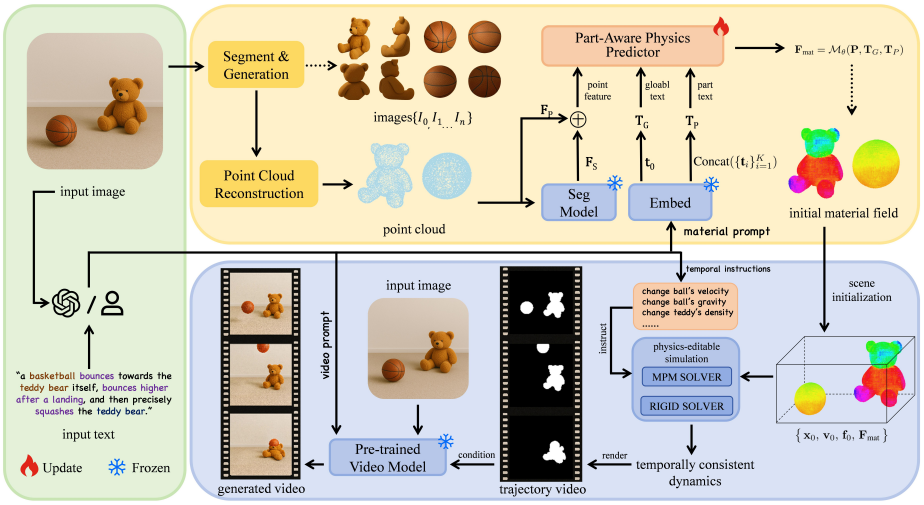

Fig. 2: Overview of our pipeline. Given the input image and text prompt, we first reconstruct the initial material field of each object from the image. Then we generate the scene's trajectory video based on a physics-editable simulator with temporal instructions, and finally the trajectory video is used as conditional control to guide the generation of generative video model.

$P_i \in \mathbb{R}^{N\times 3}$ by uniformly sampling N points on the surface of the i-th mesh, which is the target of the following physics prediction.

To specify the material, roughness, and other related physical properties of P_i, an input text prompt y_0 is provided, which is then encoded into text embeddings $\mathbf{t}_0 \in \mathbb{R}^{d_t}$ with CLIP [35]. Considering that one object may contain multiple materials, part-level prompts $\{y_i\}_{i=1}^K$(K denotes the number of part prompts) are optionally provided to describe the detailed properties for different parts of P_i, which are also embedded as $\{\mathbf{t}_i\}_{i=1}^K$. Accordingly, we extract global point features $\mathbf{F}_P \in \mathbb{R}^{N\times d_P}$ and part-level semantic features $\mathbf{F}_S \in \mathbb{R}^{N\times d_S}$ with a lightweight MLP and a pre-trained part segmentation encoder [23, 24, 40, 52], respectively. In order to convert text instructions into a point-based physics field that is globally coherent and locally editable, we propose soft assignment to align the feature dimension of point cloud with part-level text embeddings and hierarchical cross-attention to combine multi-granularity text embeddings with point features.

Soft Assignment. Our aim is to inject part-level semantics into point features in a way that is editable, stable, and interpretable, i.e., each point should carry an explicit distribution over part prompts while its geometric signal is preserved. Therefore, we align points and prompts in a shared latent space and use the resulting weights to mix prompt "values" back into the point stream via a residual update. Concretely, we formulate an additive refinement of point features $\mathbf{H} = \text{Concat}(\mathbf{F}_S, \mathbf{F}_P) \in \mathbb{R}^{N\times d}$ with part-prompt embeddings $\mathbf{T} = \text{Concat}(\{\mathbf{t}_i\}_{i=1}^K) \in$

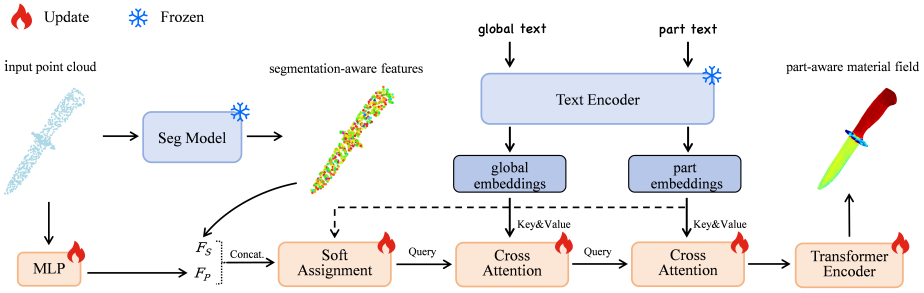

Fig. 3: Overview of our model design. We first use a fused feature from point positional feature and segmentation prior. Afterward, we use a soft assignment to preliminarily display the injected part-level features, then perform fine-grained text-level adjustments through a hierarchical cross-attention stage, and finally obtain part-aware material field features via a transformer encoder.

$\mathbb{R}^{K\times d_t}$ as:

$$A = \mathrm{softmax}_{\mathrm{row}}((\mathbf{H}\Phi)\,(\mathbf{T}\Psi)^{\top}) \in \mathbb{R}^{N\times K}, \tag{1}$$

$$\hat{\mathbf{H}} = \mathbf{H} + A\,(\mathbf{T}W) \in \mathbb{R}^{N\times d}, \tag{2}$$

where $\Phi \in \mathbb{R}^{d\times d_a}$ and $\Psi \in \mathbb{R}^{d_t\times d_a}$ project points and prompts to the alignment space with dimension d_a, $W \in \mathbb{R}^{d_t\times d}$ maps prompt "values" to the point-feature space, and $\mathrm{softmax}_{\mathrm{row}}$ normalizes across the K prompts for each point so that each row of A sums to 1. Algebraically, this block is equivalent to a *single-head cross-attention*, but we deliberately adopt this minimal form so that the assignment distribution A remains directly interpretable and its logits S can be explicitly supervised by L_{assign} in 3.2.

