NeuROK: Generative 4D Neural Object Kinematics

Chen Geng[1,*] Guangzhao He[3,*] Yue Gao[1,*]
Yunzhi Zhang[1] Shangzhe Wu[2] Jiajun Wu[1]

[1]Stanford University [2]University of Cambridge [3]Cornell University

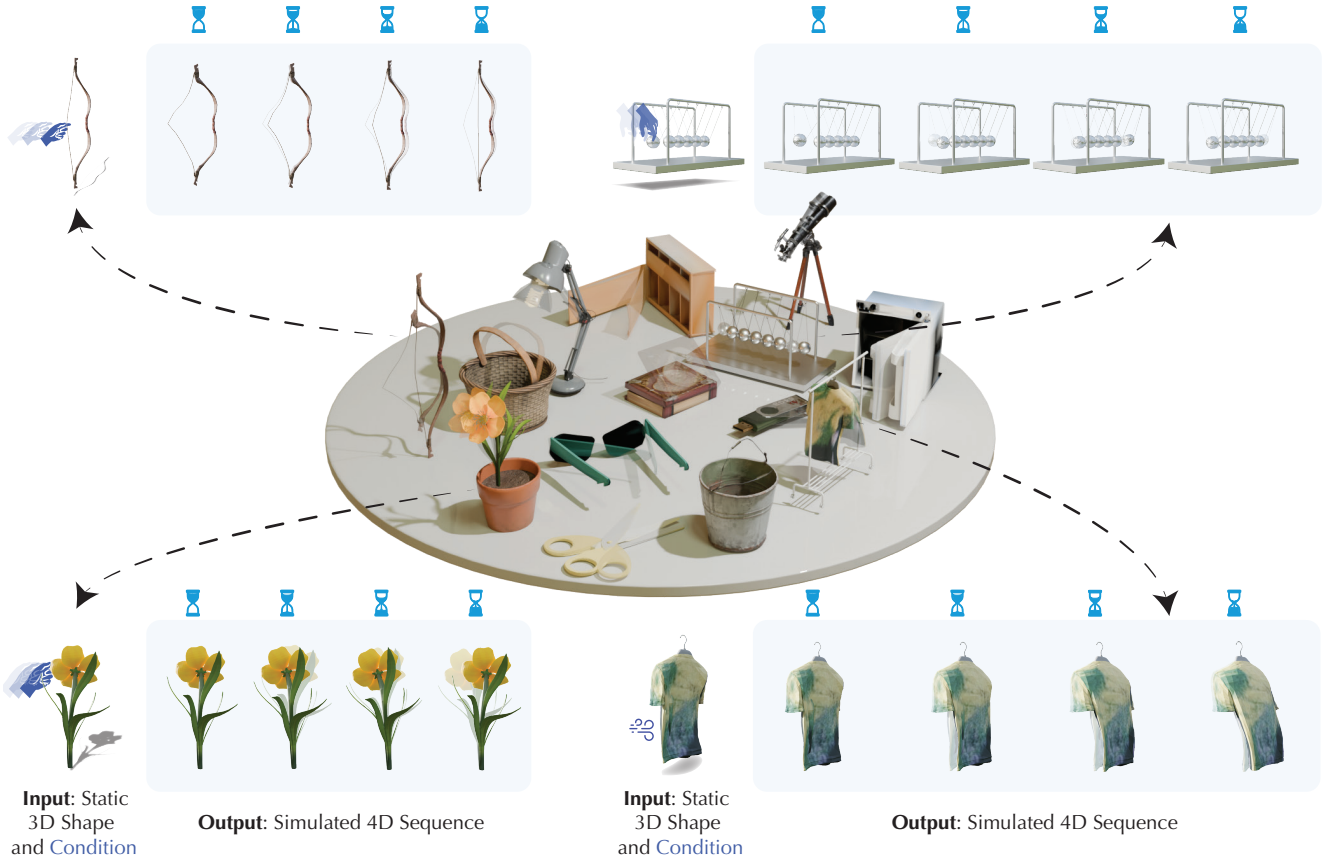


Figure 1. We present a versatile and scalable framework for generating simulative 4D dynamics of static 3D objects under physical conditions (*e.g.*, forces, actions, velocities). Trained on a large-scale 4D shape dataset without any explicit physical annotations, our method does not rely on any inductive bias of the object's dynamic structure and therefore can be applied to various types of dynamic objects, ranging from elastic bodies, cloth, and continuum bodies, to multi-body objects. Project page: https://chen-geng.com/neurok

Abstract

Data-driven approaches have revolutionized 3D vision, enabling transformers to effectively reconstruct and generate static 3D objects. However, generating simulative 4D dynamics—realistic temporal deformations of static objects under various physical conditions—remains challenging and often ad hoc, despite its importance in building comprehensive 3D world models. Most existing methods assume a predefined physical model and use system identification to estimate parameters, restricting these methods to specific categories and small-scale datasets.

We propose that these restrictions can be overcome by learning a data-driven kinematic state parameterization for object-centric physical systems. Specifically, we learn both a latent space representing all possible states of the object and a decoder that maps any sampled latent to a plausibly deformed shape of the object. We refer to this parameterization as ***Neur****al* ***O****bject* ***K****inematics* *(*NeuROK*), and learn a transformer-based encoder-decoder model on a curated large-scale 4D dataset. This formulation and the learned model significantly simplify the generation of simulative dy-*

namics since we only need to consider the dynamics within a low-dimensional latent space from the Lagrangian mechanics' perspective in classical physics. We demonstrate the effectiveness and generality of this neural simulation framework across diverse dynamic object types, showing clear advantages over prior works.

1. Introduction

> These quantities need not be the Cartesian co-ordinates of the particles, and the conditions of the problem may render some other choice of coordinates more convenient.
>
> — *L. Landau & E. Lifshitz, in Mechanics, 1960*

Given a 3D geometric snapshot of a dynamic object, humans can intuitively imagine how the object would react under different physical conditions, even without precise knowledge of the governing physical equations. However, in the community of generative AI, generating such 4D reactive behaviors with no reliance on any category-specific physical priors is far from trivial, despite the importance of this capability in constructing 3D world models for embodied AI or robotics [52, 80].

A long-standing view holds that generating such 4D simulative dynamics demands a comprehensive physical understanding of the object. This is epitomized by most existing works [105, 114] that generate 4D dynamics by adopting predefined category-specific physical models and estimating their parameters with system identification. While this paradigm is effective for target object categories (*e.g.*, articulated objects, continuum bodies, and cloth), it struggles to generalize beyond these predefined categories, and, more importantly, offers limited scalability to large-scale 4D datasets comprising diverse dynamic structures.

Is it possible to build a general-purpose simulator that generates such 4D motions without any category-specific inductive bias? We argue that this is achievable by reconsidering a critical yet long-overlooked piece: the **kinematic state parameterization** of dynamic objects. As illustrated in Fig. 2, a kinematic state parameterization defines the configuration space of state vectors that fully specify an object's geometry. Most existing approaches [69, 105, 114] adopt a kinematic state parameterization naturally inherited from the object's shape representation, *e.g.*, a dense particle set derived from mesh discretizations. While effective, this choice leads to an over-parameterized system, and thus necessitates category-specific physical constraints to prevent the system from being under-determined.

We revisit this important factor by introducing an automatically-discovered kinematic state parameterization scheme — **Neur**al **O**bject **K**inematics (NEUROK) — a latent space from which any vector sampled can be decoded into a plausible deformation of the modeled object. With this learned parameterization, the physical system can be greatly simplified: we only need to model the transition between low-dimensional latent vectors, similar to how a pendulum system can be simplified through a symbolic parameterization (Fig. 2(a)). This data-driven parameterization leads to a universal framework that simulates system dynamics from the perspective of Lagrangian mechanics [47], where category-agnostic energy functions are defined over the latent states and dynamics are directly derived using Euler-Lagrange equations.

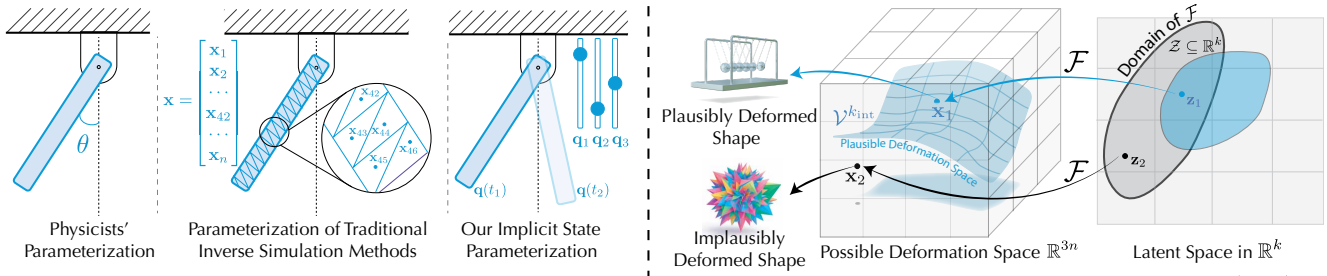
This framework forms a versatile and scalable pipeline for generative simulation of dynamic objects. Its core learning component, NEUROK, adopts a transformer-based [98] encoder-decoder architecture that learns to encode a static 3D object into a latent distribution over its possible kinematic states and to decode any sampled latent vector into a corresponding deformation field. The model can be trained solely on 4D geometric trajectories of 3D objects, eliminating any need for physical or action annotations. Moreover, this framework relies on a minimal inductive bias — that the object's deformation space is low-dimensional — making it broadly applicable to diverse dynamic objects.

