DYNABEAM: FROM MOTION-DRIVEN PARAMETER IDENTIFICATION TO REAL-TIME PHYSICAL SIMULATION OF 3D VEGETATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Our 3D world is filled with vegetation that is often in motion due to interactions with wind and other physical forces. While recent representations excel at capturing and generating these dynamic scenes, extracting their underlying physics traditionally requires computationally heavy inverse simulations or appearance-based learning that struggle with speed and generalization. In this work, we introduce DynaBeam, a framework that models dynamic 3D vegetation, ranging from houseplants to trees, as hierarchical graphs of connected beam elements. We show that the geometric structure and motion of these beams provide a strong enough signal to efficiently extract both the intrinsic physical parameters like stiffness and the external forces driving the movement. By explicitly separating geometry, motion, and force, DynaBeam bypasses expensive simulation solvers to achieve accurate dynamic scene reconstruction in mere seconds. Furthermore, because we recover physical parameters rather than fitting visual patterns, our representation natively supports real-time physical simulation under entirely new and unobserved forces. Extensive evaluations show that DynaBeam delivers faster and more accurate physical inversion and forward simulation than prior methods, enabling interactive animation and physical reconstruction from real-world videos.

1 INTRODUCTION

In natural 3D environments, vegetation is often in motion due to wind and physical contact. While modern 3D representations have made significant progress in capturing and replaying these dynamic scenes from video, simply reconstructing observed motion is not the same as recovering the underlying physics. A model might accurately replay a recorded swaying motion, but it will fail to predict how that same tree would react to a novel force, such as wind blowing from a new direction. Therefore, to enable true interactive simulation, we must move beyond memorizing visual trajectories and instead identify the actual physical parameters governing the object.

Identifying these physical properties from visual data, however, remains a major challenge. Current approaches generally fall into two extremes. The first relies on general-purpose differentiable simulators (e.g., FEM or MPM) coupled with dense volumetric representations (Jatavallabhula et al., 2021; Li et al., 2023; Xie et al., 2024; Zhang et al., 2024). While expressive, they require unrolling expensive physical simulations at every optimization step, often taking hours to invert a single scene. The second direction uses compact representations learned directly from the observed motion, such as ModalNeRF (Petitjean et al., 2023). This approach is much faster, but it is fundamentally limited by the input data: if a specific vibration mode was not excited by the observed force, it is entirely missing from the model, preventing accurate simulation of novel physical interactions. Spring–mass models combined with 3D reconstruction (Jiang et al., 2025; Zhong et al., 2024) are cheaper to simulate, but their springs resist only stretching and compression, not bending, which dominates the motion of branches.

In this work, we introduce DynaBeam, a framework that uses the natural shape of vegetation to build a highly efficient physical model. Trees and plants are mostly made of long, thin branches, meaning their bending behavior can be accurately captured using Euler–Bernoulli beam elements (Bauchau & Craig, 2009). While previous work used these models to animate trees with known physical

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

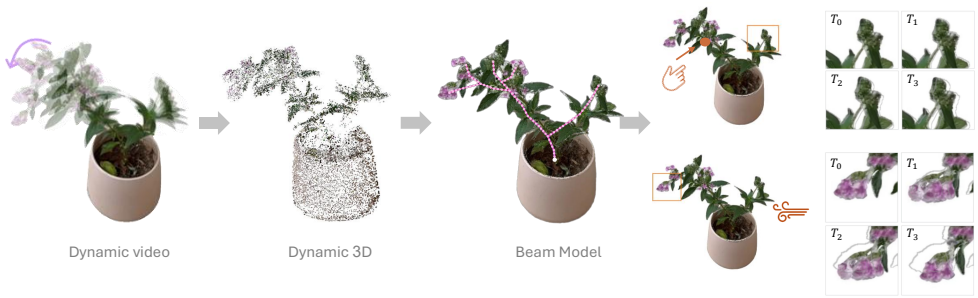


Figure 1: Given a dynamic 3D reconstruction of a plant (left), DynaBeam builds a beam model of its structure and recovers its physical parameters. The recovered model then acts as an interactive, real-time simulator (right): users apply forces not seen in the input, such as a point push (top) or a distributed wind load (bottom). Frames T_0–T_3 show the simulated response; thin lines mark the rest position.

properties (Stam, 1997), we apply this structural principle to the inverse problem: identifying the unknown physical system directly from the observed 3D shape and motion.

Instead of using a dense volumetric simulator, we represent the plant as a 3D graph of connected beams built directly from its static geometry. This skeleton natively defines the system's global mass and stiffness matrices, from which we compute its natural vibration modes. We then use the observed 3D motion to solve for the system's unknown material properties and the external forces driving the movement. Crucially, because the underlying vibration modes are derived from the physical structure rather than extracted from the motion, the resulting model is not limited by the input data. Even if a specific bending motion is barely seen in the input sequence, the corresponding mode remains present in the model, ready to respond to novel forces.

By combining this structural skeleton with the observed motion, we identify an effective dynamical system for the plant. As we demonstrate analytically, visual motion alone only reveals the ratio of stiffness to mass, meaning absolute stiffness and density cannot be independently extracted without external assumptions. Therefore, we explicitly separate the object's intrinsic physical properties, namely its stiffness-to-density ratio and damping ratio, from the sequence-specific external force. By cleanly separating geometry, motion, and force in this way, we achieve accurate physical inversion in mere seconds.

We evaluate DynaBeam quantitatively on synthetic trees with ground-truth dynamics, where it recovers physical parameters more accurately and faster than prior methods given the same inputs, and predicts responses to unseen forces that a modal basis extracted from the observed motion cannot. We further show qualitative results on dynamic 3D reconstructions obtained from real monocular videos, where the recovered model serves as an interactive, real-time simulator (Fig. 1).

In summary, our main contributions are:

- An inverse-physics method that recovers the physical parameters and driving force of 3D vegetation in seconds, without repeated forward simulation, grounded in an analysis that reduces material identification to fitting a single scale factor to the vibration spectrum of the beam structure.
- A beam-based representation whose vibration modes are derived from geometry, so the recovered model responds to forces that excite modes absent from the observed motion.
- A quantitative evaluation on synthetic trees with ground-truth dynamics, covering parameter recovery and responses to unseen forces.
- A pipeline that turns dynamic 3D reconstructions obtained from real monocular videos of plants into interactive, real-time simulators.

2 Related Work

Physical Priors and Differentiable Simulation. Physical models constrain ambiguous reconstruction problems, ranging from isometric non-rigid geometry (Parashar et al., 2017) to dynamic motion extrapolation via 3D Gaussians (Li et al., 2025) and rigid-body simulation (Liu et al., 2024b). A second direction performs physical system identification by backpropagating through differentiable simulators, such as gradSim (Jatavallabhula et al., 2021) and PAC-NeRF (Li et al., 2023), or SAFT for cloth (Stotko & Klein, 2025). However, these approaches incur high computational costs from iterative forward simulations. Closest to our work are vibration-based parameter-recovery methods. Visual Vibrometry extracts frequencies and mode shapes from video (Davis et al., 2015), while Visual Vibration Tomography extends this idea to heterogeneous 3D objects using volumetric FEM solvers (Feng et al., 2022). In contrast, we leverage tree branching topology, building analytical Euler–Bernoulli beam models directly from skeleton geometry to efficiently link vibration spectra to mechanical parameters.

Neural and Reduced Deformable Modeling. Deformable modeling spans classical modal dynamics (Pentland & Williams, 1989) to modern neural-physical hybrids. Recent frameworks couple neural representations with spring–mass systems or continuum mechanics via the Material Point Method, including PhysTwin (Jiang et al., 2025), PhysGaussian (Xie et al., 2024) and Spring-Gaus which embeds a spring–mass model directly into 3D Gaussian kernels (Zhong et al., 2024). Other approaches use video or foundation-model priors to infer or refine material properties, including PhysDreamer, Physics3D, DreamPhysics, and PhysFlow (Zhang et al., 2024; Liu et al., 2024a; Huang et al., 2025; Liu et al., 2025). To reduce simulation overhead, methods like Simplicits learn low-dimensional deformation bases (Modi et al., 2024). ModalNeRF extracts modal representations from observed scenes (Petitjean et al., 2023), while VoMP predicts volumetric mechanical properties from visual and geometric cues using a feed-forward model (Dagli et al., 2025). Unlike these approaches, our method derives a mechanically grounded modal basis tailored to slender branching structures, enabling zero-shot generalization to novel external forces without dense volumetric simulation or learned appearance priors.

Physics-Based Modeling of Trees. Prior work on tree dynamics falls into two lines. *Forward physical models* simulate trees with flexible structures (Shinya & Fournier, 1992), beam elements (Stam, 1997), or joint–spring links (Quigley et al., 2017; Murphy & Rudnicki, 2012). These models are efficient and physically grounded, but require the material parameters to be known. *Data-driven methods* instead generate plausible motion from learned priors, conditioned on a single image (Li et al., 2024) or on 3D tree geometry (Li et al., 2026b), without using any observed motion of the specific tree. Their motion is adapted to the given shape, but material-dependent quantities such as the frequency scale and damping cannot be inferred from shape alone, and thus default to what is typical of the training data. DynaBeam combines the two: it simulates an explicit beam structure, like forward models, but identifies its parameters from observed motion, so it can respond to forces never seen in the input.