Hierarchical Cross-Attention. To guide the point stream with global semantics before part-level details refine it, we propose a hierarchical cross-attention mechanism to support the interaction of the point stream and the multi-level text stream. Let $\mathbf{T}_0 = [\mathbf{t}_0] \in \mathbb{R}^{1\times d_t}$ be the global text token and $\mathbf{T}$ be the part-level token. Queries always come from the point stream, while keys/values come from text:

$$\mathbf{H}_g = \mathrm{MHA}(\hat{\mathbf{H}},\, \mathbf{T}_0) + \hat{\mathbf{H}}, \tag{3}$$

$$\mathbf{H}_p = \mathrm{MHA}(\mathbf{H}_g,\, \mathbf{T}) + \mathbf{H}_g, \tag{4}$$

The first stage uses a single global token to impose scene-level consistency on all points, acting as a coarse "style" conditioner that stabilizes subsequent language injection. The second stage then attends to the part tokens, sharpening locality and disentangling part-specific effects without overriding the coarse global guidance. Placing the global stage before the part stage reduces competition between tokens and empirically improves convergence and editability.

To aggregate non-local context while remaining permutation-invariant over points, the part-conditioned features $\mathbf{H}_p$ are encoded by a set Transformer encoder [56] to produce point embeddings $Z = Transformer(\mathbf{H}_p) \in \mathbb{R}^{N \times d_z}$. Then, standard point-wise heads decode (i) class logits followed by a softmax to yield per-point material model class probabilities $\hat{Y} \in \mathbb{R}^{N \times C}$ and (ii) continuous material parameters $\hat{M} \in \mathbb{R}^{N \times q}$. For each point i, the material field is represented as

$$M_{\text{pred}}(x_i) = \Big(\underset{c \in \{1,\dots,C\}}{\arg\max}\ \hat{Y}_{i,c},\ \hat{M}_{i,:} \Big). \tag{5}$$

3.2 Part-Level Physics Supervision

We couple point-wise supervision with structural priors so that the learned material field is (i) correct at each point, (ii) smooth *within* semantic parts, and (iii) aligned with the part prompts used for conditioning. Let $\{x_i\}_{i=1}^{N}$ be sampled 3D points on the object surface/volume, where $x_i \in \mathbb{R}^3$ and N is the number of supervised points. For each point i, the network predicts continuous material parameters $\hat{m}_i$ (e.g., $(\hat{E}_i, \hat{\nu}_i, \hat{\rho}_i)$) and a material-model class distribution $\hat{y}_i^{\text{cls}} \in \Delta^{C-1}$ over C classes. Ground truth is given by m_i and a class label $y_i^{\text{cls}} \in \{1, \dots, C\}$. We write $\text{CE}(\hat{y}_i^{\text{cls}}, y_i^{\text{cls}}) = -\log \hat{y}_{i,y_i^{\text{cls}}}^{\text{cls}}$ and use a smooth regression loss $\ell(\hat{m}_i, m_i)$ on normalized targets.

Task Supervision. This term anchors the semantic identity and calibrates physical magnitudes on a per-point basis:

$$L_{\text{task}} = \frac{1}{N} \sum_{i=1}^{N} \Big[\lambda_{\text{reg}} \cdot \ell\big(\hat{m}_i,\, m_i\big) + \lambda_{\text{cls}} \cdot \text{CE}\left(\hat{y}_i^{\text{cls}},\, y_i^{\text{cls}}\right) \Big], \tag{6}$$

Wave Continuity. In both the physical world and simulation environments, the propagation speed of elastic waves, as determined by material parameters such as Young's modulus, Poisson's ratio, and density, is an intrinsic reflection of the material's mechanical characteristics [34]. To ensure the learning of relationships between physical properties and spatial continuity, we regularize the predicted longitudinal and shear wave speeds. Given material parameters (E_i, ν_i, ρ_i) at point x_i, we compute

$$c_p(x_i) = \sqrt{\frac{E_i(1-\nu_i)}{\rho_i(1+\nu_i)(1-2\nu_i)}}, \qquad c_s(x_i) = \sqrt{\frac{E_i}{2\rho_i(1+\nu_i)}}, \tag{7}$$

where E_i is Young's modulus, ν_i is Poisson's ratio, and ρ_i is density.

We then approximate the continuous smoothness penalty $\int \|\nabla c_p(x)\|_2^2 + \|\nabla c_s(x)\|_2^2 \, dx$ on sampled points using a within-part neighborhood graph. Let $\mathcal{N}(i)$ be the k-nearest-neighbor set of point i in 3D, and let $\text{part}(i)$ be its ground-truth semantic part label. We define the within-part edge set $\mathcal{E} = \{(i,j) \mid j \in \mathcal{N}(i),\ \text{part}(i) = \text{part}(j)\}$ and minimize

$$L_{\text{smooth}} = \frac{1}{|\mathcal{E}|} \sum_{(i,j) \in \mathcal{E}} w_{ij} \Big(\big(c_p(x_i) - c_p(x_j)\big)^2 + \big(c_s(x_i) - c_s(x_j)\big)^2 \Big), \tag{8}$$

where $|\mathcal{E}|$ is the number of within-part edges and $w_{ij} = (\|x_i - x_j\|_2^2)^{-1}$ is a standard discretization weight. This loss penalizes spatial discontinuities *within* semantic parts and prevents non-physical jumps in the object.