We validate our framework by curating a large-scale 4D object dataset, training a feed-forward NEUROK model, and generating 4D dynamics across a wide range of objects. We evaluate its performance by comparing against existing methods, demonstrating its superior generalizability and effectiveness. To the best of our knowledge, this is the first data-driven framework capable of simulating object-centric physical systems without any reliance on heuristic priors or physical annotations.

2. Related Work

Physically-Inspired 4D Generation. Existing approaches to generating 4D simulative dynamics typically follow a two-step paradigm: finding a physical model of the targeted domain, and determining its parameters with system identification. This includes directly modeling physical properties of rigid objects [73, 107, 112]; modeling elastic objects with MPM [12, 15, 24, 41, 44, 55, 56, 65, 67–69, 81, 105, 114, 115], projective dynamics [10, 26, 88], or geometry-agnostic elastic simulation methods [16, 29, 82]; using spring-mass to model deformable objects [42, 118]; predicting articulations to model articulated objects [21, 43, 45, 48, 51, 70, 71, 79, 83, 103, 104, 106]; and building physical models for cloth [33, 57, 59, 62, 66, 77, 117]. While they perform well within specific domains, none can generate 4D motions without assuming a predefined dynamic structure. Our framework removes such structural biases, enabling general 4D simulative dynamics generation.

Reduced-Order Simulation. Model reduction is a common technique in forward computer graphics, yet the focus



(a) Different Kinematic Parameterizations of the Same Physical System

(b) Illustration of an Abstract Kinematic State Parameterization $(\mathcal{Z}, \mathcal{F})$

Figure 2. **Kinematic state parameterization. (a)** Several kinematic state parameterizations can be used to describe a physical system. The symbolic parameterizations used in classical mechanics are concise yet not accessible in inverse problems. Traditional inverse simulation approaches use geometry-derived parameterizations, yet require dense physical constraints to solve the over-parameterized system. We instead learn low-dimensional parameterizations that are both compact and learnable from data. **(b)** As formally defined in Def. 1, a kinematic state parameterization studied in this paper is a pair $(\mathcal{Z}, \mathcal{F})$ which contains a latent manifold $\mathcal{Z}$ and a decoder $\mathcal{F}$ that maps a sampled latent to a vertex configuration. This definition explicitly includes those kinematic state parameterizations that are not compact.

is *efficiency* rather than versatility. The goal of such approaches [5, 14, 19, 20, 31, 39, 40, 50, 63, 82, 91, 95, 97, 99, 100, 120] is typically to accelerate an existing physical simulation system where all physical constraints are known, in contrast to our category-agnostic setting. These approaches typically train instance-specific neural networks to represent the reduced-order kinematic space for a specific object, rather than learning a generalizable, amortized-inference model on a large dataset as in our framework.

Machine Learning for Dynamic Systems. Beyond 3D vision, machine learning has also been used to model non-visual dynamic systems, typically through either physics-agnostic or physics-aware approaches. Physics-agnostic methods [3, 13, 53, 58, 60, 87, 93, 94, 96] learn dynamics end-to-end — often via GNNs — using synthetic datasets of action-state pairs. Although effective in controlled settings, they struggle to generalize to real-world objects due to the scarcity of action-labeled data. In contrast, our method relies solely on 4D geometric supervision, offering greater scalability for graphics and 3D vision applications. Physics-aware methods assume known physical models and use neural networks to solve PDEs [18, 61, 64, 90], learn constitutive laws [76, 81], and learn discretization schemes [2]. While demonstrating potential in producing accurate solutions, these approaches are unsuitable for our setting which makes no assumptions about dynamic structures. Closer to our formulation are methods that use neural networks [6, 23, 30, 75] to model systems within the Lagrangian mechanics framework, but their focus is learning the system's Lagrangian from synthetic data rather than learning data-driven kinematic state parameterizations.

Neural Deformation Priors. Several graphics systems have also explored learning data-driven priors over object deformations, but most are category-specific (*e.g.*, for humans [74, 86], faces [1, 8, 9], and animals [121]) and target other tasks — most dominantly, for animating characters [11, 32, 34, 35, 38, 78, 84, 85, 101, 108–111, 119] and controlling embodied agents [54, 72, 92]. We instead formalize this idea through the concept of kinematic state parameterization and demonstrate its huge potential as a general interface in physically-inspired 4D generation.

3. Overview

3.1. Formulation and Concepts

This paper studies generating simulative dynamics of 3D object-centric[1] physical systems. Our pipeline takes a static snapshot of a 3D dynamic object and a set of physical conditions (*e.g.,* actions, forces, initial velocities) as inputs, and generates a sequence of temporally evolving 3D shapes. As a single 3D snapshot of an object cannot fully determine its physical parameters, our goal is to *generate* one plausible 4D sequence that satisfies one valid physical configuration and conforms to human physical intuition [4]. We assume no kinematic or physical priors on the dynamic structure of the modeled object. The object can be articulated, rigid, a continuum body, or even a heterogeneous combination of several dynamic types, like the examples shown in Fig. 1.

The geometry of the modeled object is represented as a mesh $\mathcal{M}_0 = (V_0, F)$ with n vertices. We denote by $\mathbf{x}^0 \in \mathbb{R}^{3n}$ the concatenated vertex positions in V_0. Our pipeline outputs a sequence of deformed meshes with timestamps ranging from 1 to T, denoted as $\{\mathcal{M}_1, \cdots, \mathcal{M}_T\}$, where $\mathcal{M}_t = (V_t, F)$, and the concatenated vertex positions are represented by $\mathbf{x}^t \in \mathbb{R}^{3n}$.

While the vertices of the mesh $\mathcal{M}_0$ can theoretically take arbitrary positions in $\mathbb{R}^{3n}$, only a small subset of these configurations correspond to plausibly re-posed shapes. In fact, a randomly sampled deformation vector from $\mathbb{R}^{3n}$ will almost certainly yield a deformed mesh far outside the distribution of valid object poses. Empirically, the set of plausible vertex position vectors of a dynamic object forms a low-dimensional configuration manifold $\mathcal{V}^{k_{\text{int}}}$ embedded in $\mathbb{R}^{3n}$, where k_{int} denotes the intrinsic degrees of freedom of the deformation space and $k_{\text{int}} \ll 3n$.

[1] We colloquially define an object-centric physical system as one in which most motion arises from a single dominant deformable object.

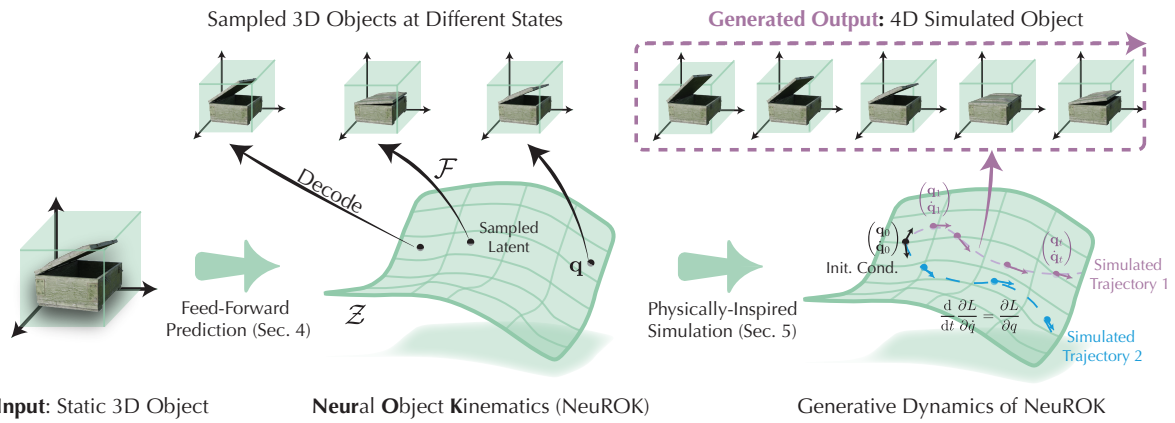


Figure 3. **Overview of our framework.** Given a static 3D shape, NeuROK uses a transformer-based encoder to predict an instance-specific latent space to represent different kinematic states of this object. Each sampled latent on the learned manifold can be decoded to a corresponding state of the input object. Under different physical conditions (*e.g.*, forces, actions, velocities), our method generates dynamic trajectories of latents by solving a physically-inspired ODE.

When studying these object-centric physical systems containing a deformable mesh with n vertices, we need to define a parameterization scheme for its kinematic states, which in turn determines the solution space for a physical simulator. We formulate this with the following definition:

Definition 1: Kinematic State Parameterization

A k-dimensional *kinematic state parameterization* for a dynamic object is a pair $(\mathcal{Z}, \mathcal{F})$, where $\mathcal{Z} \subseteq \mathbb{R}^k$ is the state space of the parameterization, and $\mathcal{F} : \mathcal{Z} \to \mathbb{R}^{3n}$ maps any state vector $\mathbf{z} \in \mathcal{Z}$ to a vertex configuration of $\mathcal{M}_0$ with n vertices.