3 Method

3.1 Method Overview

Our goal is to infer the physical properties that govern a plant's motion, namely its material and the forces acting on it, directly from the dynamic object, while keeping the approach computationally efficient and robust. Our key observation is that a plant's geometry already determines most of its dynamics. Once the branches are modeled as Euler–Bernoulli beams, the vibration mode shapes follow from the skeleton alone. The observed motion then only needs to supply what geometry cannot: a single material scale E/ρ, a damping ratio ζ, and the force $\mathbf{F}(t)/\rho$ that set the plant in motion. This turns physical inversion from a search over repeated simulations into a sequence of small fitting problems. And because the modes come from the structure rather than from the observed motion, the recovered model can respond to forces that excite modes never seen in the input.

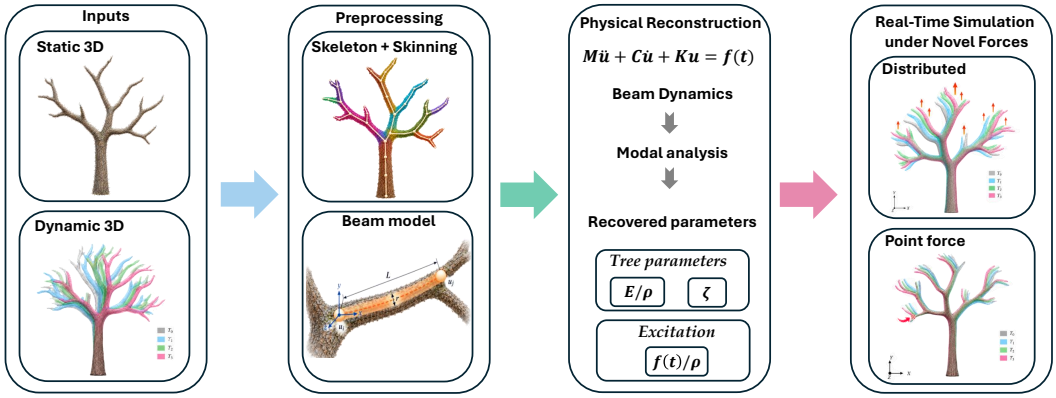


Figure 2: **Overview of DynaBeam.** From a static 3D representation and dynamic 3D observations, we extract a skeleton and skinning weights and build a beam-based physical model. Modal analysis recovers the stiffness-to-density ratio E/ρ, the damping ratio ζ, and the density-normalized force $\mathbf{F}(t)/\rho$. The recovered model can then be simulated in real time under novel distributed and point forces.

Concretely, DynaBeam takes a static 3D representation of the plant in its rest configuration (e.g., 3D Gaussians or a textured mesh) and a dynamic sequence of temporally corresponding 3D trajectories (Fig. 2). We extract a skeleton from the static geometry and build the beam model (Secs. 3.2 and 3.3). We then recover E/ρ by matching the observed frequencies to a precomputed reference spectrum, estimate ζ from the free decay, and fit the force that explains the remaining motion (Sec. 3.4). Finally, the recovered model can re-simulate the observed sequence or be driven by novel forces in real time (Sec. 3.5).

3.2 Preliminaries

Euler–Bernoulli Beam Structures. We model a slender structure, such as a plant, as a skeleton of N nodes connected by three-dimensional Euler–Bernoulli beam elements, which capture stretching, bending, and twisting. Each node has three translational and three rotational degrees of freedom, so the state is $\mathbf{u}(t) \in \mathbb{R}^{6N}$. We assume a linear isotropic material with Young's modulus E and a fixed Poisson ratio ν, to which the observed motion is largely insensitive (Zhang et al., 2024). The assembled stiffness matrix K is proportional to E, and the lumped mass matrix M is proportional to the density ρ. With the root clamped, the dynamics are

$$M\ddot{\mathbf{u}}(t) + C\dot{\mathbf{u}}(t) + K\mathbf{u}(t) = \mathbf{F}(t), \tag{1}$$

where C is the damping matrix, parameterized below by a modal damping ratio (Appendix C.1).

Modal Dynamics. Rather than tracking thousands of coupled nodal degrees of freedom, we represent the motion in the structure's natural vibration modes, given by $K\boldsymbol{\phi}_i = \omega_i^2 M\boldsymbol{\phi}_i$. With mass-normalized modes, the motion is approximated as $\mathbf{u}(t) \approx \Phi\mathbf{q}(t)$. Assuming a single damping ratio ζ shared by all modes, the system decouples into independent damped oscillators (Appendix C.1.),

$$\ddot{q}_i(t) + 2\zeta\omega_i\dot{q}_i(t) + \omega_i^2 q_i(t) = \boldsymbol{\phi}_i^\top \mathbf{F}(t). \tag{2}$$

3.3 Structural Initialization and Observations

Skeleton and Skinning. We build the beam model from a static 3D representation of the plant in its rest configuration, such as 3D Gaussians or a textured mesh. From it, we extract a skeleton consisting of node positions, connectivity, local branch radii, and a root node, which directly defines the beam elements of Sec. 3.2 (Appendix I). From the same representation, we compute skinning weights that bind each visual element, such as a Gaussian or a mesh vertex, to the skeleton. Because the visual and mechanical models share one rest configuration, any structural motion, whether observed or simulated, can be transferred to the full representation for rendering. Leaves and fine twigs not captured by the skeleton are bound to their nearest branch through the same skinning.

Dynamic 3D Observations. To infer the physical parameters, our method relies on a dynamic sequence providing temporally corresponding 3D trajectories, denoted $\mathbf{x}_p(t) \in \mathbb{R}^3$ for a set of observed points p. For each point, we define its time-varying displacement relative to its rest position $\mathbf{x}_p^0$, obtained from the static representation, as $\mathbf{y}_p(t) = \mathbf{x}_p(t) - \mathbf{x}_p^0$. We relate the stacked displacements $\mathbf{y}(t)$ linearly to the structural state, $\mathbf{y}(t) \approx H\mathbf{u}(t)$, where the observation operator H maps structural degrees of freedom to observed displacements. The observed points and the construction of H are described in Appendix E.

3.4 Physical Parameter and Force Recovery

We first establish which physical quantities can be recovered from motion alone.

Proposition 1 (Identifiable Physical Parameters)**.** *For a fixed structural geometry and mass discretization, the motion of the beam-based system depends on the material and the external force only through the stiffness-to-density ratio E/ρ and the density-normalized force $\widetilde{\mathbf{F}}(t) = \mathbf{F}(t)/\rho$. The absolute Young's modulus E, density ρ, and force $\mathbf{F}(t)$ cannot be uniquely identified from kinematic observations alone.*

The proof is given in Appendix C.2. Intuitively, a stiff, dense plant can vibrate at the same frequencies as a softer, lighter one, and a heavy plant pushed by a strong force can move exactly like a light plant pushed by a weak force. We therefore recover E/ρ and $\widetilde{\mathbf{F}}(t)$.

Material Recovery. Material recovery relies on the following scaling property of the beam model.

Proposition 2 (Spectral Scaling)**.** *For a fixed structural geometry, the natural frequencies of the system equal those of a reference model with $E/\rho = 1$, scaled by the global factor $\sqrt{E/\rho}$. The mode shapes do not depend on E/ρ.*

Proposition 2 (proof in Appendix C.3) means the eigenvalue problem needs to be solved only once, for a reference model. Recovering E/ρ then reduces to finding a single scale factor that stretches the reference frequencies onto the observed ones. We first extract the dominant vibration frequencies from the observed motion, together with the spatial pattern in which the plant vibrates at each one.

Because a given force may leave some modes unexcited, we do not simply pair observed and reference frequencies by their index. Instead we pair them primarily by the similarity of their vibration patterns, which is valid because mode shapes are unaffected by the material. Finally, we estimate the optimal global frequency scale $s^\star$ by a weighted least-squares fit over the matched observed and reference frequencies, giving more weight to strong, confidently matched peaks. Since $s = \sqrt{E/\rho}$, we obtain $\widehat{E/\rho} = (s^\star)^2$. Full details are given in Appendix D.1.

Modal Coordinates. By Proposition 2, the recovered model has the reference mode shapes, with frequencies $\hat{\omega}_i = s^\star \omega_i^{(0)}$. We keep only the modes that can be recovered from the observations: we discard modes above the Nyquist limit of the sequence, and modes whose projection onto the observed points, $\|H\boldsymbol{\phi}_i\|_2$, is negligible. For the remaining basis Φ, we estimate the modal coordinates at each frame by ridge regression,

$$\hat{\mathbf{q}}(t) = \arg\min_{\mathbf{q}} \|H\Phi\mathbf{q} - \mathbf{y}(t)\|_2^2 + \lambda\|\mathbf{q}\|_2^2. \tag{3}$$

The regularization stabilizes the solution when different modes produce similar motion at the observed points. Because the fit is performed independently at each frame, unavailable observations can simply be omitted (Appendix D.2).

Damping Recovery. We estimate ζ from a free-decay interval after the applied force has ended, where the amplitude decreases only because of intrinsic damping. We fit each mode with a decaying sinusoid, $\hat{q}_i(t) \approx c_i + e^{-\zeta\hat{\omega}_i t}\left[a_i \cos(\omega_{d,i} t) + b_i \sin(\omega_{d,i} t)\right]$, using mode-specific amplitudes a_i, b_i, a residual offset c_i, and a single ζ shared across all reliable modes (Appendix D.3).