Contrastive Regularization. Beyond the within-part continuity enforced by L_{smooth}, we add a physically motivated contrast to keep parts separable near interfaces. From (E_i, ν_i) we compute the shear modulus $\mu_i = \frac{E_i}{2(1+\nu_i)}$ and the bulk modulus $K_i = \frac{E_i}{3(1-2\nu_i)}$. We form a log-domain, ℓ_2-normalized embedding

$$e_i = \text{norm}\big([\log \mu_i,\ \log K_i]\big), \tag{9}$$

where $\text{norm}(\cdot)$ denotes ℓ_2 normalization. For an anchor i, we choose a positive $p \in P(i)$ and a negative $n \in N(i)$, where $P(i) = \{j \mid \text{part}(j) = \text{part}(i)\}$ and $N(i) = \{j \mid \text{part}(j) \neq \text{part}(i)\}$, and minimize the triplet loss

$$L_{\text{con}} = \frac{1}{|\mathcal{T}|} \sum_{(i,p,n)\in\mathcal{T}} \max\big(0,\ \|e_i - e_p\|_2^2 - \|e_i - e_n\|_2^2 + m\big), \tag{10}$$

where $\mathcal{T}$ is the set of sampled triplets and $m > 0$ is the margin. Intuitively, L_{smooth} enforces within-part propagation consistency, while L_{con} preserves inter-part separability in elastic response.

Prompt–Part Assignment. Soft assignment provides an interpretable point-to-prompt distribution. Let s_{ik} denote the similarity logit between point i and prompt $k \in \{1, \ldots, K\}$, and define $a_i = \text{softmax}_k(s_{ik}) \in \Delta^{K-1}$ as the assignment distribution. We encourage this distribution to match the ground-truth part label via cross-entropy:

$$L_{\text{assign}} = \frac{1}{N} \sum_{i=1}^{N} \text{CE}\big(a_i,\ \pi(\text{part}(i))\big), \tag{11}$$

where $\pi(\cdot)$ maps a part label to its prompt index and $\text{CE}(a_i, \pi) = -\log a_{i,\pi}$. This couples the language interface to geometry, improving interpretability and editability without altering the task-loss pathway.

Overall Objective. The overall loss is a weighted sum of the above terms:

$$L = L_{\text{task}} + \lambda_{\text{smooth}} L_{\text{smooth}} + \lambda_{\text{con}} L_{\text{con}} + \lambda_{\text{assign}} L_{\text{assign}}, \tag{12}$$

where $(\lambda_{\text{smooth}}, \lambda_{\text{con}}, \lambda_{\text{assign}})$ are tuned on a validation set.

3.3 Physics-Editable Video Generation

Previous methods [25, 41] rely on manual initialization and only generate dynamics under initial conditions; as a result, the generated dynamics are short in duration and lack subsequent process control, which limits the diversity. Physical simulation of PhysChoreo enhances the controllability and diversity of dynamics by introducing a physics-editable dynamic simulation. Specifically, the simulator executes predefined, time-indexed action sequences during runtime, enabling

complex effects without restarting the simulation. Afterward, the dynamic trajectories are fed to a pre-trained video model to generate.

Physics-Editable Dynamics. We introduce an editable guide for MPM [15] and Rigid [32] simulations that supports temporal control. Specifically, we maintain the physical properties of each object in the scene. The corresponding properties, including constitutive parameters like Young's modulus and density, external force fields like gravity and wind, and object momentum like velocity, can be controlled individually. To maintain physical plausibility under non-steady states, we apply continuity constraints to transitions and limit extreme values. For object/part j at simulation step n, we smooth edited parameters $\boldsymbol{\theta}_j^n = (E_j^n, \nu_j^n, \rho_j^n)$ toward the user target $\tilde{\boldsymbol{\theta}}_j^n$ by

$$\boldsymbol{\theta}_j^{n+1} = (1-\gamma_n)\boldsymbol{\theta}_j^n + \gamma_n \tilde{\boldsymbol{\theta}}_j^n, \quad \gamma_n = 1 - e^{-\Delta t/\tau_j}. \tag{13}$$

where Δt is the solver step, τ_j is the transition time constant, and γ_n is the smoothing weight. We further bound stiffness by the MPM wave speed $c_j^2 = (\lambda_j + 2\mu_j)/\rho_j$, where $\lambda_j = \frac{E_j \nu_j}{(1+\nu_j)(1-2\nu_j)}$ and $\mu_j = \frac{E_j}{2(1+\nu_j)}$, using

$$s = \min\left(1, \frac{c_{\max}^2}{\max_j c_j^2 + \epsilon}\right), \quad E_j' = sE_j. \tag{14}$$

Thus edits are temporally continuous, and the shared scale s preserves relative stiffness. This makes simulation both flexible and stable, achieving fine-grained control without resetting the scene. Through this temporal control, we can create diverse yet physically realistic visual effects. For example, by eliminating the density of internal particles, we can achieve hollow or deflating; by controlling the force field on different objects, we can achieve counter-intuitive motion or bullet time; and by changing the material model, we can achieve transformation. All dynamic behaviors can be applied temporally, providing precision, controllability, and diversity for dynamics.