Determining a kinematic state parameterization is the first step when studying a physical system, and it dictates how the system should be solved. As in Fig. 2(a), concise symbolic parameterizations are commonly used to simplify the solution space, but such representations are generally inaccessible in 4D generation where only the raw 3D geometry $\mathcal{M}_0$ is given. Consequently, most approaches adopt geometry-derived parameterizations, such as the high-dimensional particles (material points) used in MPM [41]. Such parameterizations are commonly redundant and under-constrained since some configurations will yield implausibly deformed shapes, as in Fig. 2(b).

To solve dynamics in high-dimensional solution space defined by the redundant parameterization, prior works introduce category-specific physical equations and constraints to prevent the system from being under-determined. These formulations are effective in targeted domains, yet they struggle to model objects beyond the designated category.

3.2. Proposed Solution

We address the above-discussed problem by introducing a kinematic state parameterization learned from data:

Definition 2: Neural Object Kinematics

If a k-dimensional kinematic state parameterization $(\mathcal{Z}, \mathcal{F})$ uses a neural network to represent $\mathcal{F}$, and the range of $\mathcal{F}$ is $\mathcal{V}^{k_{\text{int}}}$, we refer to this pair as a **Neural Object Kinematics** (NeuROK) of a dynamic object.

We train an encoder-decoder model to infer NeuROK of a given object $\mathcal{M}_0$. The model comprises an encoder that encodes $\mathcal{M}_0$ to an instance-specific latent space $\mathcal{Z}$ of the object's kinematic states and a decoder $\mathcal{F}$ that decodes any sampled latent to a plausibly deformed shape. This model is learned with a generative objective, as detailed in Sec. 4.

A successfully learned NeuROK greatly simplifies the solution space of the physical system, since we only need to model the dynamics between latent vector $\mathbf{z}$ in a low-dimensional space. It also eliminates the need for inter-particle physical equations employed in mainstream simulation approaches to keep the deformed shape intact and plausible, as any sampled latent can be mapped into a validly deformed mesh. This allows us to study the system as a whole by considering the energy landscape over different kinematic states of an entire system.

Formalizing this intuition, we simulate this system from the Lagrangian mechanics' perspective in classical physics. The learned NeuROK can be seen as the *generalized-coordinates* of the object-centric physical system, and such systems can be solved in a generic manner by defining the Lagrangian function of the system and solving Euler-Lagrange equations [47]. We detail this process in Sec. 5.

An overview of our framework can be found in Fig. 3.

4. Generative Learning of NeuROK

This section discusses the methodology of learning an encoder-decoder model to predict a NeuROK $(\mathcal{Z}(\mathcal{M}_0), \mathcal{F}(\cdot; \mathcal{M}_0))$ from an input mesh $\mathcal{M}_0$ of a 3D snapshot of a dynamic object. We model the latent

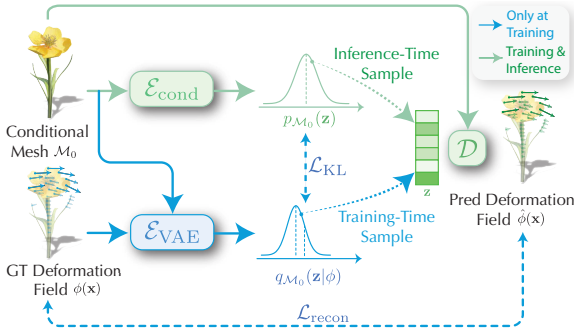


Figure 4. **Generative learning of NEUROK.** During training, we randomly sample an instance mesh and one of its possible deformation fields from the training set, and supervise all three models with KL and reconstruction targets. During inference, we only use $\mathcal{E}_{\text{cond}}$ to obtain the prior distribution $p_{\mathcal{M}_0}(\mathbf{z})$ for the instance $\mathcal{M}_0$ and sample from this distribution a latent, which is further decoded to a predicted deformation field with decoder $\mathcal{D}$.

state space $\mathcal{Z}(\mathcal{M}_0)$ associated with $\mathcal{M}_0$ by studying a surrogate task: learning a generative distribution $p_{\mathcal{M}_0}(\phi)$ over all plausible deformation fields[2] $\phi(\mathbf{x}) : \mathbb{R}^3 \to \mathbb{R}^3$ of $\mathcal{M}_0$. Concretely, we train a conditional variational auto-encoder [46] to learn three models to approximate the instance-specific prior distribution $p_{\mathcal{M}_0}(\phi)$:

1. A **kinematic prior encoder** $\mathcal{E}_{\text{cond}}(\mathcal{M}_0)$ that takes in the conditioning input mesh and outputs the parameters for a prior distribution $p_{\mathcal{M}_0}(\mathbf{z})$ over the latent space $\mathbb{R}^k$.
2. A **variational deformation encoder** $\mathcal{E}_{\text{VAE}}(\phi, \mathcal{M}_0)$ that takes in a deformation field ϕ and a conditional mesh $\mathcal{M}_0$ and produces the parameters of a posterior distribution $q_{\mathcal{M}_0}(\mathbf{z} \mid \phi)$.
3. A **deformation decoder** $\mathcal{D}(\mathbf{z}, \mathcal{M}_0)$ that takes in a sampled latent $\mathbf{z}$ from the conditional prior distribution $p_{\mathcal{M}_0}(\mathbf{z})$ and decodes it into a deformed mesh $\mathcal{M}_{\mathbf{z}}$.

After learning these three models, we extract the high-density region of the latent probability distribution $p_{\mathcal{M}_0}(\mathbf{z})$ as the NEUROK kinematic state space $\mathcal{Z}(\mathcal{M}_0)$, and use the probabilistic decoder $\mathcal{D}(\mathbf{z}, \mathcal{M}_0)$ as the NEUROK mapping $\mathcal{F}(\cdot; \mathcal{M}_0)$.

An overview of these models can be found in Fig. 4. We design these three models with scalable transformer-based architectures and train them on a large-scale 4D dataset to let them learn generalizable kinematic priors.

4.1. Model Architecture

We now discuss the model architectures of $\mathcal{E}_{\text{cond}}$, $\mathcal{E}_{\text{VAE}}$, and $\mathcal{D}$. As a general principle, we use transformers [98] as backbones to ensure that they scale well to large-scale datasets.

Kinematic Prior Encoder. $\mathcal{E}_{\text{cond}}$ takes a conditional mesh $\mathcal{M}_0$ as input and outputs the kinematic prior distribution for $\mathcal{M}_0$. To encode $\mathcal{M}_0$, we evenly sample n_{sample} points from the surface of the input mesh to form a point cloud V_{sample}. We then use the position embedding layer following 3DShape2Vecset [113] to obtain point-wise features $\mathbf{F}_{\text{cond}} \in \mathbb{R}^{n_{\text{sample}} \times F_{\text{pos}}}$, where F_{pos} is the feature dimension of position embeddings. To allow the encoder to take varying numbers of point samples from a single mesh during encoding, we adopt a perceiver-based architecture [37, 116] and store a series of learnable tokens $\{\mathbf{e}_i\}_{i=1}^K$, where K is the number of tokens. With the learnable tokens, we apply multiple blocks of cross-attention and self-attention layers to obtain K encoded features $\{\mathbf{f}_i\}_{i=1}^K$. We flatten the features to form $\mu_{\text{cond}} \in \mathbb{R}^{K \times F_{\text{token}}}$, where F_{token} is the dimension of each token. We use the normal distribution $\mathcal{N}(\mu_{\text{cond}}, \mathbf{I})$ as the instance-specific prior distribution $p_{\mathcal{M}_0}(\mathbf{z})$.

Variational Deformation Encoder. $\mathcal{E}_{\text{VAE}}$ outputs posterior distribution $q_{\mathcal{M}_0}(\mathbf{z} \mid \phi)$ by taking two inputs: a deformation field ϕ and an instance-specific mesh $\mathcal{M}_0$. To parameterize these inputs, at training time, we sample a deformed mesh of $\mathcal{M}_0$. This deformed mesh is represented as $\mathcal{M}_{\mathbf{z}} = (V_{\mathbf{z}}, F)$ with a shared topology F as $\mathcal{M}_0 = (V_0, F)$. Similar to $\mathcal{E}_{\text{cond}}$, we sample a point cloud V_{sample} on the surface of $\mathcal{M}_0$. We then compute the vertex deformation[3] $\delta_{\mathbf{z}} = V_{\mathbf{z}} - V_0$ from $\mathcal{M}_0$ to $\mathcal{M}_{\mathbf{z}}$ and use barycentric interpolation to compute the deformation vector $\delta_{\text{sample}} \in \mathbb{R}^{d_{\text{deform}} \times n_{\text{sample}}}$ of the sampled points, where d_{deform} is the dimensionality of the deformation representation. We then concatenate δ_{sample} with the position vectors of V_{sample} and encode it using the position embedding layer [113] to get the point-wise feature $\mathbf{F}_{\text{VAE}}$ as inputs to the transformer. We similarly use a perceiver-based [37] architecture and store $2 \times K$ learnable tokens. These tokens are mapped to $2 \times K$ features $\{\mathbf{f}_i^{\text{VAE}}\}_{i=1}^{2 \times K}$. We separate those features into two sets and flatten them to represent the mean $\mu_{\text{VAE}} \in \mathbb{R}^{K \times F_{\text{token}}}$ and variance $\sigma_{\text{VAE}} \in \mathbb{R}^{K \times F_{\text{token}}}$ of the posterior. The output posterior distribution $q_{\mathcal{M}_0}(\mathbf{z} \mid \phi)$ is modeled as a Gaussian distribution $\mathcal{N}(\mu_{\text{VAE}}, \sigma_{\text{VAE}})$.