Force Recovery. Having recovered the temporal coordinates $\hat{\mathbf{q}}(t)$, natural frequencies $\hat{\omega}_i$, and damping ratio $\hat{\zeta}$, we can directly reconstruct the external modal forces that drove the motion:

$$\widetilde{F}_{\mathrm{modal},i}(t) = \ddot{q}_i(t) + 2\hat{\zeta}\hat{\omega}_i \dot{q}_i(t) + \hat{\omega}_i^2 q_i(t). \tag{4}$$

By definition, this modal force is exactly the projection of the true physical force onto the mode shapes, meaning $\widetilde{\mathbf{F}}_{\text{modal}}(t) = \Phi^T \widetilde{\mathbf{F}}(t)$. While this fully captures how the internal skeleton was excited, we cannot simply invert Φ^T to recover the actual physical force $\widetilde{\mathbf{F}}(t)$ because our minimal active basis is vastly smaller than the skeleton's full $6N$ degrees of freedom.

To further solve for the actual $\widetilde{\mathbf{F}}(t)$, we must constrain the problem using prior environmental knowledge. We define an explicit basis of K spatial patterns, forming a matrix $V = [\widetilde{\mathbf{F}}_1, \ldots, \widetilde{\mathbf{F}}_K] \in \mathbb{R}^{6N \times K}$. For example, to model a global directional wind, we use $K = 3$ patterns representing uniform pushes along the X, Y, and Z translational degrees of freedom, with the rotational degrees of freedom padded with zeros.

By modeling the actual force as a combination of these patterns, $\widetilde{\mathbf{F}}(t) = V\mathbf{a}(t)$, and substituting this into our projection identity, we obtain $\Phi^T V \mathbf{a}(t) \approx \widetilde{\mathbf{F}}_{\text{modal}}(t)$. We can now stably compute the time-varying strength of these explicit force components by solving a linear least squares problem at each frame:

$$\hat{\mathbf{a}}(t) = \arg\min_{\mathbf{a}} \left\| \Phi^T V \mathbf{a} - \widetilde{\mathbf{F}}_{\text{modal}}(t) \right\|_2^2. \tag{5}$$

The total interpretable physical force is then recovered as $\widetilde{\mathbf{F}}(t) = V\hat{\mathbf{a}}(t)$. Finally, consistent with Proposition 1, this formulation naturally yields the density-normalized force, $\widetilde{\mathbf{F}}(t) = \mathbf{F}(t)/\rho$, which can be scaled to an absolute physical force if ρ is known.

3.5 Reconstruction and Novel-Force Simulation

Reconstruction. The coordinates $\hat{\mathbf{q}}(t)$ of Sec. 3.4 are fitted to the measurements independently at each frame, without any equations of motion; they describe the observed motion but cannot predict it. To test whether the recovered physics alone explains the observations, we instead generate new coordinates $\mathbf{q}_{\text{sim}}(t)$ by integrating the modal equations from rest under the recovered force, using Newmark-β time stepping:

$$\ddot{q}_{\text{sim},i}(t) + 2\hat{\zeta}\hat{\omega}_i \dot{q}_{\text{sim},i}(t) + \hat{\omega}_i^2 q_{\text{sim},i}(t) = \boldsymbol{\phi}_i^\top \widetilde{\mathbf{F}}(t). \tag{6}$$

The resulting motion $\mathbf{u}_{\text{sim}}(t) = \Phi \mathbf{q}_{\text{sim}}(t)$ uses only $\widehat{E/\rho}$, $\hat{\zeta}$, and $\widetilde{\mathbf{F}}(t)$, so its agreement with the observations tests the complete recovered model (Appendix D.6).

Novel-Force Simulation. Since $\widehat{E/\rho}$, $\hat{\zeta}$, and Φ do not depend on the observed force, we can replace $\widetilde{\mathbf{F}}(t)$ with a user-defined force $\widetilde{\mathbf{F}}_{\text{novel}}(t)$ that differs in location, direction, magnitude, or temporal profile (Appendix D.6). We project it onto the modes, $\boldsymbol{\phi}_i^\top \widetilde{\mathbf{F}}_{\text{novel}}(t)$, and integrate the same modal equations. Modes that were weakly excited in the observed sequence remain in Φ and respond whenever the new force has a nonzero projection onto them. Each step integrates only a few independent scalar oscillators, so simulation runs in real time. In both cases, the skeleton motion is transferred to the 3D Gaussians through the skinning weights for rendering.

4 Experiments

Data. We evaluate on a synthetic dataset of 72 branching tree structures spanning five botanical categories (aspen, oak, platanus, elm, and bald cypress). Each tree is created by generating an image with a text-to-image model and lifting it to a 3D Gaussian Splatting representation using TRELLIS (Xiang et al., 2025). The result is tetrahedralized and animated using our implementation of a high-resolution volumetric FEM simulation, with species-specific, spatially varying material parameters from the USDA Wood Handbook (Ross & USDA Forest Service., 2010; Simpson, 1993). Each tree is driven by a short distributed force pulse along $+x$, followed by free decay. This simulation supplies the inputs shared by all methods: the rest shape and temporally corresponding 3D trajectories of skeleton joints and Gaussian points. The FEM model itself is never available to any method. It also provides the ground-truth physical parameters and, for the generalization experiments, ground-truth responses to novel forces. Full details are given in Appendix F.

Baselines. For parameter recovery (Sec. 4.1), we compare against two alternatives. **VoMP** (Dagli et al., 2025) predicts material fields $E(x), \rho(x), \nu(x)$ feed-forward from the 3DGS geometry and appearance, without using the observed motion. For parameter recovery, we reduce its prediction to a scalar E/ρ; for reconstruction, we simulate its full predicted material field with the same FEM simulator used to generate the ground truth. **PhysFlow** (Liu et al., 2025) is an inverse-simulation method that repeatedly runs an MPM simulator and compares the result with the observations. We evaluate two variants: a *refinement* variant, which fixes $\rho = 1$ so that the recovered Young's modulus equals E/ρ and is initialized from the VoMP estimate without access to the force or damping, and an *oracle* variant (PhysFlow†), given the ground-truth density, force, and damping and a coarse sweep over E before refinement (Appendix K.2). For generalization to novel forces (Sec. 4.2), we compare against **ModalNeRF** (Petitjean et al., 2023), which extracts a modal basis directly from the observed motion. We re-implement its modal principle on our dynamic mesh representation so that the comparison isolates the choice of modal basis. DynaBeam is the only method that recovers the material ratio, damping, and external force jointly from the observed motion alone. Implementation details of DynaBeam are given in Appendix H, and those of the baselines, including required adaptations, in Appendix K.

We evaluate three questions: whether the observed motion suffices to recover the physical parameters and reproduce the observed dynamics (Sec. 4.1), whether the recovered model predicts the tree's response to novel, unseen forces (Sec. 4.2), and whether the pipeline transfers to real captured videos and supports interactive simulation (Sec. 4.3).

4.1 Physical Parameter Recovery

We first test whether the physical parameters that generated a sequence can be recovered from it. DynaBeam estimates a single homogeneous E/ρ, a damping ratio ζ shared across modes, and the time-varying coefficients of a low-dimensional basis of global translational forces (Sec. 3.4). The ground-truth E/ρ is the ratio of the spatially averaged FEM material fields. We measure parameter accuracy with the absolute log distance and absolute log relative errors (ALDE, ALRE). We measure reconstruction of the observed motion with trajectory relative L_2 error and cosine similarity, obtained by forward re-simulating each method's recovered model and evaluating on a held-out 20% of skeleton joints. Metric definitions are given in Appendix G.

Results. Table 1 summarizes the results. Among methods without ground-truth inputs, DynaBeam performs best on all metrics. It recovers E/ρ with a mean ALDE of 0.15, corresponding to a factor of roughly 1.4 from the ground truth, while also estimating damping and force, and reproduces the observed motion with a cosine similarity of 0.89. VoMP reaches similar parameter accuracy (ALDE 0.17), but its reconstruction is substantially worse (cosine 0.59, relative L_2 2.31), even though it is given the ground-truth damping and force. Because VoMP does not use the observed motion, its errors in E/ρ shift the natural frequencies, and the simulated motion drifts out of phase with the observation. The PhysFlow refinement variant, initialized from VoMP, degrades both the parameter estimate (ALDE 0.23) and the reconstruction (cosine 0.26), since without the true force and damping its objective does not match the observed dynamics. The oracle PhysFlow† serves as a practical upper bound on reconstruction accuracy: it is given the ground-truth density, force, and damping, and then performs expensive MPM-based inverse refinement. It achieves the lowest errors, but solves a considerably easier problem, since these quantities are supplied from ground truth and a coarse sweep over E locates the correct region before local refinement. It also requires 1.5–5 hours per tree, whereas DynaBeam runs in seconds.

4.2 Generalization to Novel Forces

We next test whether the recovered model predicts the response to forces not seen during reconstruction. We randomly select 22 trees from the dataset, each observed only under the distributed $+x$ force. From this single sequence, DynaBeam recovers its physical parameters and a single global force-calibration factor (Appendix D.5), and ModalNeRF extracts its modal basis (see Appendix K.3 for how novel forces are applied to each baseline). Both models are then kept fixed and driven by two novel forces: a distributed force orthogonal (90°) to the training direction, and a point force applied at a single node. The point-force location is chosen essentially arbitrarily: we only require that its response excites at least one mode not significantly excited in training, a condition satisfied by

Table 1: Physical-parameter recovery and reconstruction of the observed training motion, averaged over trees. Best result among methods without ground-truth inputs in bold. PhysFlow refines the VoMP estimate with $\rho = 1$ and no force or damping. †Oracle: given the ground-truth ρ, force, and damping, and initialized by a coarse sweep over E. The PhysFlow variants are averaged over 45 trees, due to their runtime; other methods over all 72.