Video Generation. Given the input image and text prompt, we first use Dust3r [45] to estimate the positions and scales of all the objects in the scene, initializing the spatial information via point cloud reconstruction and part-level physics prediction. Considering that the reconstructed points primarily lie on the surface of objects, volumetric completion is required to obtain a solid representation for physical simulation. To this end, we employ a surface-to-interior propagation algorithm to generate particles for filling the interior of objects, which ensures seamless transitions from surface to interior. To assign the physical properties for filled points, we identify each interior particle's nearest surface particle using a k-nearest neighbors search and directly inherit the surface particle's properties to it. This process ensures seamless material transitions from object boundaries to interior volume during simulation. Finally, we use physics-editable simulators to produce point cloud motion trajectories, which are fed to a pre-trained video model [43] as conditions to generate videos with physical realism.

4 Dataset

We introduce the dataset used to train and evaluate our prediction model. It is one of the largest point cloud datasets that couples part segmentation with global descriptions, and part-level physical properties with textual annotations.

4.1 Dataset Pipeline

Our pipeline can be divided into three steps. First, we clean the labels of the segmented 3D data, including merging labels of the same object that are over-segmented and deleting labels that will not be used for part descriptions. We input the rendering of the object and the object's category name into GPT-5 [39] for reasoning, guiding it to generate an overall object description with vague physical descriptions within physically reasonable ranges based on geometric features. Then, we feed the rendering and name of each part along with the overall object description into the VLM for further reasoning, generating textual descriptions and physical properties for each part. Finally, we refine annotations by manually defining constraints between materials and physical properties, where GLM-4.5 [8] is employed to calibrate the correspondence between materials and physical properties. We combine the reasoning output of both VLM and LLM, ensuring the correctness of annotations, while manually verifying 9% of samples to guarantee quality.

4.2 Dataset Overview

We collect **9,580** samples that span 24 semantic categories with segmentation information from PartNet [31]. Each sample has an overall description, part-level physical properties for each part, including the real material, Young's modulus, density, and Poisson's ratio. Besides, to make the data more adaptable to simulation, we provide the mapped simulator material tag that corresponds to the initial real material description. Moreover, to increase the difficulty of training and evaluation, we deliberately introduce "counterfactual" labels for 5% of the examples (e.g., a gelatinous blade, a metallic flower), thereby promoting the model's ability to learn the associations between text and physics.

5 Experiments

In this section, we conduct extensive experimental comparisons on both the prediction of physical properties and the generation of physics-controllable videos. Ablation studies are further conducted to corroborate the effectiveness of our newly proposed modules.

5.1 Implementation Details.

We use GPT-5 [39] reasoning to split text into a global description, part prompts, solver instructions, as well as an input text prompt. For instance reconstruction, we use Grounded-Sam [17, 26, 36] to segment instances from the image and use InstantMesh [51] to generate corresponding meshes. A pre-trained part segmentation encoder PartField [23] is then adopted to obtain 96-dim part-level semantics, concatenated with 96-dim embedding from a MLP, is conditioned with frozen CLIP [35] 256-dim text features from both global description and part prompts; soft assignment is computed with $\tau = 0.07$; the concatenated features are mapped to 512 dimensions, then pass through 8-head cross-attention and a 6-layer, 8-head transformer encoder. We predict continuous physical quantities and 6 material models for simulation. The model is trained on 2 A6000 GPUs, with a learning rate of 3×10^{-4}, batch size of 32, AdamW [29], cosine decay [28], respective weights $\lambda_{\text{reg}} = 1, \lambda_{\text{cls}} = 0.3, \lambda_{\text{assign}} = 0.1, \lambda_{\text{smooth}} = 0.02, \lambda_{\text{con}} = 5 \times 10^{-4}$, and patience-based early stopping on validation with $P = 10$, $\Delta = 10^{-4}$, checkpoint retained with training halts at epoch 41. For video generation, we use Taichi-based [12] simulation and a pre-trained image-to-video model Wan2.2-Fun-5B-Control [43]. PhysChoreo can be deployed on a single RTX 5090 GPU. The physics reconstruction and pose estimation stages take around 120 seconds, and the simulation takes about 30 seconds. We choose GPT-5 [39] for multiple rounds of instruction optimization with complex dynamics to achieve better results. Please refer to Suppl. for details of prompts, dataset, training particulars, and solvers.

5.2 Physical Property Prediction

Baselines. We compare the predictor of PhysChoreo against a recently proposed multi-view prediction method Pixie [18] and two large vision-language model methods NeRF2Physics [54] and PUGS [38]. Pixie renders the multiple views from objects, adds CLIP [35] features to voxels on that basis, and maps the feature field to physical properties through a U-Net [37]. NeRF2Physics and PUGS render the object from multiple views, then feed the images into a VLM to infer possible physical properties.

Evaluation Metric. We apply a logarithmic transformation to E and ρ on the obtained results, i.e., we evaluate the numerical errors of $[\log E, \nu, \log \rho]$, as well as the material model prediction accuracy. For fairness, all methods are evaluated in the same six materials, since material types are used for simulation. For LLM-based baselines, NeRF2Physics [54] and [38], we add the simulation requirement and six categories to the prompt and restrict their outputs. Since Pixie [18] already predicts five overlapping simulation materials, to avoid disadvantaging Pixie, we adopt a conservative upper-bound mapping: if GT is plastic, we additionally accept its closest available materials, elastic/metal/rigid, because Pixie does not contain plastic. For evaluation, we randomly select 100 samples from the test set that are not involved in the training and map the materials predicted by other methods to our material models.

Table 1: Quantitative comparison of material model prediction and physical property errors across baselines.