Deformation Decoder. $\mathcal{D}(\mathbf{z}, \mathcal{M}_0)$ is a decoder that decodes sampled latent $\mathbf{z}$ to a deformed mesh from $\mathcal{M}_0$. To implement this, we sample n_{sample} points from the surface of the input mesh to form a query point cloud V_{query}. As the latent space has a dimensionality of $K \times F_{\text{token}}$, we reshape $\mathbf{z}$ into K latent tokens $\{\mathbf{e}_i\}_{i=1}^K$, each with F_{token} dimensions. We then pass the query point cloud and the latent tokens to several blocks of self-attention and cross-attention layers, and predict n_{sample} features $\{\mathbf{f}_i\}_{i=1}^{n_{\text{sample}}}$. We further pass the features into an MLP to get the final deformation vectors $\delta_{\text{pred}} = \{\delta_i\}_{i=1}^{n_{\text{sample}}}$. We deform V_{query} using the predicted deformation vectors, and drive the mesh vertices V_0 by averaging the deformations over K_{drive} nearest sampled points.

[2] To parameterize deformation fields for use in neural networks, we sample points on the mesh and treat their deformations as the parameterization of ϕ.

[3] Practically, we parameterize the deformation of each point with dual quaternions. See the Supp. Mat. for more discussion.

4.2. Dataset and Training

All three models are trained simultaneously on a large-scale 4D dataset of deforming meshes of dynamic objects. We construct this dataset by curating instances from existing works [25, 104] and physical simulation. The details of the dataset can be found in the Supp. Mat.

At each training iteration, we randomly sample an instance from all training instances of the dataset. For this instance, we randomly select two frames in its deformation sequence and obtain two meshes with shared topology. We use the first mesh as $\mathcal{M}_0$ and sample the deformation from the first mesh to the second mesh to form the sampled deformation vector δ_{sample}. These are passed into three models to get the reconstructed deformation δ_{pred}. The models are supervised with the standard conditional VAE target:

$$\mathcal{L} = ||\delta_{\text{sample}} - \delta_{\text{pred}}||_2^2 + \lambda D_{KL}(q_{\mathcal{M}_0}(\mathbf{z} \mid \phi)||p_{\mathcal{M}_0}(\mathbf{z})), \quad (1)$$

where λ is a hyper-parameter and we set $\lambda = 0.01$.

4.3. Dimension Reduction

The raw latent space of the learned VAE can be high-dimensional. To obtain a reduced-order latent space, we further perform a dimension reduction process to compress $\mathcal{Z} \subseteq \mathbb{R}^k$ to a lower-dimensional latent space $\mathcal{Q} \subseteq \mathbb{R}^{k_q}$, where $k_q \ll k$.

We perform the dimension reduction through the Active Subspace Method [22] that reduces the dimensionality of a high-dimensional space $\mathcal{Z}$ by considering a surrogate function $\mathcal{G}(\mathbf{z}) = g(A\mathbf{z} + \epsilon(\mathbf{z}))$, where $G : \mathbb{R}^k \to \mathbb{R}$, $A \in \mathbb{R}^{k_q \times k}$, and $g : \mathbb{R}^{k_q} \to \mathbb{R}$. In this way, the span of the rows of A identifies the directions that matter for $\mathcal{G}$ [7]. We define G in a way that identifies the influence of $\mathbf{z} \in \mathcal{Z}$ on the predicted deformation. Therefore, we formalize G as the 2-norm of δ_{pred} predicted from a set of sampled points on $\mathcal{M}_0$.

5. Generative 4D Simulation

With the predicted NEUROK, our initial task of generating a dynamic sequence of meshes is converted into generating a series of $\{\mathbf{z}_i\}_{i=1}^T$, with $\mathbf{z} \in \mathcal{Z}(\mathcal{M}_0)$. Note that our mapping $\mathcal{F}$ in the learned NEUROK will map any sampled latent $\mathbf{z}$ to a plausibly deformed shape that corresponds to a valid configuration of the studied object-centric physical system. This observation motivates us to use methods from Lagrangian mechanics [47] to generate such dynamics.

5.1. Preliminaries: Lagrangian Mechanics

Lagrangian mechanics studies a physical system by defining a set of parameters $\{\mathbf{q}_1, \cdots, \mathbf{q}_k\}$ that completely define the state of the system in a *configuration space*. Such parameters are called *generalized coordinates* of the system, and their time derivatives $\dot{\mathbf{q}}$ are called *generalized velocities*.

From this perspective, $\mathcal{Z}(\mathcal{M}_0)$ effectively forms a configuration space of the studied object-centric physical system, and any $\mathbf{z} \in \mathcal{Z}(\mathcal{M}_0)$ is a vector of generalized coordinates of the system. Therefore, we can generate the dynamics of $\mathbf{z}$ by using principles in Lagrangian mechanics.

Lagrangian mechanics solves the dynamics of generalized coordinates by defining a smooth function L over the latent space and solving the Euler-Lagrange equation:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{\mathbf{z}}} = \frac{\partial L}{\partial \mathbf{z}}. \quad (2)$$

For most physical systems we study in this paper, we define Lagrangian function $L(\mathbf{z}, \dot{\mathbf{z}}) = T(\mathbf{z}, \dot{\mathbf{z}}) - V(\mathbf{z})$ using the kinetic energy T and potential energy V of the system.

5.2. Euler-Lagrange Equations for NEUROK

With the defined Lagrangian functions and the learned NEUROK, we solve the dynamics of $\mathbf{z} \in \mathcal{Z}(\mathcal{M}_0)$ with:

$$mG(\mathbf{z})\ddot{\mathbf{z}} + C(\mathbf{z}, \dot{\mathbf{z}}) + \nabla_{\mathbf{z}} V = 0, \quad (3)$$

where $G(\mathbf{z}) = J_{\mathbf{z}}^T J_{\mathbf{z}}$, $J_{\mathbf{z}}$ is the Jacobian of $\mathcal{F}$, $C_i = m\sum_{j,k} \Gamma_{ijk}(\mathbf{z})\dot{\mathbf{z}}_j\dot{\mathbf{z}}_k$, Γ_{ijk} is the Christoffel symbol. Its derivation can be found in the Supp. Mat. We solve it with numerical solvers and get the trajectory of $\{\mathbf{z}_i\}_{i=1}^T$.

5.3. Boundary Conditions

Our system takes in conditions such as actions to generate 4D dynamics. They are incorporated by optimizing $(\mathbf{z}_0, \dot{\mathbf{z}}_0)$ to minimize $||\mathbf{x}_0 - \mathcal{F}(\mathbf{z}_0)||_2^2 + ||\dot{\mathbf{x}}_0 - J_{\mathbf{z}}\dot{\mathbf{z}}_0||_2^2$, where $\mathbf{x}_0, \dot{\mathbf{x}}_0$ are input positions and velocities of selected particles of the system. After solving $(\mathbf{z}_0, \dot{\mathbf{z}}_0)$, they serve as the initial condition for solving Eq. (3). See the Supp. Mat. for details.

6. Experiments

6.1. Neural Object Kinematics Learning

We evaluate the effectiveness of NEUROK for learning kinematic state parameterization. Given an object and a target pose of the same object, we estimate an initial kinematic state and identify the optimal latent state vector that deforms the input shape to match the target. For quantitative assessment, we use the test dataset of PartNet-Mobility [104], and evaluate reconstruction accuracy using Chamfer distances [28] and a volumetric consistency metric (IoU).

Baselines. We evaluate two types of baselines for learning object kinematics. NeuralDeformationGraphs (NDG) [11], CANOR [34], and KeyPointDeformer (KPD) [38] model kinematics using implicit representations. FreeArt3D [17] and SINGAPO [71] explicitly learn articulation structures, limiting them to specific object categories.

Results. As shown in Tab. 1 and Fig. 5, our framework consistently outperforms existing methods in inverse kinematics, demonstrating its flexibility and effectiveness.