Method	Recovers	Runtime ↓	ALRE ↓	ALDE ↓	Cosine ↑	Rel. L_2 ↓
VoMP	E/ρ	22.3 s	0.0239	0.166	0.590	2.31
PhysFlow	E/ρ	∼30 min	0.0338	0.235	0.255	9.02
DynaBeam (ours)	E/ρ, ζ, F/ρ	**2.1 s**	**0.0219**	**0.152**	**0.887**	**0.388**
PhysFlow†	E	1.5–5 h	0.00185	0.0128	0.995	0.0712

Table 2: Reconstruction and generalization over 22 trees, reported as median [Q1, Q3]. Best median per scenario in bold. Metrics are defined in Sec. 4.2 and Appendix G; target-mode energy is shown in Figure 3c.

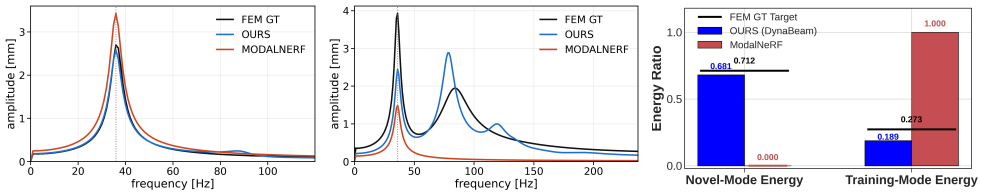
Scenario	Method	Rel. L_2 ↓	Cosine ↑	Modal Dist. ↓
Training	ModalNeRF	0.395 [0.252, 0.517]	**0.988** [0.980, 0.995]	**0.007** [0.001, 0.019]
	DynaBeam (ours)	**0.194** [0.141, 0.225]	0.981 [0.974, 0.990]	0.039 [0.016, 0.093]
Distributed new dir.	ModalNeRF	0.999 [0.993, 1.000]	0.056 [0.031, 0.121]	0.784 [0.423, 0.943]
	DynaBeam (ours)	**0.231** [0.178, 0.324]	**0.977** [0.956, 0.985]	**0.043** [0.015, 0.073]
Point force	ModalNeRF	1.014 [1.001, 1.177]	0.044 [0.011, 0.170]	0.848 [0.556, 0.968]
	DynaBeam (ours)	**0.450** [0.369, 0.738]	**0.905** [0.685, 0.951]	**0.099** [0.047, 0.225]

most candidate nodes (Appendix F). We compare their predictions against the FEM response to the same forces. Besides trajectory relative L_2 error and cosine similarity, we compare modal content by projecting the motion onto the first 30 FEM eigenmodes. *Modal distance error* is the total-variation distance between the predicted and ground-truth distributions over modes. *Target-mode energy* is the share of the predicted motion that lies in the modes dominating the ground-truth response, and is reported in Figure 3c. Full definitions are given in Appendix G.

Results. Table 2 reports the results. Both methods reconstruct the training motion well (median cosine similarity above 0.98), but under novel forces only DynaBeam remains accurate: its median cosine similarity is 0.977 for the distributed force and 0.905 for the point force, compared with 0.056 and 0.044 for ModalNeRF. Figure 3 shows why. Both methods match the ground truth at the dominant training frequency (Fig. 3a), but a novel point force excites an additional high-frequency peak, which DynaBeam reproduces and ModalNeRF misses (Fig. 3b). Across all 22 trees, ModalNeRF keeps nearly all of its motion in the training modes, whereas DynaBeam places a substantial share in the newly excited mode (Fig. 3c); Figure 4 shows the same effect in space. ModalNeRF's basis spans only the modes present in the training sequence, while DynaBeam computes its modes from the stiffness and mass matrices of the beam model, so modes absent from training can still respond to a new force.

4.3 Interactive Real-Time Simulation from Real Videos

Finally, we apply DynaBeam to real monocular videos of plants. The static geometry and 3D trajectories are reconstructed from the video using off-the-shelf segmentation, depth, image-to-3D, and point-tracking models, and the structural graph is built from a few manual first-frame annotations (Appendix J). DynaBeam then recovers the physical model exactly as in the synthetic experiments. Since no ground truth is available for real plants, we evaluate these results qualitatively: we drive the recovered models with novel point and distributed forces and render the response through the skinned Gaussians (Figure 1). Simulation runs at 60.3 fps on RTX 3090. The supplementary material includes videos and interactive HTML viewers in which users can apply forces to the reconstructed plants in real time.



(a) Training motion. (b) Novel point-force excitation. (c) Energy Distribution under Novel Point Force

Figure 3: Frequency and modal analysis of generalization. **(a, b)** Spectra for a representative tree. **(a)** Under the training force, all methods match the dominant frequency. **(b)** Under a novel point force, the FEM response shows an additional high-frequency peak that DynaBeam reproduces and ModalNeRF misses. **(c)** Median share of motion in the training modes and in the dominant newly excited mode over 22 trees: DynaBeam follows the FEM distribution, while ModalNeRF stays in the training modes.

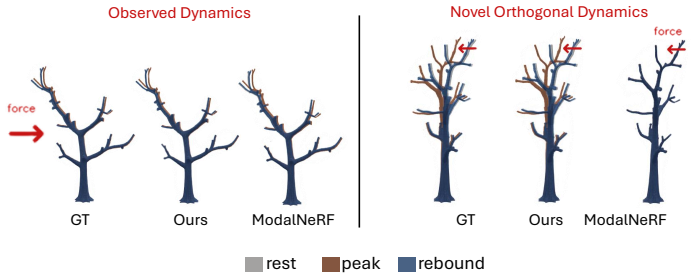


Figure 4: Generalization beyond the observed force. **Left:** response to the training force; both methods match the FEM ground truth. **Right:** responses to novel forces in orthogonal direction. ModalNeRF produces little motion along the newly excited directions, while DynaBeam follows the ground truth.

5 Limitations and Conclusion

Limitations. Our method relies on an accurate structural geometry, and errors in the recovered skeleton or branch dimensions can directly affect the inferred dynamics and may be absorbed into the recovered E/ρ (Appendix I). The Euler–Bernoulli formulation further assumes slender structures undergoing small-to-moderate elastic deformations, and may become inaccurate under large deformations or other nonlinear effects. We additionally assume a homogeneous isotropic material, which does not capture spatially varying or direction-dependent mechanical properties. The observed dynamics also identify only the ratio E/ρ, rather than E and ρ independently, unless one of them is known from additional information. Finally, foliage is carried by the branches through skinning, which captures the dominant branch dynamics, including in our leafy real-world captures; modeling leaf dynamics and the effect of dense foliage on the vibration modes is a natural direction for future work.

Conclusion. We presented DynaBeam, a framework that recovers an explicit physical model of vegetation from its branching structure and motion. By modeling plants as graphs of Euler–Bernoulli beams, the vibration modes follow from the geometry alone, and the observed motion only needs to determine the stiffness-to-density ratio, the damping ratio, and the density-normalized force. This turns physical inversion into a few small fitting problems that run in seconds. Because the modes are derived from the structure rather than extracted from the motion, the recovered model responds more accurately than motion-derived modal bases to forces that excite modes never seen in the input. Applied to reconstructions from real videos, DynaBeam turns captured plants into interactive objects that can be driven by new forces in real time. More broadly, our results suggest that exploiting the structure of an object class, rather than relying on general-purpose simulation, can make physical inversion both fast and generalizable.

AI USE STATEMENT

Generative AI tools were used in this work for three purposes: to improve the writing, to support the search for related literature, and to generate part of our synthetic dataset. They were not used to develop the method, design the experiments, or derive the theoretical results. For writing, language models were applied to text drafted by the authors, to improve grammar, clarity, and concision, including in technical descriptions and figure captions; all scientific content, analysis, and conclusions are the authors' own. For literature discovery, AI tools helped surface potentially relevant papers, and every cited work was then read and verified by the authors against the original publication. For dataset construction, an image-generation model produced the initial tree images, which were then lifted to 3D with TRELLIS (Xiang et al., 2025), and two language models were used to screen the generated images for species consistency (Appendix F). All subsequent steps, including skeleton extraction, material assignment, FEM simulation, and evaluation, follow the pipeline described in the paper. The authors reviewed all AI-assisted content, and all reported results come from our own experimental runs. The authors remain accountable for everything presented in this paper, including any text, claims, or artifacts created with the help of generative AI.

REPRODUCIBILITY STATEMENT

The beam model, modal formulation, and recovery procedure are specified in Sec. 3.2–3.5. The full beam derivation is given in Appendix B, and complete proofs of both propositions in Appendices C.2 and C.3. Each recovery step, including spectral processing and mode matching, modal-coordinate estimation, damping and force recovery, force calibration, and forward re-simulation, is detailed in Appendices D.1–D.6, with numerical settings in Appendix H. Skeleton extraction and skinning are described in Appendix I. The synthetic dataset, including tree generation, material assignment, and FEM ground-truth simulation, is described in Appendix F, and all metrics are defined in Appendix G. Baseline implementations and their required adaptations are described in Appendix K, and the real-video reconstruction pipeline in Appendix J. DynaBeam involves no learned components or random initialization, so its results are deterministic given the input. We will release the code, the synthetic dataset, and the interactive viewers upon publication.