Method	Mat.Acc.↑	log$E$ err.↓	ν err.↓	logρ err.↓
NeRF2Physics	0.628	2.033	0.064	0.521
PUGS	0.283	2.778	0.076	0.627
Pixie	0.349	4.129	0.103	0.848
Ours	**0.789**	**0.661**	**0.061**	**0.249**

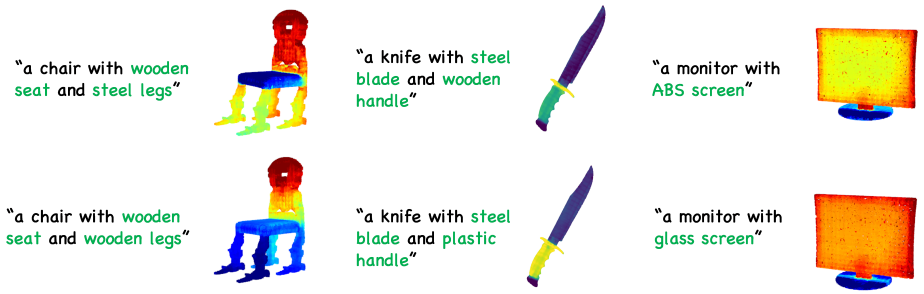

Fig. 4: Our model can achieve part-level physical property controllable prediction through text condition.

Results. Table 1 shows the comparison results between our method and baselines. Our method achieves the best performance on all metrics. Since other methods predict fixed material fields while we can learn text- and part-aware distribution, as shown in Fig. 4, our method can use text to control the physical properties of specific parts, providing preliminary controllability for subsequent simulation.

5.3 Image-to-Video Generation

Baselines. PhysChoreo uses the simulation results to guide video generation. To thoroughly assess the quality of generation, we compare our method with two physics-based methods, PhysGen3D [3] and PhysCtrl [44], two open-source generative video models, Wan2.2-5B [43] and CogVideoX-3 [53], and the state-of-the-art closed-source video model, Veo 3.1 [6].

Evaluation Metric. Inspired by VideoPhy [1], we evaluate 10 generated videos using three metrics: (1) Physical Commonsense (PC): whether the motion of the objects follows physically plausible deformations and dynamics; (2) Semantic Alignment (SA): the degree of match between the video content and motion and the text content, with a particular focus on the alignment of temporal instructions; (3) Visual Quality (VQ): the detailed visual quality of the video. Each metric is scored on a 5-point scale. Then, based on these metrics, Gemini-2.5-Pro [42] scores each video and 31 real users choose the preferred selection of each case.

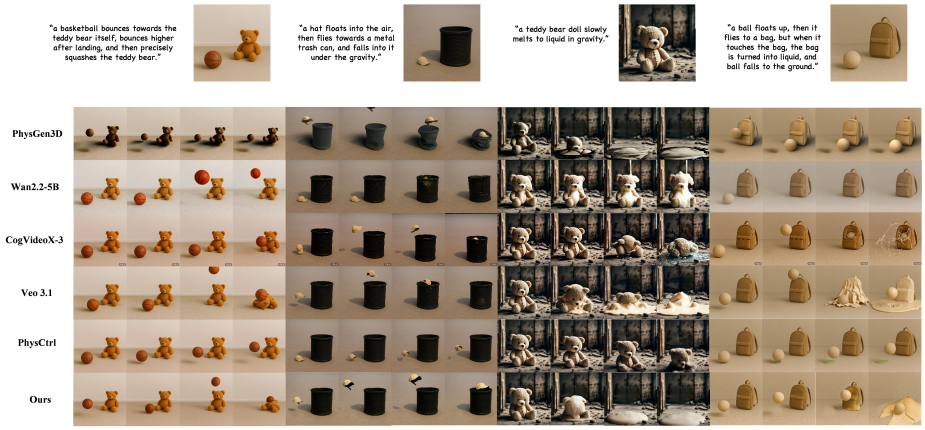

Fig. 5: Qualitative comparison between PhysChoreo and existing image-to-video generation models.

Table 2: User study results comparing videos from all methods(%).

Method	SA (↑)	PC (↑)	VQ (↑)	Total (↑)
PhysGen3D	7.7	13.0	6.7	10.42
PhysCtrl	8.5	17.0	15.2	14.1
Wan2.2-5B	3.0	4.0	6.6	4.6
CogVideoX-3	3.5	6.4	8.2	6.0
Veo 3.1	9.9	17.7	20.5	16.1
Ours	**67.4**	**41.9**	**42.8**	**50.1**

Table 3: Quantitative video comparison under VLM evaluation.

Method	SA (↑)	PC (↑)	VQ (↑)	AVG (↑)
PhysGen3D	2.30	2.10	3.50	2.63
PhysCtrl	3.75	4.25	4.30	4.1
Wan2.2-5B	1.75	1.70	4.20	2.55
CogVideoX-3	2.40	2.55	4.15	3.04
Veo 3.1	4.10	4.20	**4.90**	4.40
Ours	**4.70**	**4.55**	4.75	**4.67**

Results. We report the VLM scores in Table 3 and 642 valid selection results in Table 2. Thanks to physics-based simulation, our generated videos achieve the best PC. Since we explicitly predict the dynamics of objects, PhysChoreo achieves a significant lead on SA. While the visual quality of our generated videos is not as good as Veo3.1's, our method achieves the best average score. Qualitative results in Fig. 5 are basically consistent with our evaluation scores. Furthermore, we demonstrate in Fig. 6 that, by manipulating the physical properties in the scene, PhysChoreo can generate various behaviors like liquefying upon collision, collapsing upon landing, and counter-intuitive bounces.