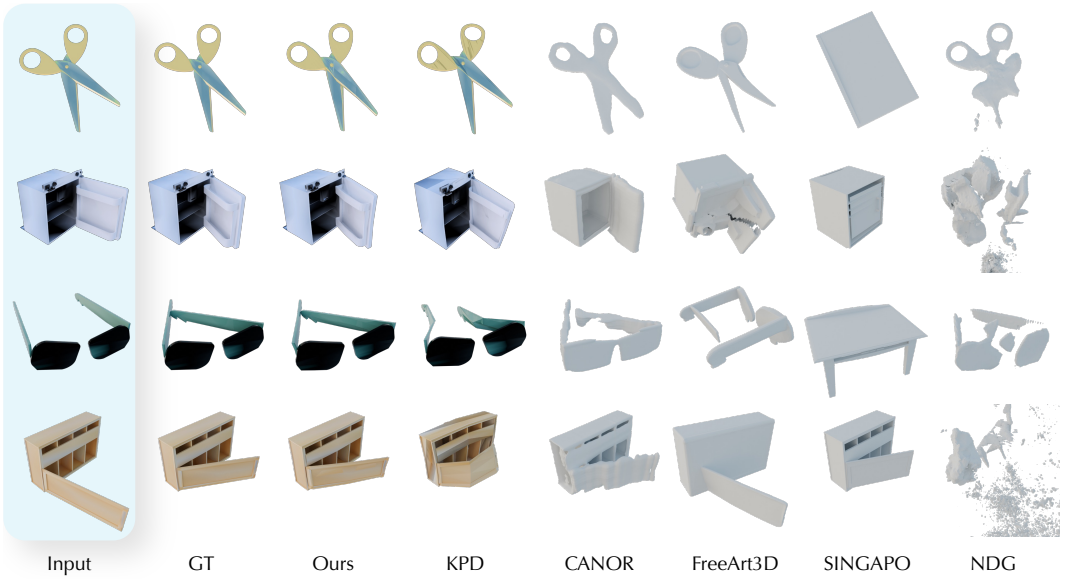


Figure 5. **Qualitative comparison on learning object kinematics.** We evaluate different methods on learning compact and smooth kinematic spaces. Given an input object and the shape of a target pose, we perform inverse kinematics and find the best-matching kinematic state. We compare how well the reconstructed shape decoded from the obtained state vectors matches the target.

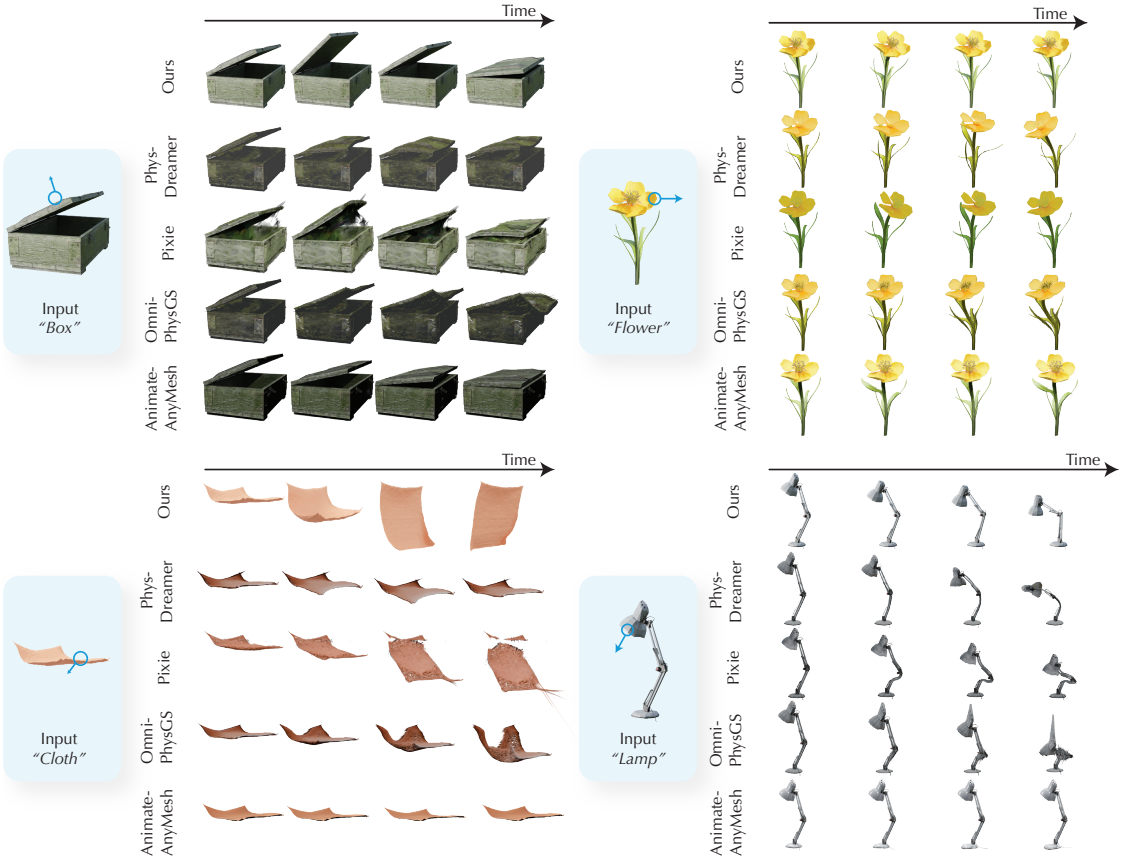


Figure 6. **Qualitative comparison on physically-inspired 4D generation.** We compare against baselines on the task of generating physically-plausible 4D motion given a single shape and conditioning actions.

6.2. Generative 4D Simulation

We show that our pipeline generates 4D simulative dynamics for diverse objects, evaluated across eight objects.

Baselines. We compare against representative methods for generating 4D dynamics from 3D shapes. PhysDreamer [114] distills physical parameters from video

Table 1. **Quantitative comparison on inverse-kinematics optimization.**

	Chamfer (L1) ↓	Chamfer (L2) ↓	IoU ↑
NeuralDeformationGraphs [11]	0.670	0.724	0.289
SINGAPO [71]	0.313	0.200	0.091
FreeArt3D [17]	0.169	0.139	0.354
CANOR [34]	0.082	0.067	0.568
KeyPointDeformer [38]	0.067	0.067	0.570
NeuROK (ours)	**0.028**	**0.028**	**0.764**
NeuROK w/o Model Reduction	0.045	0.059	0.711
NeuROK w/o Data Augmentation	0.036	0.041	0.724
NeuROK w/o Dual-Quaternion	0.033	0.037	0.728

Table 2. **Quantitative comparison on physically-inspired generation.** We report user study preferences along with metrics from VBench [36] and WorldScore [27]. AQ: Aesthetic Quality, DD: Dynamic Degrees, IQ: Imaging Quality, CLIP: CLIP score [89], MM: Motion Magnitude.

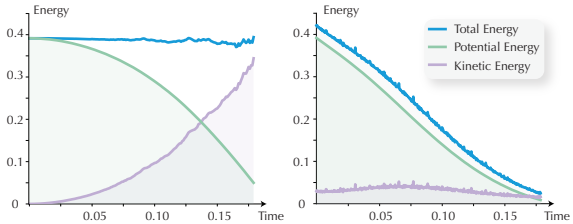
	User Study		VBench [36]			WorldScore [27]	
	Alignment ↑	Realism ↑	AQ ↑	DD ↑	IQ ↑	CLIP ↑	MM ↑
PhysDreamer [114]	5.95%	5.36%	0.362	0.500	48.432	0.716	0.783
OmniPhysGS [69]	1.67%	0.48%	0.380	0.625	48.937	0.690	0.544
Pixie [49]	5.12%	4.17%	0.392	0.625	46.177	0.659	0.857
AnimateAnyMesh [102]	5.83%	6.67%	0.450	0.625	48.370	0.730	0.889
NeuROK (ours)	**81.43%**	**83.33%**	**0.483**	**0.750**	**51.100**	**0.761**	**2.343**

Figure 7. **Simulating real objects.** Our model can be used to simulate real-captured objects. See the Supp. Mat. for more results.

models, Pixie [49] predicts simulation parameters using amortized-inference networks, and OmniPhysGS [69] represents each asset with material-aware Constitutive Gaussians for general physics-based dynamics. AnimateAnyMesh [102] is an end-to-end 4D generator trained on large-scale 4D data.

Metrics. Predicting 4D dynamics from 3D shapes is inherently ambiguous, so we evaluate plausibility and visual quality of the generated motions. A user study with 105 users assesses action alignment and realism. We also report metrics from VBench [36] and WorldScore [27].

Results. Quantitative and qualitative comparisons in Tab. 2 and Fig. 6 show that existing baselines perform well only within their specialized domains. Physically based methods (PhysDreamer [114], OmniPhysGS [69], Pixie [49]) can handle certain material categories but generalize poorly, while end-to-end methods (AnimateAnyMesh [102]) lack fine-grained conditioning and struggle on rarely encountered object types. Across all settings, our method consistently generates the most physically plausible and visually realistic 4D dynamics, demonstrating strong generalization to diverse object categories.



(a) Energy Evolution of Generated Traj. with Euler-Lagrangian Dynamics

(b) Energy Evolution of Interpolated Traj. w/o Euler-Lagrangian Dynamics

Figure 8. **Analysis of energy conservation.** Our approach maintains physical consistency in the generated trajectories through Euler–Lagrangian modeling. Under this formulation, the total energy of the simulated motion remains approximately constant.