REFERENCES

Oliver A Bauchau and James I Craig. Euler-bernoulli beam theory. In *Structural analysis*, pp. 173–221. Springer, 2009.

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.

Abe Davis, Katherine L Bouman, Justin G Chen, Michael Rubinstein, Fredo Durand, and William T Freeman. Visual vibrometry: Estimating material properties from small motion in video. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pp. 5335–5343, 2015.

Berthy T Feng, Alexander C Ogren, Chiara Daraio, and Katherine L Bouman. Visual vibration tomography: Estimating interior material properties from monocular video. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16210–16219. IEEE, 2022.

Adam W Harley, Yang You, Xinglong Sun, Yang Zheng, Nikhil Raghuraman, Yunqi Gu, Sheldon Liang, Wen-Hsuan Chu, Achal Dave, Suya You, et al. Alltracker: Efficient dense point tracking at high resolution. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5253–5262. IEEE, 2025.

Yixin Hu, Teseo Schneider, Bolun Wang, Denis Zorin, and Daniele Panozzo. Fast tetrahedral meshing in the wild. *ACM Transactions on Graphics (ToG)*, 39(4):117–1, 2020.

Tianyu Huang, Haoze Zhang, Yihan Zeng, Zhilu Zhang, Hui Li, Wangmeng Zuo, and Rynson WH Lau. Dreamphysics: Learning physics-based 3d dynamics with video diffusion priors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 3733–3741, 2025.

Krishna Murthy Jatavallabhula, Miles Macklin, Florian Golemo, Vikram Voleti, Linda Petrini, Martin Weiss, Breandan Considine, Jérôme Parent-Lévesque, Kevin Xie, Kenny Erleben, Liam Paull, Florian Shkurti, Derek Nowrouzezahrai, and Sanja Fidler. gradsim: Differentiable simulation for system identification and visuomotor control. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=c_E8kFWfhp0.

Hanxiao Jiang, Hao-Yu Hsu, Kaifeng Zhang, Hsin-Ni Yu, Shenlong Wang, and Yunzhu Li. Phystwin: Physics-informed reconstruction and simulation of deformable objects from videos. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7219–7230. IEEE, 2025.

Cornelius Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *Journal of research of the National Bureau of Standards*, 45(4): 255–282, 1950.

Dong-Yang Li, Wang Zhao, Yuxin Chen, Wenbo Hu, Meng-Hao Guo, Fang-Lue Zhang, Ying Shan, and Shi-Min Hu. Pixal3d: Pixel-aligned 3d generation from images. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–12, 2026a.

Jinxi Li, Ziyang Song, and Bo Yang. Trace: Learning 3d gaussian physical dynamics from multi-view videos. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8820–8829. IEEE, 2025.

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. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=tVkrbkz42vc.

Yaokun Li, Lihe Ding, Xiao Chen, Guang Tan, and Tianfan Xue. Dynamictree: Interactive real tree animation via sparse voxel spectrum. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1241–1251, 2026b.

Zhengqi Li, Richard Tucker, Noah Snavely, and Aleksander Holynski. Generative image dynamics. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24142–24153, 2024. doi: 10.1109/CVPR52733.2024.02279.

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

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

Zhuoman Liu, Weicai Ye, Yan Luximon, Pengfei Wan, and Di Zhang. Unleashing the potential of multi-modal foundation models and video diffusion for 4d dynamic physical scene simulation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11016–11025. IEEE, 2025.

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.

Kevin D Murphy and Mark Rudnicki. A physics-based link model for tree vibrations. *American Journal of Botany*, 99(12):1918–1929, 2012.

Shaifali Parashar, Daniel Pizarro, and Adrien Bartoli. Isometric non-rigid shape-from-motion with riemannian geometry solved in linear time. *IEEE transactions on pattern analysis and machine intelligence*, 40(10):2442–2454, 2017.

Alex Pentland and John Williams. Good vibrations: Modal dynamics for graphics and animation. In *Proceedings of the 16th annual conference on Computer graphics and interactive techniques*, pp. 215–222, 1989.

Automne Petitjean, Yohan Poirier-Ginter, Ayush Tewari, Guillaume Cordonnier, and George Drettakis. Modalnerf: Neural modal analysis and synthesis for free-viewpoint navigation in dynamically vibrating scenes. In *Computer Graphics Forum*, volume 42, pp. e14888. Wiley Online Library, 2023.

Ed Quigley, Yue Yu, Jingwei Huang, Winnie Lin, and Ronald Fedkiw. Real-time interactive tree animation. *IEEE transactions on visualization and computer graphics*, 24(5):1717–1727, 2017.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. In *International Conference on Learning Representations*, volume 2025, pp. 28085–28128, 2025.

Robert J. Ross and Forest Products Laboratory USDA Forest Service. *Wood handbook: wood as an engineering material*. 2010. doi: 10.2737/fpl-gtr-190. URL http://dx.doi.org/10.2737/FPL-GTR-190.

Mikio Shinya and Alain Fournier. Stochastic motion—motion under the influence of wind. In *Computer graphics forum*, volume 11, pp. 119–128. Wiley Online Library, 1992.

W. T. Simpson. *Specific gravity, moisture content, and density relationship for wood*. 1993. doi: 10.2737/fpl-gtr-76. URL http://dx.doi.org/10.2737/FPL-GTR-76.

Jos Stam. Stochastic dynamics: Simulating the effects of turbulence on flexible structures. In *Computer Graphics Forum*, volume 16, pp. C159–C164. Wiley Online Library, 1997.

David Stotko and Reinhard Klein. Saft: Shape and appearance of fabrics from template via differentiable physical simulations from monocular video. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 27660–27670. IEEE, 2025.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. VGGT-Ω. *arXiv preprint arXiv:2605.15195*, 2026.

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

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21469–21480. IEEE, 2025.

Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4389–4398. IEEE, 2024.

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*, pp. 388–406. Springer, 2024.

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*, pp. 407–423. Springer, 2024.

A Appendix Overview

Appendix B details the Euler–Bernoulli beam model; Appendix C the modal representation, damping, and proofs of Propositions 1–2; Appendix D the parameter and force recovery procedure; Appendix E the observation model; Appendix F dataset construction; Appendix G evaluation metrics; Appendix H implementation details; Appendix I skeleton extraction; Appendix J reconstruction from real videos; and Appendix K baseline details.

B EULER–BERNOULLI BEAM MODEL

This appendix summarizes the mechanical formulation used to construct the beam model underlying our reconstruction and simulation framework, including the beam interpolation, element stiffness, coordinate transformation, global assembly, and lumped mass model.

B.1 STRUCTURAL REPRESENTATION AND DEGREES OF FREEDOM

The tree skeleton is represented as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where nodes correspond to branch junctions or structural samples and edges to elastic beam segments. Let $\bar{\mathbf{x}}_i \in \mathbb{R}^3$ denote the reference position of node i. Each node has six degrees of freedom,

$$\mathbf{u}_i = [u_{x,i} \quad u_{y,i} \quad u_{z,i} \quad \theta_{x,i} \quad \theta_{y,i} \quad \theta_{z,i}]^\top,$$

and the global state is $\mathbf{u} \in \mathbb{R}^{6N}$.

For an edge $e = (i, j)$,

$$L_e = \|\bar{\mathbf{x}}_j - \bar{\mathbf{x}}_i\|_2, \qquad \mathbf{e}_{x,e} = \frac{\bar{\mathbf{x}}_j - \bar{\mathbf{x}}_i}{L_e}, \tag{7}$$

with the remaining local axes spanning the cross-section. We assume a linear, isotropic material and the standard Euler–Bernoulli assumptions: small elastic deformations, slender beams, and cross-sections that remain planar and perpendicular to the beam centerline during bending (Bauchau & Craig, 2009).

B.2 CROSS-SECTIONAL QUANTITIES

For edge e, let A_e be the cross-sectional area, $I_{y,e}$ and $I_{z,e}$ the second moments of area, and J_e the polar second moment. For a general cross-section $\mathcal{A}_e$,

$$A_e = \int_{\mathcal{A}_e} dA, \qquad I_{y,e} = \int_{\mathcal{A}_e} z^2\, dA, \qquad I_{z,e} = \int_{\mathcal{A}_e} y^2\, dA, \qquad J_e = I_{y,e} + I_{z,e}. \tag{8}$$

For a circular cross-section of radius r_e,

$$A_e = \pi r_e^2, \qquad I_{y,e} = I_{z,e} = \frac{\pi r_e^4}{4}, \qquad J_e = \frac{\pi r_e^4}{2}. \tag{9}$$

For an isotropic material,

$$G = \frac{E}{2(1+\nu)}.$$

In our experiments, ν is fixed, so recovering E/ρ also determines G/ρ.

B.3 EULER–BERNOULLI INTERPOLATION

For transverse displacement $w(x)$, Euler–Bernoulli theory gives

$$\frac{d^2}{dx^2}\left(EI\frac{d^2 w}{dx^2}\right) = q(x).$$

Within each element, external loads are represented by equivalent nodal forces, so $q(x) = 0$ in the element interior. For constant E and I, this implies $w^{(4)}(x) = 0$, and the exact element displacement is cubic.