5.4 Ablation Studies

Model Design. To evaluate the effectiveness of our modules, we conduct ablation experiments in Table 4(iteration denotes the early stopping epoch). Our baseline is to directly predict physical properties through a transformer model. Based on this, we gradually incorporate cross-attention, soft assignment, and segmentation priors. In (a), the error is too large to accurately predict the physical properties. In (b) and (c), text features significantly improve performance.

Fig. 6: PhysChoreo can generate physically realistic and visually appealing videos by changing various physical properties during the runtime.

Table 4: Ablation study on different components of our framework.

Method	Cross Attn.	Dual Stage.	Assignment	Seg. Prior	Mat. Acc. (↑)	Total Err. (↓)	Iter. (↓)
(a)					0.5507	3.4230	**10**
(b)	✓				0.7393	0.9789	35
(c)	✓	✓			0.8077	0.5801	36
(d)	✓	✓	✓		0.8372	0.5046	26
Ours	✓	✓	✓	✓	**0.8605**	**0.3318**	41

In (d), soft assignment that explicitly enhances the injection of text features, improves efficiency. Finally, the segmentation prior increases the training time for the added dimensions but gets the best performance.

Physics-based Supervision. We conduct ablation experiments on loss functions. In Table 5, without text-part assignment supervision $\mathcal{L}_{\text{assign}}$, performance decreases slightly, but iterations increase markedly. Without $\mathcal{L}_{\text{smooth}}$, performance declines slightly for the high-frequency noise within the parts' attributes. Removing contrast supervision $\mathcal{L}_{\text{con}}$ achieves a similar error score, but material accuracy decreases. This is reasonable because smooth supervision improves the prediction of boundary points.

Part-level Knowledge. To assess the effectiveness of part-level knowledge within PhysChoreo, we conduct several experiments and add comparison videos with manually specified physics. These oracle global settings share the same simulator and video pipeline, but miss part-specific and temporally edited dynamics. As shown in Fig. 7, when directly using global properties instead of part-level prediction, the results fail to maintain correct part-level dynamics. Finally, with part-level properties predicted by PhysChoreo, our results exhibit more distinct part-level responses.

5.5 Failure case

Common failure cases occur in the rendering stage. Our simulation ensures stable physical trajectories, but current video models occasionally fail to handle correct light and shadow details in dynamic motions, leading to visible artifacts in fine-grained appearance, which is shown in Fig. 7.

Table 5: Ablation study on our loss components.

Method	Mat. Acc. (↑)	Total Err. (↓)	Iter. (↓)
w/o $\mathcal{L}_{\text{assign}}$	0.8534	0.3753	50
w/o $\mathcal{L}_{\text{smooth}}$	0.8578	0.3579	**37**
w/o $\mathcal{L}_{\text{con}}$	0.8310	0.3451	38
Full method	**0.8605**	**0.3318**	41

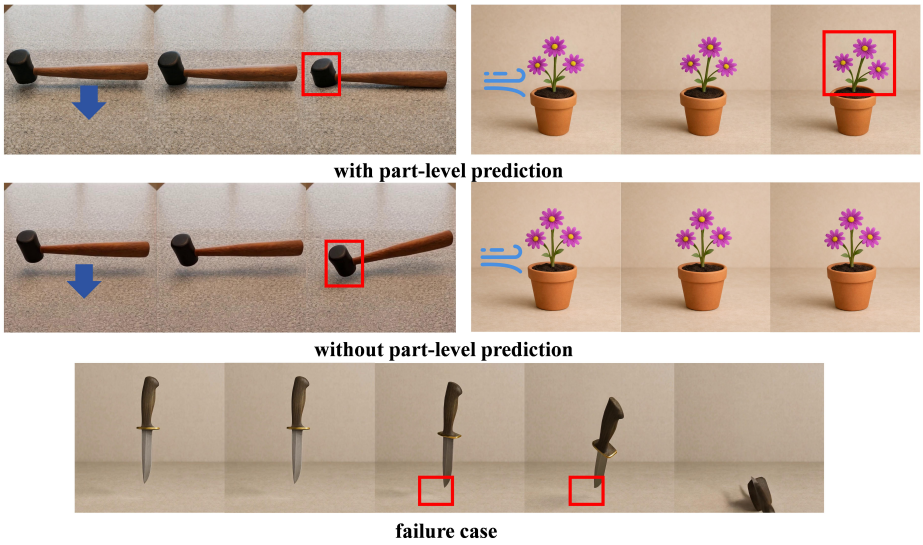

Fig. 7: Qualitative results of part-level ablation and failure case.

6 Conclusion

We introduce PhysChoreo, demonstrating excellent performance in reconstructing physical properties and generating rich dynamic videos, with the prediction model and solver also having broader applications such as robot simulation.

Limitation. Our method focuses on independent objects, thus still lacking adequacy for large-scale scenes. Additionally, despite adopting point sampling methods, the internal physical states cannot be precisely predicted. Future work includes addressing these toward broader scenarios.

Acknowledgements

This work is supported in part by the National Natural Science Foundation of China (NSFC) under Grant No. 62371164.