Figure 9. **Generalization on unseen categories.** Our model can generalize to novel object categories that are completely not present in the training data.

Simulating Real Objects. Our pipeline can also simulate and manipulate real scenes. We scan a real scene and apply our approach to simulate the dynamics of the objects within it. As shown in Fig. 7, our method successfully simulates the closing motion of the laptop on the desk.

6.3. Analysis and Ablation Studies

Analysis of Physical Consistency. We analyze the physical consistency of the proposed framework in Fig. 8. As demonstrated, our method preserves the basic conservation law of energy by leveraging a physically-inspired framework from Lagrangian mechanics.

Generalization on Unseen Categories. Our method learns common dynamic structures from the training dataset and successfully generalizes them to entirely new object categories. As shown in Fig. 9, a NeuROK variant trained only on PartNet-Mobility [104] categories can still generate plausible dynamics for unseen object types.

Ablation Studies. We evaluate the impact of key design choices in Tab. 1. The results show that model reduction, training data augmentation, and our deformation parameterization each contribute significantly to the overall performance of the proposed framework.

7. Conclusion

We have introduced a novel framework, NeuROK, for generating 4D simulative dynamics from static 3D shapes, bridging physical principles and learned latent spaces through a physically inspired formulation. Our work opens up promising future research directions and introduces a new research paradigm in 4D visual generation.

Acknowledgments. This work is in part supported by NSF RI #2211258 and #2338203, ONR YIP N00014-24-1-2117, ONR MURI N00014-22-1-2740, the Stanford Institute for Human-Centered AI (HAI), and the Magic Grant from the Brown Institute for Media Innovation. We acknowledge the compute support from the NSF ACCESS program #CIS250696, Stanford Data Science and Marlowe Computing Platform, and the AMD University Program for AI & HPC Cluster. We thank Robyn Lockwood (Stanford Language Center) for editorial and writing suggestions that improved the clarity of the manuscript. We thank Chong Zeng and Ruocheng Wang for early feedback on the manuscript and members of Stanford Vision and Learning Lab and Stanford Graphics Lab for fruitful discussion.

References

[1] Stephen W Bailey, Dalton Omens, Paul Dilorenzo, and James F O'Brien. Fast and deep facial deformations. *ACM Transactions on Graphics (TOG)*, 39(4):94–1, 2020. 3

[2] Yohai Bar-Sinai, Stephan Hoyer, Jason Hickey, and Michael P Brenner. Learning data-driven discretizations for partial differential equations. *Proceedings of the National Academy of Sciences*, 116(31):15344–15349, 2019. 3

[3] Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. *Advances in neural information processing systems*, 29, 2016. 3

[4] Peter W Battaglia, Jessica B Hamrick, and Joshua B Tenenbaum. Simulation as an engine of physical scene understanding. *Proceedings of the national academy of sciences*, 110(45):18327–18332, 2013. 3

[5] Peter Benner, Serkan Gugercin, and Karen Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM review*, 57(4):483–531, 2015. 3

[6] Ravinder Bhattoo, Sayan Ranu, and NM Krishnan. Learning articulated rigid body dynamics with lagrangian graph neural network. *Advances in Neural Information Processing Systems*, 35:29789–29800, 2022. 3

[7] David Bindel. Numerical methods for data science, 2018. 6

[8] Volker Blanz and Thomas Vetter. Face recognition based on fitting a 3d morphable model. *IEEE Transactions on pattern analysis and machine intelligence*, 25(9):1063–1074, 2003. 3

[9] James Booth, Anastasios Roussos, Stefanos Zafeiriou, Allan Ponniah, and David Dunaway. A 3d morphable model learnt from 10,000 faces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5543–5552, 2016. 3

[10] Sofien Bouaziz, Sebastian Martin, Tiantian Liu, Ladislav Kavan, and Mark Pauly. Projective dynamics: fusing constraint projections for fast simulation. *ACM Transactions on Graphics (TOG)*, 33(4):1–11, 2014. 2

[11] Aljaz Bozic, Pablo Palafox, Michael Zollhofer, Justus Thies, Angela Dai, and Matthias Nießner. Neural deformation graphs for globally-consistent non-rigid reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1450–1459, 2021. 3, 6, 8

[12] Junhao Cai, Yuji Yang, Weihao Yuan, Yisheng He, Zilong Dong, Liefeng Bo, Hui Cheng, and Qifeng Chen. Gic: Gaussian-informed continuum for physical property identification and simulation. *Advances in Neural Information Processing Systems*, 37:75035–75063, 2024. 2

[13] Michael B Chang, Tomer Ullman, Antonio Torralba, and Joshua B Tenenbaum. A compositional object-based approach to learning physical dynamics. *arXiv preprint arXiv:1612.00341*, 2016. 3

[14] Yue Chang, Peter Yichen Chen, Zhecheng Wang, Maurizio M Chiaramonte, Kevin Carlberg, and Eitan Grinspun. Licrom: Linear-subspace continuous reduced order modeling with neural fields. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–12, 2023. 3

[15] Boyuan Chen, Hanxiao Jiang, Shaowei Liu, Saurabh Gupta, Yunzhu Li, Hao Zhao, and Shenlong Wang. Physgen3d: Crafting a miniature interactive world from a single image. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6178–6189, 2025. 2

[16] Chuhao Chen, Zhiyang Dou, Chen Wang, Yiming Huang, Anjun Chen, Qiao Feng, Jiatao Gu, and Lingjie Liu. Vid2sim: Generalizable, video-based reconstruction of appearance, geometry and physics for mesh-free simulation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26545–26555, 2025. 2

[17] Chuhao Chen, Isabella Liu, Xinyue Wei, Hao Su, and Minghua Liu. Freeart3d: Training-free articulated object generation using 3d diffusion. In *SIGGRAPH Asia 2025 Conference Papers*, 2025. 6, 8

[18] Honglin Chen, Rundi Wu, Eitan Grinspun, Changxi Zheng, and Peter Yichen Chen. Implicit neural spatial representations for time-dependent pdes. In *International Conference on Machine Learning*, pages 5162–5177. PMLR, 2023. 3

[19] Peter Yichen Chen, Jinxu Xiang, Dong Heon Cho, Yue Chang, GA Pershing, Henrique Teles Maia, Maurizio M Chiaramonte, Kevin Carlberg, and Eitan Grinspun. Crom: Continuous reduced-order modeling of pdes using implicit neural representations. *arXiv preprint arXiv:2206.02607*, 2022. 3

[20] Peter Yichen Chen, Maurizio M Chiaramonte, Eitan Grinspun, and Kevin Carlberg. Model reduction for the material point method via an implicit neural representation of the deformation map. *Journal of Computational Physics*, 478: 111908, 2023. 3

[21] Zoey Chen, Aaron Walsman, Marius Memmel, Kaichun Mo, Alex Fang, Karthikeya Vemuri, Alan Wu, Dieter Fox, and Abhishek Gupta. URDFormer: A pipeline for constructing articulated simulation environments from real-world images. *arXiv preprint arXiv:2405.11656*, 2024. 2

[22] Paul G Constantine, Eric Dow, and Qiqi Wang. Active subspace methods in theory and practice: applications to kriging surfaces. *SIAM Journal on Scientific Computing*, 36(4): A1500–A1524, 2014. 6

[23] Miles Cranmer, Sam Greydanus, Stephan Hoyer, Peter Battaglia, David Spergel, and Shirley Ho. Lagrangian neural networks. *arXiv preprint arXiv:2003.04630*, 2020. 3

[24] Rishit Dagli, Donglai Xiang, Vismay Modi, Charles Loop, Clement Fuji Tsang, Anka He Chen, Anita Hu, Gavriel State, David IW Levin, and Maria Shugrina. Vomp: Predicting volumetric mechanical property fields. *arXiv preprint arXiv:2510.22975*, 2025. 2

[25] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, 36:35799–35813, 2023. 6

[26] Tao Du, Kui Wu, Pingchuan Ma, Sebastien Wah, Andrew Spielberg, Daniela Rus, and Wojciech Matusik. Diffpd: Differentiable projective dynamics. *ACM Transactions on Graphics (ToG)*, 41(2):1–21, 2021. 2

[27] Haoyi Duan, Hong-Xing Yu, Sirui Chen, Li Fei-Fei, and Jiajun Wu. Worldscore: A unified evaluation benchmark for world generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2025. 8

[28] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 605–613, 2017. 6

[29] Yutao Feng, Yintong Shang, Xuan Li, Tianjia Shao, Chenfanfu Jiang, and Yin Yang. Pie-nerf: Physics-based interactive elastodynamics with nerf. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4450–4461, 2024. 2

[30] Marc Finzi, Ke Alexander Wang, and Andrew G Wilson. Simplifying hamiltonian and lagrangian neural networks via explicit constraints. *Advances in neural information processing systems*, 33:13880–13889, 2020. 3

[31] Lawson Fulton, Vismay Modi, David Duvenaud, David IW Levin, and Alec Jacobson. Latent-space dynamics for reduced deformable simulation. In *Computer graphics forum*, pages 379–391. Wiley Online Library, 2019. 3