Using $\xi = x/L \in [0, 1]$, we write

$$w(\xi) = H_1(\xi)w_1 + H_2(\xi)\theta_1 + H_3(\xi)w_2 + H_4(\xi)\theta_2,$$

with the cubic Hermite basis

$$H_1(\xi) = 1 - 3\xi^2 + 2\xi^3, \quad H_2(\xi) = L(\xi - 2\xi^2 + \xi^3),$$
$$H_3(\xi) = 3\xi^2 - 2\xi^3, \qquad H_4(\xi) = L(-\xi^2 + \xi^3).$$

The cubic Hermite shape functions are obtained by starting from a general cubic polynomial and enforcing the nodal interpolation conditions on displacement and slope at the two beam endpoints. These functions interpolate the endpoint transverse translations and slopes exactly; the factors of L in H_2 and H_4 convert nodal rotations into transverse displacement. The same interpolation is used independently in the two local bending planes.

B.4 Element Stiffness from Strain Energy

The axial, torsional, and bending energies yield the corresponding element stiffness terms. For axial deformation and torsion,

$$k_{\text{axial}} = \frac{EA}{L}, \qquad k_{\text{torsion}} = \frac{GJ}{L}.$$

For bending,

$$U_{\text{bend}} = \frac{1}{2}\int_0^L EI\left(\frac{d^2w}{dx^2}\right)^2 dx.$$

Substituting $w(x) = \sum_{k=1}^{4} H_k(x)q_k$ gives

$$k_{ij} = \int_0^L EI\, H_i''(x)H_j''(x)\, dx.$$

Evaluating these integrals produces the four characteristic bending coefficients

$$\frac{12EI}{L^3}, \qquad \frac{6EI}{L^2}, \qquad \frac{4EI}{L}, \qquad \frac{2EI}{L},$$

corresponding to transverse translation, translation–rotation coupling, self-rotation, and cross-node rotation, respectively.

B.5 Three-Dimensional Element Stiffness Matrix

For element $e = (i, j)$, the local degrees of freedom are ordered as

$$\mathbf{u}_{\text{loc}}^{(e)} = [\ u_i \quad v_i \quad w_i \quad \theta_{x,i} \quad \theta_{y,i} \quad \theta_{z,i} \quad u_j \quad v_j \quad w_j \quad \theta_{x,j} \quad \theta_{y,j} \quad \theta_{z,j}\ ]^\top.$$

Define

$$a = \frac{EA}{L}, \qquad t = \frac{GJ}{L},$$
$$b_y = \frac{12EI_z}{L^3}, \quad c_y = \frac{6EI_z}{L^2}, \quad d_y = \frac{4EI_z}{L}, \quad e_y = \frac{2EI_z}{L},$$
$$b_z = \frac{12EI_y}{L^3}, \quad c_z = \frac{6EI_y}{L^2}, \quad d_z = \frac{4EI_y}{L}, \quad e_z = \frac{2EI_y}{L}.$$

The local 12×12 stiffness matrix is

$$\mathbf{K}_{\text{loc}}^{(e)} = \begin{bmatrix}
a & 0 & 0 & 0 & 0 & 0 & -a & 0 & 0 & 0 & 0 & 0 \\
0 & b_y & 0 & 0 & 0 & c_y & 0 & -b_y & 0 & 0 & 0 & c_y \\
0 & 0 & b_z & 0 & -c_z & 0 & 0 & 0 & -b_z & 0 & -c_z & 0 \\
0 & 0 & 0 & t & 0 & 0 & 0 & 0 & 0 & -t & 0 & 0 \\
0 & 0 & -c_z & 0 & d_z & 0 & 0 & 0 & c_z & 0 & e_z & 0 \\
0 & c_y & 0 & 0 & 0 & d_y & 0 & -c_y & 0 & 0 & 0 & e_y \\
-a & 0 & 0 & 0 & 0 & 0 & a & 0 & 0 & 0 & 0 & 0 \\
0 & -b_y & 0 & 0 & 0 & -c_y & 0 & b_y & 0 & 0 & 0 & -c_y \\
0 & 0 & -b_z & 0 & c_z & 0 & 0 & 0 & b_z & 0 & c_z & 0 \\
0 & 0 & 0 & -t & 0 & 0 & 0 & 0 & 0 & t & 0 & 0 \\
0 & 0 & -c_z & 0 & e_z & 0 & 0 & 0 & c_z & 0 & d_z & 0 \\
0 & c_y & 0 & 0 & 0 & e_y & 0 & -c_y & 0 & 0 & 0 & d_y
\end{bmatrix}.$$

For circular cross-sections, $I_y = I_z = I$, so the two bending planes share the same coefficients. The matrix is symmetric, as required by conservation of elastic energy.

B.6 Local-to-Global Transformation and Assembly

Let $\mathbf{R}_e \in SO(3)$ map global vectors into the local coordinate frame of element e, and define

$$\mathbf{T}_e = \text{diag}(\mathbf{R}_e, \mathbf{R}_e, \mathbf{R}_e, \mathbf{R}_e).$$

Then

$$\mathbf{K}_{\text{global}}^{(e)} = \mathbf{T}_e^\top \mathbf{K}_{\text{loc}}^{(e)} \mathbf{T}_e.$$

If $\mathbf{S}_e \in \{0,1\}^{12\times 6N}$ selects the endpoint degrees of freedom of edge e, the global stiffness matrix is

$$\mathbf{K} = \sum_{e\in\mathcal{E}} \mathbf{S}_e^\top \mathbf{K}_{\text{global}}^{(e)} \mathbf{S}_e.$$

In implementation, this superposition is performed using sparse indexed accumulation (scatter-add), without explicitly constructing $\mathbf{S}_e$.

B.7 Lumped Mass Matrix and Root Boundary Condition

We use a lumped-mass approximation. For element e,

$$m_e = \rho A_e L_e,$$

and half of its translational mass is assigned to each endpoint. Thus,

$$m_i = \frac{1}{2} \sum_{e\in\mathcal{E}(i)} m_e.$$

Each node also carries a diagonal rotational inertia $\mathbf{J}_i = \text{diag}(J_{x,i}, J_{y,i}, J_{z,i})$, obtained by accumulating the contributions of its incident elements. The nodal and global mass matrices are therefore

$$\mathbf{M}_i = \text{diag}(m_i, m_i, m_i, J_{x,i}, J_{y,i}, J_{z,i}), \qquad \mathbf{M} = \text{blkdiag}(\mathbf{M}_1, \ldots, \mathbf{M}_N).$$

For a fixed geometry, the mass moments of inertia are proportional to the element mass, since

$$I = \int r^2 \, dm = \rho \int r^2 \, dV.$$

The geometric integral is fixed, while $m_e = \rho V_e$; hence both the element mass and its mass moments of inertia scale linearly with the material density ρ.

The root node r is clamped,

$$\mathbf{u}_r = \mathbf{0},$$

so its six degrees of freedom are removed from the dynamic system. The resulting constrained matrices $\mathbf{K}_c$ and $\mathbf{M}_c$ define the linear elastic model used throughout the paper. Their modal decomposition and the dependence of the eigenfrequencies on E/ρ are derived in Appendix C.

C Modal Representation and Material Scaling

This appendix provides the theoretical and computational details underlying the modal formulation used in our method. We first define the reduced modal representation of the Euler–Bernoulli beam system. We then prove Proposition 1, concerning the identifiability of the density-normalized physical quantities, and Proposition 2, concerning the spectral scaling with E/ρ and the invariance of the mode shapes. Building on these results, we describe how the reference spectrum is used to recover E/ρ, and conclude with the modal truncation and numerical eigenproblem computation used to construct the reduced basis.

C.1 Modal Representation

Starting from the constrained mass and stiffness matrices $\mathbf{M}_c$ and $\mathbf{K}_c$ derived from the Euler–Bernoulli beam model in Appendix B, the natural vibration modes are defined by the generalized eigenvalue problem

$$\mathbf{K}_c \boldsymbol{\phi}_i = \omega_i^2 \mathbf{M}_c \boldsymbol{\phi}_i, \tag{10}$$

where ω_i is the i-th natural angular frequency, $\boldsymbol{\phi}_i$ is its corresponding mode shape, and

$$f_i = \frac{\omega_i}{2\pi} \tag{11}$$

is the corresponding natural frequency.

Since $\mathbf{K}_c$ and $\mathbf{M}_c$ are symmetric and $\mathbf{M}_c$ is positive definite, the modes can be chosen mass-orthonormal:

$$\boldsymbol{\phi}_i^\top \mathbf{M}_c \boldsymbol{\phi}_j = \delta_{ij}. \tag{12}$$

Collecting the retained modes in

$$\boldsymbol{\Phi} = [\boldsymbol{\phi}_1 \quad \cdots \quad \boldsymbol{\phi}_r], \tag{13}$$

gives

$$\boldsymbol{\Phi}^\top \mathbf{M}_c \boldsymbol{\Phi} = \mathbf{I}, \qquad \boldsymbol{\Phi}^\top \mathbf{K}_c \boldsymbol{\Phi} = \boldsymbol{\Omega}^2, \tag{14}$$

where

$$\boldsymbol{\Omega} = \mathrm{diag}(\omega_1, \ldots, \omega_r). \tag{15}$$

The displacement is represented in this reduced basis as

$$\mathbf{u}(t) \approx \boldsymbol{\Phi}\mathbf{q}(t), \tag{16}$$

where $\mathbf{q}(t)$ contains the time-varying modal coordinates.