References

1. Bansal, H., Lin, Z., Xie, T., Zong, Z., Yarom, M., Bitton, Y., Jiang, C., Sun, Y., Chang, K.W., Grover, A.: Videophy: Evaluating physical commonsense for video generation. arXiv preprint arXiv:2406.03520 (2024)
2. Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al.: Stable video diffusion: Scaling latent video diffusion models to large datasets. arXiv preprint arXiv:2311.15127 (2023)
3. Chen, B., Jiang, H., Liu, S., Gupta, S., Li, Y., Zhao, H., Wang, S.: Physgen3d: Crafting a miniature interactive world from a single image. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 6178–6189 (2025)
4. Chen, H., Xia, M., He, Y., Zhang, Y., Cun, X., Yang, S., Xing, J., Liu, Y., Chen, Q., Wang, X., et al.: Videocrafter1: Open diffusion models for high-quality video generation. arXiv preprint arXiv:2310.19512 (2023)
5. Chen, H., Zhang, Y., Cun, X., Xia, M., Wang, X., Weng, C., Shan, Y.: Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7310–7320 (2024)
6. DeepMind, G.: Veo: a text-to-video generation system. Tech. Rep. Tech. Report, Google LLC (2025), `https://storage.googleapis.com/deepmind-media/veo/Veo-3-Tech-Report.pdf`, version 3
7. Gillman, N., Herrmann, C., Freeman, M., Aggarwal, D., Luo, E., Sun, D., Sun, C.: Force prompting: Video generation models can learn and generalize physics-based control signals. Advances in Neural Information Processing Systems **38**, 103174–103201 (2026)
8. GLM, T., Zeng, A., Xu, B., Wang, B., Zhang, C., Yin, D., Zhang, D., Rojas, D., Feng, G., Zhao, H., et al.: Chatglm: A family of large language models from glm-130b to glm-4 all tools. arXiv preprint arXiv:2406.12793 (2024)
9. Gu, Z., Yan, R., Lu, J., Li, P., Dou, Z., Si, C., Dong, Z., Liu, Q., Lin, C., Liu, Z., et al.: Diffusion as shader: 3d-aware video diffusion for versatile video generation control. In: Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers. pp. 1–12 (2025)
10. He, Y., Yang, T., Zhang, Y., Shan, Y., Chen, Q.: Latent video diffusion models for high-fidelity long video generation. arXiv preprint arXiv:2211.13221 (2022)
11. Hu, Y., Anderson, L., Li, T.M., Sun, Q., Carr, N., Ragan-Kelley, J., Durand, F.: Difftaichi: Differentiable programming for physical simulation. arXiv preprint arXiv:1910.00935 (2019)
12. Hu, Y., Li, T.M., Anderson, L., Ragan-Kelley, J., Durand, F.: Taichi: a language for high-performance computation on spatially sparse data structures. ACM Transactions on Graphics (TOG) **38**(6), 1–16 (2019)
13. Hu, Y., Liu, J., Yang, X., Xu, M., Kuang, Y., Xu, W., Dai, Q., Freeman, W.T., Durand, F.: Quantaichi: a compiler for quantized simulations. ACM Transactions on Graphics (TOG) **40**(4), 1–16 (2021)
14. Huang, T., Zhang, H., Zeng, Y., Zhang, Z., Li, H., Zuo, W., Lau, R.W.: Dreamphysics: Learning physics-based 3d dynamics with video diffusion priors. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 3733–3741 (2025)
15. Jiang, C., Schroeder, C., Teran, J., Stomakhin, A., Selle, A.: The material point method for simulating continuum materials. In: Acm siggraph 2016 courses, pp. 1–52 (2016)

16. Kang, B., Yue, Y., Lu, R., Lin, Z., Zhao, Y., Wang, K., Huang, G., Feng, J.: How far is video generation from world model: A physical law perspective. arXiv preprint arXiv:2411.02385 (2024)
17. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4015–4026 (2023)
18. Le, L., Lucas, R., Wang, C., Chen, C., Jayaraman, D., Eaton, E., Liu, L.: Pixie: Fast and generalizable supervised learning of 3d physics from pixels. arXiv preprint arXiv:2508.17437 (2025)
19. Li, Q., Xing, Z., Wang, R., Zhang, H., Dai, Q., Wu, Z.: Magicmotion: Controllable video generation with dense-to-sparse trajectory guidance. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12112–12123 (2025)
20. Li, Z., Yu, H.X., Liu, W., Yang, Y., Herrmann, C., Wetzstein, G., Wu, J.: Wonderplay: Dynamic 3d scene generation from a single image and actions. In: Proceedings of the IEEE/CVF international conference on computer vision (2025)
21. Liang, J., Fan, Y., Zhang, K., Timofte, R., Van Gool, L., Ranjan, R.: Movideo: Motion-aware video generation with diffusion model. In: European conference on computer vision. pp. 56–74. Springer (2024)
22. Lin, Y., Lin, C., Xu, J., Mu, Y.: Omniphysgs: 3d constitutive gaussians for general physics-based dynamics generation. arXiv preprint arXiv:2501.18982 (2025)
23. Liu, M., Uy, M.A., Xiang, D., Su, H., Fidler, S., Sharp, N., Gao, J.: Partfield: Learning 3d feature fields for part segmentation and beyond. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9704–9715 (2025)
24. Liu, M., Zhu, Y., Cai, H., Han, S., Ling, Z., Porikli, F., Su, H.: Partslip: Low-shot part segmentation for 3d point clouds via pretrained image-language models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 21736–21746 (2023)
25. Liu, S., Ren, Z., Gupta, S., Wang, S.: Physgen: Rigid-body physics-grounded image-to-video generation. In: European Conference on Computer Vision. pp. 360–378. Springer (2024)
26. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C., Yang, J., Su, H., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In: European conference on computer vision. pp. 38–55. Springer (2024)
27. Liu, Z., Ye, W., Luximon, Y., Wan, P., Zhang, D.: Unleashing the potential of multi-modal foundation models and video diffusion for 4d dynamic physical scene simulation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 11016–11025 (2025)
28. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983 (2016)
29. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
30. Mittal, H., Zhuang, P., Lee, H.Y., Tulsiani, S.: Uniphy: Learning a unified constitutive model for inverse physics simulation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 16208–16218 (2025)
31. Mo, K., Zhu, S., Chang, A.X., Yi, L., Tripathi, S., Guibas, L.J., Su, H.: Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 909–918 (2019)