[32] Simon Giebenhain, Tobias Kirschstein, Markos Georgopoulos, Martin Rünz, Lourdes Agapito, and Matthias Nießner. Learning neural parametric head models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21003–21012, 2023. 3

[33] Michelle Guo, Matt Jen-Yuan Chiang, Igor Santesteban, Nikolaos Sarafianos, Hsiao-yu Chen, Oshri Halimi, Aljaž Božič, Shunsuke Saito, Jiajun Wu, C Karen Liu, et al. Pgc: Physics-based gaussian cloth from a single pose. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21215–21225, 2025. 2

[34] Guangzhao He, Chen Geng, Shangzhe Wu, and Jiajun Wu. Category-agnostic neural object rigging. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22078–22088, 2025. 3, 6, 8

[35] Qixing Huang, Xiangru Huang, Bo Sun, Zaiwei Zhang, Junfeng Jiang, and Chandrajit Bajaj. Arapreg: An as-rigid-as possible regularization loss for learning deformable shape generators. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5815–5825, 2021. 3

[36] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024. 8

[37] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International conference on machine learning*, pages 4651–4664. PMLR, 2021. 5

[38] Tomas Jakab, Richard Tucker, Ameesh Makadia, Jiajun Wu, Noah Snavely, and Angjoo Kanazawa. Keypointdeformer: Unsupervised 3d keypoint discovery for shape control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12783–12792, 2021. 3, 6, 8

[39] Doug L James and Dinesh K Pai. Dyrt: Dynamic response textures for real time deformation simulation with graphics hardware. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 582–585, 2002. 3

[40] Doug L James, Jernej Barbič, and Dinesh K Pai. Precomputed acoustic transfer: output-sensitive, accurate sound generation for geometrically complex vibration sources. *ACM Transactions on Graphics (TOG)*, 25(3):987–995, 2006. 3

[41] 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. 2, 4

[42] Hanxiao Jiang, Hao-Yu Hsu, Kaifeng Zhang, Hsin-Ni Yu, Shenlong Wang, and Yunzhu Li. Phystwin: Physics-informed reconstruction and simulation of deformable objects from videos. *ICCV*, 2025. 2

[43] Zhenyu Jiang, Cheng-Chun Hsu, and Yuke Zhu. Ditto: Building digital twins of articulated objects from interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5616–5626, 2022. 2

[44] Takuhiro Kaneko. Improving physics-augmented continuum neural radiance field-based geometry-agnostic system identification with lagrangian particle optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5470–5480, 2024. 2

[45] Justin Kerr, Chung Min Kim, Mingxuan Wu, Brent Yi, Qianqian Wang, Ken Goldberg, and Angjoo Kanazawa. Robot see robot do: Imitating articulated object manipulation with monocular 4d reconstruction. *arXiv preprint arXiv:2409.18121*, 2024. 2

[46] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 5

[47] Lev Davidovich Landau and Evgeniĭ Mikhaĭlovich Lifshitz. *Mechanics*. CUP Archive, 1960. 2, 4, 6

[48] Long Le, Jason Xie, William Liang, Hung-Ju Wang, Yue Yang, Yecheng Jason Ma, Kyle Vedder, Arjun Krishna, Dinesh Jayaraman, and Eric Eaton. Articulate-anything: Automatic modeling of articulated objects via a vision-language foundation model. *arXiv preprint arXiv:2410.13882*, 2024. 2

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

[50] Kookjin Lee and Kevin T Carlberg. Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *Journal of Computational Physics*, 404:108973, 2020. 3

[51] Jiahui Lei, Congyue Deng, William B Shen, Leonidas J Guibas, and Kostas Daniilidis. Nap: Neural 3d articulated object prior. *Advances in Neural Information Processing Systems*, 36:31878–31894, 2023. 2

[52] Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, et al. Behavior-1k: A benchmark for embodied ai with 1,000 everyday activities and realistic simulation. In *Conference on Robot Learning*, pages 80–93. PMLR, 2023. 2

[53] Chenchang Li, Zihao Ai, Tong Wu, Xiaosa Li, Wenbo Ding, and Huazhe Xu. Deformnet: Latent space modeling and dynamics prediction for deformable object manipulation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 14770–14776. IEEE, 2024. 3

[54] Sizhe Lester Li, Annan Zhang, Boyuan Chen, Hanna Matusik, Chao Liu, Daniela Rus, and Vincent Sitzmann. Controlling diverse robots by inferring jacobian fields with deep networks. *Nature*, pages 1–7, 2025. 3

[55] Xuan Li, Yadi Cao, Minchen Li, Yin Yang, Craig Schroeder, and Chenfanfu Jiang. Plasticitynet: Learning to simulate metal, sand, and snow for optimization time integration. *Advances in Neural Information Processing Systems*, 35:27783–27796, 2022. 2

[56] 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. 2

[57] Xuan Li, Chang Yu, Wenxin Du, Ying Jiang, Tianyi Xie, Yunuo Chen, Yin Yang, and Chenfanfu Jiang. Dress-1-to-3: Single image to simulation-ready 3d outfit with diffusion prior and differentiable physics. *ACM Transactions on Graphics (TOG)*, 44(4):1–16, 2025. 2

[58] Yunzhu Li, Jiajun Wu, Russ Tedrake, Joshua B Tenenbaum, and Antonio Torralba. Learning particle dynamics for manipulating rigid bodies, deformable objects, and fluids. *arXiv preprint arXiv:1810.01566*, 2018. 3

[59] Yifei Li, Tao Du, Kui Wu, Jie Xu, and Wojciech Matusik. Diffcloth: Differentiable cloth simulation with dry frictional contact. *ACM Transactions on Graphics (TOG)*, 42 (1):1–20, 2022. 2

[60] Yunzhu Li, Shuang Li, Vincent Sitzmann, Pulkit Agrawal, and Antonio Torralba. 3d neural scene representations for visuomotor control. In *Conference on Robot Learning*, pages 112–123. PMLR, 2022. 3

[61] Yichen Li, Peter Yichen Chen, Tao Du, and Wojciech Matusik. Learning preconditioners for conjugate gradient pde solvers. In *International Conference on Machine Learning*, pages 19425–19439. PMLR, 2023. 3

[62] Yifei Li, Hsiao-yu Chen, Egor Larionov, Nikolaos Sarafianos, Wojciech Matusik, and Tuur Stuyck. Diffavatar: Simulation-ready garment optimization with differentiable simulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4368–4378, 2024. 2

[63] Yue Li, Gene Wei-Chin Lin, Egor Larionov, Aljaz Bozic, Doug Roble, Ladislav Kavan, Stelian Coros, Bernhard Thomaszewski, Tuur Stuyck, and Hsiao-Yu Chen. Self-supervised learning of latent space dynamics. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 8(4):1–18, 2025. 3

[64] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020. 3

[65] Zizhang Li, Hong-Xing Yu, Wei Liu, Yin Yang, Charles Herrmann, Gordon Wetzstein, and Jiajun Wu. Wonderplay: Dynamic 3d scene generation from a single image and actions. *arXiv preprint arXiv:2505.18151*, 2025. 2

[66] Junbang Liang, Ming Lin, and Vladlen Koltun. Differentiable cloth simulation for inverse problems. *Advances in neural information processing systems*, 32, 2019. 2

[67] Jiajing Lin, Zhenzhong Wang, Dejun Xu, Shu Jiang, Yunpeng Gong, and Min Jiang. Phys4dgen: Physics-compliant 4d generation with multi-material composition perception. *arXiv preprint arXiv:2411.16800*, 2024. 2

[68] Jiajing Lin, Shu Jiang, Qingyuan Zeng, Zhenzhong Wang, and Min Jiang. Visionlaw: Inferring interpretable intrinsic dynamics from visual observations via bilevel optimization. *arXiv preprint arXiv:2508.13792*, 2025.