Damping. The full equation of motion (Sec. 3.2) contains a damping term $C\dot{\mathbf{u}}(t)$. We do not model the damping matrix C explicitly. Instead, when transforming to modal coordinates, we assume that damping acts independently on each mode and is described by a single dimensionless damping ratio ζ, shared by all retained modes. This is a modelling assumption that keeps the modal equations decoupled and adds only one scalar parameter. Substituting $\mathbf{u}(t) \approx \Phi\mathbf{q}(t)$ and projecting onto the modal basis, the dynamics become independent damped oscillators,

$$\ddot{q}_i(t) + 2\zeta\omega_i\dot{q}_i(t) + \omega_i^2 q_i(t) = F_i^{\mathrm{modal}}(t), \qquad F_i^{\mathrm{modal}}(t) = \boldsymbol{\phi}_i^\top \mathbf{F}(t). \tag{17}$$

The estimation of ζ from observed motion is described in Appendix D.3.

C.2 Proof of Proposition 1

Proof. We begin from the Euler–Bernoulli beam model derived in Appendix B. For a fixed structural geometry, discretizations, boundary conditions, and Poisson ratio, every entry of the element stiffness matrix derived in Appendix B.5 is proportional to Young's modulus E. Since the shear modulus satisfies $G = E/[2(1+\nu)]$, fixing ν also makes the torsional stiffness proportional to E. The local-to-global transformation and assembly described in Appendices B.6 preserve this proportionality. Therefore, the global stiffness matrix can be written as

$$K(E) = E\,\bar{K}, \tag{18}$$

where $\bar{K}$ is independent of E.

Similarly, as shown in Appendix B.7, the translational mass terms are proportional to the material density through $m = \rho V$. The rotational terms are mass moments of inertia, which, as shown in the same appendix, are also proportional to ρ for a fixed geometry. Hence, every entry of the assembled mass matrix scales linearly with ρ, and we may write

$$M(\rho) = \rho\,\bar{M}, \tag{19}$$

where $\bar{M}$ is independent of ρ.

Consider first the undamped forced equation of motion,

$$M(\rho)\ddot{\mathbf{u}}(t) + K(E)\mathbf{u}(t) = \mathbf{F}(t). \tag{20}$$

Substituting the factorizations above gives

$$\rho\bar{M}\ddot{\mathbf{u}}(t) + E\bar{K}\mathbf{u}(t) = \mathbf{F}(t). \tag{21}$$

Dividing by ρ yields

$$\bar{M}\ddot{\mathbf{u}}(t) + \frac{E}{\rho}\bar{K}\mathbf{u}(t) = \frac{\mathbf{F}(t)}{\rho}. \tag{22}$$

Thus, for a fixed geometry, the kinematic response depends on the material and external force only through the stiffness-to-density ratio E/ρ and the density-normalized force $\widetilde{\mathbf{F}}(t) = \mathbf{F}(t)/\rho$.

Damping is introduced at the modal level through a single dimensionless ratio ζ (Appendix C.1). The modes and natural frequencies used in this modal projection are those of the normalized system in Eq. 22, whose eigenproblem $\frac{E}{\rho}\bar{K}\boldsymbol{\phi}_i = \omega_i^2 \bar{M}\boldsymbol{\phi}_i$ depends on the material only through E/ρ. The modal damping term $2\zeta\omega_i$ therefore also depends only on E/ρ and the dimensionless ζ, so the damping model introduces no additional absolute material parameter, and the identifiability argument below is unchanged.

This induces a global scaling ambiguity. For any $\alpha > 0$, consider

$$E' = \alpha E, \qquad \rho' = \alpha\rho, \qquad \mathbf{F}'(t) = \alpha\mathbf{F}(t). \tag{23}$$

Then

$$\frac{E'}{\rho'} = \frac{E}{\rho}, \qquad \frac{\mathbf{F}'(t)}{\rho'} = \frac{\mathbf{F}(t)}{\rho}. \tag{24}$$

The normalized equations of motion are therefore unchanged, and the two parameter sets produce exactly the same kinematic trajectory $\mathbf{u}(t)$ under the same initial conditions.

Consequently, kinematic observations alone cannot distinguish between the family of physically scaled systems

$$(E, \rho, \mathbf{F}(t)) \sim (\alpha E, \alpha\rho, \alpha\mathbf{F}(t)), \qquad \alpha > 0. \tag{25}$$

Consequently, the absolute Young's modulus E, density ρ, and force $\mathbf{F}(t)$ are not uniquely identifiable from kinematic observations, whereas the normalized quantities E/ρ and $\widetilde{\mathbf{F}}(t) = \mathbf{F}(t)/\rho$ are invariant under this scaling ambiguity.

□

C.3 Proof of Proposition 2

Proof. As established in the proof of Proposition 1, for a fixed geometry and discretization the system matrices factor as $K(E) = E\bar{K}$ and $M(\rho) = \rho\bar{M}$.

Substituting these factorizations into the generalized eigenvalue problem defined in Appendix C.1 gives

$$E\bar{K}\phi_i = \omega_i^2 \rho\bar{M}\phi_i. \tag{26}$$

Dividing by E yields

$$\bar{K}\phi_i = \left(\omega_i^2 \frac{\rho}{E}\right)\bar{M}\phi_i. \tag{27}$$

Now consider the unit-material reference model used in our method, defined by $E/\rho = 1$. Its generalized eigenvalue problem is

$$\bar{K}\phi_i = \left(\omega_i^{(0)}\right)^2 \bar{M}\phi_i, \tag{28}$$

where $\omega_i^{(0)}$ is the i-th reference natural frequency.

Comparing the two eigenvalue problems shows that they have the same eigenvectors ϕ_i, while their eigenvalues satisfy

$$\omega_i^2 \frac{\rho}{E} = \left(\omega_i^{(0)}\right)^2. \tag{29}$$

Therefore,

$$\omega_i = \sqrt{\frac{E}{\rho}}\,\omega_i^{(0)}. \tag{30}$$

Since $f_i = \omega_i/(2\pi)$,

$$f_i = \sqrt{\frac{E}{\rho}}\,f_i^{(0)}. \tag{31}$$

Thus, changing E/ρ applies a single global scale factor $\sqrt{E/\rho}$ to the natural-frequency spectrum while leaving the spatial mode shapes unchanged.

□

C.4 Material Recovery from the Reference Spectrum

Proposition 2 allows us to solve the structural eigenproblem only once, for the unit-material reference model $E/\rho = 1$. For any material ratio, the natural frequencies are then $f_i = s f_i^{(0)}$ with $s = \sqrt{E/\rho}$. Material recovery therefore reduces to estimating a single global scale factor $s^\star$ that aligns the reference frequencies with the observed ones, giving $\widehat{E/\rho} = (s^\star)^2$. The extraction of observed frequencies, their matching to structural modes, and the weighted fit of $s^\star$ are described in Appendix D.

C.5 Modal Truncation and Eigenproblem Computation

The complete system contains one mode for every unconstrained degree of freedom, but only a reduced subset is required in practice. The motivation can be seen from the static response of an individual mode:

$$\omega_i^2 q_i = F_i^{\text{modal}} \qquad \Longrightarrow \qquad q_i = \frac{F_i^{\text{modal}}}{\omega_i^2}. \tag{32}$$

For comparable modal-force magnitudes, high-frequency modes therefore produce smaller displacements and typically contribute less to the large-scale visible motion. The final active subset additionally depends on observability, since a mode may be poorly excited or weakly represented in the measured trajectories. The practical selection of the active modes is described in Appendix D.

Numerically, the generalized eigenvalue problem defined in Appendix C.1 is converted into a standard symmetric eigenproblem. Since $\mathbf{M}_c$ is symmetric positive definite, we use its Cholesky factorization

$$\mathbf{M}_c = \mathbf{L}\mathbf{L}^\top. \tag{33}$$

Defining

$$\mathbf{y} = \mathbf{L}^\top \boldsymbol{\phi} \tag{34}$$

gives

$$\underbrace{\mathbf{L}^{-1}\mathbf{K}_c\mathbf{L}^{-\top}}_{\widetilde{\mathbf{K}}}\mathbf{y} = \omega^2 \mathbf{y}, \tag{35}$$

after which the physical mode is recovered as

$$\boldsymbol{\phi} = \mathbf{L}^{-\top}\mathbf{y}. \tag{36}$$

Because only the lowest part of the spectrum is required, we compute a reduced set of eigenpairs using sparse Lanczos/Krylov methods rather than diagonalizing the full system (Lanczos, 1950). The resulting modal basis is reused during material recovery and forward simulation.

D Physical Parameter Recovery from Observed Motion

This appendix provides the implementation and mathematical details of the recovery procedure summarized in the main text. Starting from temporally corresponding 3D trajectories, the goal is to recover the intrinsic dynamical parameters of the beam model and the excitation that generated the observed motion. The procedure consists of four main stages: identifying the vibration frequencies and their corresponding structural modes, recovering the material ratio E/ρ, estimating the temporal modal coordinates and damping, and finally recovering an interpretable density-normalized force. We then re-simulate the recovered system to verify that these quantities are sufficient to reproduce the observed dynamics.

D.1 Spectral Analysis and Material Recovery

As defined in the main text, the observed trajectory of point p is $\mathbf{x}_p(t) \in \mathbb{R}^3$, and its displacement relative to the rest configuration is

$$\mathbf{y}_p(t) = \mathbf{x}_p(t) - \mathbf{x}_p^0. \tag{37}$$

The first step is to determine which natural vibration frequencies of the structure are visible in these trajectories. Directly analyzing individual tracks is unreliable because different points may move

with very different amplitudes and phases. A point close to the root, for example, may move only slightly even when the tree undergoes a strong global oscillation, while a branch tip may undergo a much larger displacement. We therefore preprocess the trajectories and combine their spectral evidence.