32. Müller, M., Macklin, M., Chentanez, N., Jeschke, S., Kim, T.Y.: Detailed rigid body simulation with extended position based dynamics. In: Computer Graphics Forum. vol. 39, pp. 101–112. Wiley Online Library (2020)
33. Ni, H., Shi, C., Li, K., Huang, S.X., Min, M.R.: Conditional image-to-video generation with latent flow diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18444–18455 (2023)
34. Png, M., Huang, M., Bahreman, M., Kube, C.M., Lowe, M.J., Lan, B.: Accurate wave velocity measurement from diffuse wave fields. NDT & E International **156**, 103431 (2025)
35. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021)
36. Ren, T., Liu, S., Zeng, A., Lin, J., Li, K., Cao, H., Chen, J., Huang, X., Chen, Y., Yan, F., et al.: Grounded sam: Assembling open-world models for diverse visual tasks. arxiv 2024. arXiv preprint arXiv:2401.14159
37. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)
38. Shuai, Y., Yu, R., Chen, Y., Jiang, Z., Song, X., Wang, N., Zheng, J., Ma, J., Yang, M., Wang, Z., et al.: Pugs: Zero-shot physical understanding with gaussian splatting. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 4478–4485. IEEE (2025)
39. Singh, A., Fry, A., Perelman, A., Tart, A., Ganesh, A., El-Kishky, A., McLaughlin, A., Low, A., Ostrow, A., Ananthram, A., et al.: Openai gpt-5 system card. arXiv preprint arXiv:2601.03267 (2025)
40. Su, H., Huang, T., Wan, Z., Wu, X., Zuo, W.: S^2am3d: Scale-controllable part segmentation of 3d point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14357–14366 (2026)
41. Tan, X., Jiang, Y., Li, X., Zong, Z., Xie, T., Yang, Y., Jiang, C.: Physmotion: Physics-grounded dynamics from a single image. arXiv preprint arXiv:2411.17189 (2024)
42. Team, G., Anil, R., Borgeaud, S., Alayrac, J.B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A.M., Hauth, A., Millican, K., et al.: Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805 (2023)
43. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
44. Wang, C., Chen, C., Huang, Y., Dou, Z., Liu, Y., Gu, J., Liu, L.: Physctrl: Generative physics for controllable and physics-grounded video generation. arXiv preprint arXiv:2509.20358 (2025)
45. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3r: Geometric 3d vision made easy. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20697–20709 (2024)
46. Wang, Y., Chen, X., Ma, X., Zhou, S., Huang, Z., Wang, Y., Yang, C., He, Y., Yu, J., Yang, P., et al.: Lavie: High-quality video generation with cascaded latent diffusion models. International Journal of Computer Vision **133**(5), 3059–3078 (2025)
47. Wiedemer, T., Li, Y., Vicol, P., Gu, S.S., Matarese, N., Swersky, K., Kim, B., Jaini, P., Geirhos, R.: Video models are zero-shot learners and reasoners. arXiv preprint arXiv:2509.20328 (2025)

48. Xie, T., Zhao, Y., Jiang, Y., Jiang, C.: Physanimator: Physics-guided generative cartoon animation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 10793–10804 (2025)
49. Xie, T., Zong, Z., Qiu, Y., Li, X., Feng, Y., Yang, Y., Jiang, C.: Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4389–4398 (2024)
50. Xing, J., Xia, M., Zhang, Y., Chen, H., Yu, W., Liu, H., Liu, G., Wang, X., Shan, Y., Wong, T.T.: Dynamicrafter: Animating open-domain images with video diffusion priors. In: European Conference on Computer Vision. pp. 399–417. Springer (2024)
51. Xu, J., Cheng, W., Gao, Y., Wang, X., Gao, S., Shan, Y.: Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. arXiv preprint arXiv:2404.07191 (2024)
52. Yang, Y., Huang, Y., Guo, Y.C., Lu, L., Wu, X., Lam, E.Y., Cao, Y.P., Liu, X.: Sampart3d: Segment any part in 3d objects. arXiv preprint arXiv:2411.07184 (2024)
53. Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., et al.: Cogvideox: Text-to-video diffusion models with an expert transformer. arXiv preprint arXiv:2408.06072 (2024)
54. Zhai, A.J., Shen, Y., Chen, E.Y., Wang, G.X., Wang, X., Wang, S., Guan, K., Wang, S.: Physical property understanding from language-embedded feature fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 28296–28305 (2024)
55. Zhang, T., Yu, H.X., Wu, R., Feng, B.Y., Zheng, C., Snavely, N., Wu, J., Freeman, W.T.: PhysDreamer: Physics-based interaction with 3d objects via video generation. In: European Conference on Computer Vision. Springer (2024)
56. Zhao, H., Jiang, L., Jia, J., Torr, P.H., Koltun, V.: Point transformer. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 16259–16268 (2021)