[69] Yuchen Lin, Chenguo Lin, Jianjin Xu, and Yadong Mu. Omniphysgs: 3d constitutive gaussians for general physics-based dynamics generation. *arXiv preprint arXiv:2501.18982*, 2025. 2, 8

[70] Jiayi Liu, Ali Mahdavi-Amiri, and Manolis Savva. Paris: Part-level reconstruction and motion analysis for articulated objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 352–363, 2023. 2

[71] Jiayi Liu, Denys Iliash, Angel X Chang, Manolis Savva, and Ali Mahdavi-Amiri. Singapo: Single image controlled generation of articulated parts in objects. *arXiv preprint arXiv:2410.16499*, 2024. 2, 6, 8

[72] Ruoshi Liu, Alper Canberk, Shuran Song, and Carl Vondrick. Differentiable robot rendering. *arXiv preprint arXiv:2410.13851*, 2024. 3

[73] 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. 2

[74] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: a skinned multi-person linear model. *ACM Transactions on Graphics (TOG)*, 34(6):1–16, 2015. 3

[75] Michael Lutter, Christian Ritter, and Jan Peters. Deep lagrangian networks: Using physics as model prior for deep learning. *arXiv preprint arXiv:1907.04490*, 2019. 3

[76] Pingchuan Ma, Peter Yichen Chen, Bolei Deng, Joshua B Tenenbaum, Tao Du, Chuang Gan, and Wojciech Matusik. Learning neural constitutive laws from motion observations for generalizable pde dynamics. In *International Conference on Machine Learning*, pages 23279–23300. PMLR, 2023. 3

[77] Miles Macklin, Matthias Müller, and Nuttapong Chentanez. Xpbd: position-based simulation of compliant constrained dynamics. In *Proceedings of the 9th International Conference on Motion in Games*, pages 49–54, 2016. 2

[78] Arman Maesumi, Paul Guerrero, Noam Aigerman, Vladimir Kim, Matthew Fisher, Siddhartha Chaudhuri, and Daniel Ritchie. Explorable mesh deformation subspaces from unstructured 3d generative models. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023. 3

[79] Zhao Mandi, Yijia Weng, Dominik Bauer, and Shuran Song. Real2code: Reconstruct articulated objects via code generation. *arXiv preprint arXiv:2406.08474*, 2024. 2

[80] Zhao Mandi, Yifan Hou, Dieter Fox, Yashraj Narang, Ajay Mandlekar, and Shuran Song. Dexmachina: Functional retargeting for bimanual dexterous manipulation. *arXiv preprint arXiv:2505.24853*, 2025. 2

[81] Himangi Mittal, Peiye Zhuang, Hsin-Ying Lee, and Shubham Tulsiani. Uniphy: Learning a unified constitutive model for inverse physics simulation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16208–16218, 2025. 2, 3

[82] Vismay Modi, Nicholas Sharp, Or Perel, Shinjiro Sueda, and David IW Levin. Simplicits: Mesh-free, geometry-agnostic elastic simulation. *ACM Transactions on Graphics (TOG)*, 43(4):1–11, 2024. 2, 3

[83] Atsuhiro Noguchi, Umar Iqbal, Jonathan Tremblay, Tatsuya Harada, and Orazio Gallo. Watch it move: Unsupervised discovery of 3d joints for re-posing of articulated objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3677–3687, 2022. 2

[84] Pablo Palafox, Aljaž Božič, Justus Thies, Matthias Nießner, and Angela Dai. Npms: Neural parametric models for 3d deformable shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12695–12705, 2021. 3

[85] Pablo Palafox, Nikolaos Sarafianos, Tony Tung, and Angela Dai. Spams: Structured implicit parametric models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12851–12860, 2022. 3

[86] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985, 2019. 3

[87] Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter Battaglia. Learning mesh-based simulation with graph networks. In *International conference on learning representations*, 2020. 3

[88] Yi-Ling Qiao, Alexander Gao, and Ming Lin. Neuphysics: Editable neural geometry and physics from monocular videos. *Advances in Neural Information Processing Systems*, 35:12841–12854, 2022. 2

[89] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 8

[90] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019. 3

[91] Cristian Romero, Dan Casas, Jesús Pérez, and Miguel Otaduy. Learning contact corrections for handle-based subspace dynamics. *ACM Transactions on Graphics (ToG)*, 40 (4):1–12, 2021. 3

[92] Quanyuan Ruan, Jiabao Lei, Wenhao Yuan, Yanglin Zhang, Dekun Lu, Guiliang Liu, and Kui Jia. Prof. robot: Differentiable robot rendering without static and self-collisions. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22562–22572, 2025. 3

[93] Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter Battaglia. Learning to simulate complex physics with graph networks. In *International conference on machine learning*, pages 8459–8468. PMLR, 2020. 3

[94] Connor Schenck and Dieter Fox. Spnets: Differentiable fluid dynamics for deep neural networks. In *Conference on Robot Learning*, pages 317–335. PMLR, 2018. 3

[95] Nicholas Sharp, Cristian Romero, Alec Jacobson, Etienne Vouga, Paul Kry, David IW Levin, and Justin Solomon. Data-free learning of reduced-order kinematics. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–9, 2023. 3

[96] Bokui Shen, Zhenyu Jiang, Christopher Choy, Silvio Savarese, Leonidas J Guibas, Anima Anandkumar, and Yuke Zhu. Action-conditional implicit visual dynamics for deformable object manipulation. *The International Journal of Robotics Research*, 43(4):437–455, 2024. 3

[97] Siyuan Shen, Yang Yin, Tianjia Shao, He Wang, Chenfanfu Jiang, Lei Lan, and Kun Zhou. High-order differentiable autoencoder for nonlinear model reduction. *arXiv preprint arXiv:2102.11026*, 2021. 3

[98] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2, 5

[99] Hrishikesh Viswanath, Yue Chang, Aleksey Panas, Julius Berner, Peter Yichen Chen, and Aniket Bera. Reduced-order neural operators: Learning lagrangian dynamics on highly sparse graphs. *arXiv preprint arXiv:2407.03925*, 2024. 3

[100] Jiahong Wang, Yinwei Du, Stelian Coros, and Bernhard Thomaszewski. Neural modes: Self-supervised learning of nonlinear modal subspaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23158–23167, 2024. 3

[101] Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. 3dn: 3d deformation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1038–1046, 2019. 3

[102] Zijie Wu, Chaohui Yu, Fan Wang, and Xiang Bai. Animateanymesh: A feed-forward 4d foundation model for text-driven universal mesh animation. *arXiv preprint arXiv:2506.09982*, 2025. 8

[103] Hongchi Xia, Entong Su, Marius Memmel, Arhan Jain, Raymond Yu, Numfor Mbiziwo-Tiapo, Ali Farhadi, Abhishek Gupta, Shenlong Wang, and Wei-Chiu Ma. Drawer: Digital reconstruction and articulation with environment realism. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21771–21782, 2025. 2

[104] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, et al. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11097–11107, 2020. 2, 6, 8

[105] 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. 2

[106] Xinli Xu, Wenhang Ge, Dicong Qiu, ZhiFei Chen, Dongyu Yan, Zhuoyun Liu, Haoyu Zhao, Hanfeng Zhao, Shunsi Zhang, Junwei Liang, et al. Gaussianproperty: Integrating physical properties to 3d gaussians with lmms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7231–7240, 2025. 2

[107] Zhenjia Xu, Jiajun Wu, Andy Zeng, Joshua B Tenenbaum, and Shuran Song. Densephysnet: Learning dense physical object representations via multi-step dynamic interactions. *arXiv preprint arXiv:1906.03853*, 2019. 2

[108] Haitao Yang, Xiangru Huang, Bo Sun, Chandrajit Bajaj, and Qixing Huang. Gencorres: Consistent shape matching via coupled implicit-explicit shape generative models. *arXiv preprint arXiv:2304.10523*, 2023. 3

[109] Haitao Yang, Bo Sun, Liyan Chen, Amy Pavel, and Qixing Huang. Geolatent: A geometric approach to latent space design for deformable shape generators. *ACM Transactions on Graphics (TOG)*, 42(6):1–20, 2023.

[110] Wang Yifan, Noam Aigerman, Vladimir G Kim, Siddhartha Chaudhuri, and Olga Sorkine-Hornung. Neural cages for detail-preserving 3d deformations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 75–83, 2020.

[111] Seungwoo Yoo, Juil Koo, Kyeongmin Yeo, and Minhyuk Sung. Neural pose representation learning for generating and transferring non-rigid object poses. *arXiv preprint arXiv:2406.09728*, 2024. 3

[112] 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. 2

[113] Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Transactions On Graphics (TOG)*, 42(4):1–16, 2023. 5

[114] 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, 2024. 2, 7, 8

[115] Haoyu Zhao, Hao Wang, Xingyue Zhao, Hao Fei, Hongqiu Wang, Chengjiang Long, and Hua Zou. Physsplat: Efficient physics simulation for 3d scenes via mllm-guided gaussian splatting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5242–5252, 2025. 2

[116] Zibo Zhao, Wen Liu, Xin Chen, Xianfang Zeng, Rui Wang, Pei Cheng, Bin Fu, Tao Chen, Gang Yu, and Shenghua Gao. Michelangelo: Conditional 3d shape generation based on shape-image-text aligned latent representation. *Advances in neural information processing systems*, 36:73969–73982, 2023. 5

[117] Yang Zheng, Qingqing Zhao, Guandao Yang, Wang Yifan, Donglai Xiang, Florian Dubost, Dmitry Lagun, Thabo Beeler, Federico Tombari, Leonidas Guibas, et al. Physavatar: Learning the physics of dressed 3d avatars from visual observations. In *European Conference on Computer Vision*, pages 262–284. Springer, 2024. 2

[118] 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, 2024. 2

[119] Keyang Zhou, Bharat Lal Bhatnagar, and Gerard Pons-Moll. Unsupervised shape and pose disentanglement for 3d meshes. In *European conference on computer vision*, pages 341–357. Springer, 2020. 3

[120] Zeshun Zong, Xuan Li, Minchen Li, Maurizio M Chiaramonte, Wojciech Matusik, Eitan Grinspun, Kevin Carlberg, Chenfanfu Jiang, and Peter Yichen Chen. Neural stress fields for reduced-order elastoplasticity and fracture. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023. 3

[121] Silvia Zuffi, Angjoo Kanazawa, David Jacobs, and Michael J. Black. 3D menagerie: Modeling the 3D shape and pose of animals. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3