Invalid observations are removed and short temporal gaps are interpolated. Each scalar coordinate trajectory is linearly detrended to suppress slow drift that would otherwise produce strong energy near zero frequency. We then normalize each coordinate by its temporal RMS. This prevents a small number of large-displacement points from dominating the global spectrum and allows different tracked locations to contribute more evenly.

Before computing the Fourier transform, we apply a Hann window. Since the observed sequence contains only a finite temporal interval, its endpoints generally do not coincide with complete oscillation periods. A sharp truncation would therefore spread energy from a true frequency into neighboring frequency bins. The window reduces this spectral leakage and makes the dominant peaks easier to identify.

For a sequence of duration T, sampled at frequency f_s, the relevant temporal limits are

$$\Delta f \simeq \frac{1}{T}, \qquad f_{\mathrm{Nyquist}} = \frac{f_s}{2}. \tag{38}$$

The first quantity determines the physical frequency resolution of the sequence, while the second determines the highest frequency that can be represented without temporal aliasing. We do not rely on zero-padding for frequency estimation: although it produces a smoother interpolated spectrum, it does not increase the information contained in a sequence of fixed duration.

We compute a Fourier transform for each coordinate and aggregate the resulting spectral power,

$$P(f) = \sum_{p,c} \left| \mathcal{F}[y_{p,c}(t)] \right|^2, \tag{39}$$

where $c \in \{x, y, z\}$ and $\mathcal{F}$ denotes the Fourier transform. Importantly, we sum *power* rather than the complex Fourier coefficients themselves. Different parts of the structure may oscillate at the same frequency with opposite phases; summing complex coefficients could therefore cancel a physically strong vibration. Power aggregation preserves the contribution of all observed points independently of phase.

Prominent peaks of $P(f)$ provide the candidate observed frequencies, $\hat{f}_1, \ldots, \hat{f}_K$. A frequency peak alone, however, is not sufficient to determine which theoretical beam mode produced it. The excitation may strongly activate some modes and barely activate others, so the first visible spectral peak need not correspond to the first structural eigenmode. We therefore also use the *spatial pattern* associated with each peak.

Observed spatial vibration patterns. A spectral peak tells us that the object is moving strongly at a certain frequency, but it does not tell us *which structural mode* produced that motion. To establish this correspondence, we also need to recover the spatial shape of the vibration at each detected frequency.

For every candidate frequency $\hat{f}_k$, we therefore examine a small frequency neighborhood around the corresponding peak and collect the complex Fourier coefficients of all observed 3D coordinates. Looking at a small neighborhood rather than a single Fourier bin makes the estimate less sensitive to finite sequence length, noise, and spectral leakage.

If one physical vibration mode dominates this neighborhood, the different nearby frequency bins should all describe essentially the same spatial motion, differing mainly in their complex amplitude and phase. We use a local SVD to extract this common dominant pattern. In practice, the leading left singular vector gives a denoised estimate of the observed vibration shape, which we denote by $\hat{\mathbf{y}}_k$.

Intuitively, the spectrum tells us *how fast* the object is oscillating, while $\hat{\mathbf{y}}_k$ tells us *how the different observed points move together* at that frequency. This spatial pattern is what allows us to compare the measured vibration with the structural eigenmodes of the beam model.

Matching observed and structural modes. For the unit-material reference model $E/\rho = 1$, let ϕ_j denote its j-th structural mode and $f_j^{(0)}$ its corresponding reference frequency. Proposition 2 establishes that changing E/ρ rescales the frequencies but does not change the spatial mode shapes. We may therefore compare the observed vibration patterns directly with the precomputed reference modes even before the material scale is known.

Since $\phi_j \in \mathbb{R}^{6N}$ describes the complete structural state, including rotational degrees of freedom that are not directly observed, we first project it into the same space as the measured trajectories:

$$\mathbf{b}_j = H\phi_j.$$

The vector $\mathbf{b}_j$ is therefore the 3D motion that would be visible at the tracked locations if structural mode j were excited.

We compare $\hat{\mathbf{y}}_k$ and $\mathbf{b}_j$ using the Modal Assurance Criterion (MAC),

$$\mathrm{MAC}(k,j) = \frac{\left|\hat{\mathbf{y}}_k^H \mathbf{b}_j\right|^2}{\left(\hat{\mathbf{y}}_k^H \hat{\mathbf{y}}_k\right)\left(\mathbf{b}_j^H \mathbf{b}_j\right)}.$$

MAC measures the alignment between two spatial vibration patterns independently of their overall amplitude. This is important because the magnitude of the observed Fourier coefficient depends on the unknown excitation strength and therefore should not affect mode identity. A MAC value close to 1 indicates that the two patterns agree up to a global complex scale, while a low value indicates different spatial motion.

We use the resulting scores to associate observed peaks with reference modes,

$$\hat{f}_k \leftrightarrow f_{j(k)}^{(0)}.$$

Matches are accepted only when their spatial agreement is sufficiently strong and distinguishable from competing candidates. We additionally enforce an injective assignment so that two unrelated observed peaks are not independently assigned to the same structural mode.

Spatial matching can nevertheless become ambiguous. A mode may be weakly visible at the tracked points, two modes may have very similar projected shapes, or nearly degenerate modes may exchange their numerical ordering. In these cases we use an order-preserving frequency fallback. Since all modal frequencies are multiplied by the same positive material scale, their ordering is preserved as E/ρ changes. We therefore search for an increasing subsequence of the theoretical frequencies that is consistent with the observed peaks, while allowing theoretical modes that were not excited strongly enough to be observed to be skipped.

Recovering the material scale. Once the correspondences are known, the remaining unknown is only the common frequency scale. From Proposition 2,

$$f_j = \sqrt{\frac{E}{\rho}}\, f_j^{(0)}.$$

Defining

$$s = \sqrt{\frac{E}{\rho}},$$

all reliable correspondences should therefore satisfy

$$\hat{f}_k \simeq s f_{j(k)}^{(0)}.$$

Instead of estimating E/ρ independently from every spectral peak, we use all accepted modes to estimate one common scale,

$$s^\star = \arg\min_s \sum_k w_k \left(\hat{f}_k - s f_{j(k)}^{(0)}\right)^2.$$

The weight w_k reflects the reliability of each correspondence, combining its spectral strength and spatial matching confidence, so that weak or uncertain peaks have less influence on the recovered scale.

Finally,

$$\boxed{\widehat{E/\rho} = (s^\star)^2.}$$

The important point is that the observed modes do not provide independent material parameters. They jointly constrain the same scalar E/ρ. Multiple correctly matched modes therefore make the estimate more robust while still respecting the physical scaling shared by the complete structure.

D.2 Final Modal Model and Coordinate Estimation

Recovering E/ρ determines the natural frequencies of the beam system, but it does not yet tell us how strongly each mode was activated at each point in time. We next reconstruct these temporal modal amplitudes, $\mathbf{q}(t)$, which provide the low-dimensional representation of the observed motion.

We first construct the final modal model using the recovered material scale. By Proposition 2, this produces the same spatial mode shapes as the reference system, with their frequencies scaled according to $\widehat{E/\rho}$.

Not every computed structural mode should be retained. We discard modes whose recovered frequency lies above the Nyquist limit of the observed sequence, since such modes cannot be reliably reconstructed from the available temporal sampling. We also discard modes whose motion is effectively invisible in the observation space. For mode ϕ_i, its visible magnitude is measured through

$$\|H\phi_i\|_2.$$

A mode with negligible value may represent a legitimate internal deformation of the beam graph, but if it produces almost no translation at the observed points, its temporal amplitude cannot be estimated reliably from the measurements.

Collecting the remaining modes gives the active basis

$$\Phi = [\phi_1, \ldots, \phi_r].$$

We then define

$$B = H\Phi.$$

Each column of B is the visible displacement pattern produced by one retained structural mode. Thus, B describes how the internal modal coordinates are mapped to the available observations.

Why inspect the SVD of B? Even when two modes are individually visible, they may produce nearly identical displacement patterns at the tracked points. In that situation, the measurements may reveal their combined motion while providing little information about how that motion should be divided between the two modal coordinates.

We therefore compute an SVD of B and inspect its singular values. Large singular values correspond to combinations of modes that are well represented in the observations, whereas very small singular values indicate directions in modal space that produce little distinguishable change in the measured trajectories. The SVD is therefore used as a *diagnostic of observability and conditioning*; it does not itself estimate the temporal modal coordinates.

At every frame, the observed displacement satisfies approximately

$$\mathbf{y}(t) \approx H\Phi\mathbf{q}(t).$$

We estimate the modal coordinates by solving

$$\hat{\mathbf{q}}(t) = \arg\min_{\mathbf{q}} \|H\Phi\mathbf{q} - \mathbf{y}(t)\|_2^2 + \lambda\|\mathbf{q}\|_2^2.$$

The first term asks the reconstructed modal motion to agree with the measured 3D trajectories. The second is a ridge regularizer. Its role is different from the SVD: the SVD tells us whether

the observation problem is poorly conditioned, while the ridge term stabilizes the actual inversion by preventing small observational errors from producing arbitrarily large modal coefficients along poorly constrained directions.

If some points are missing or occluded at a particular frame, the corresponding rows of H and $\mathbf{y}(t)$ can simply be omitted for that frame. The same formulation can therefore operate with a temporally varying set of valid observations.

The result is a sequence

$$\hat{\mathbf{q}}(t) = [\hat{q}_1(t), \ldots, \hat{q}_r(t)]^T$$

describing how strongly each retained physical vibration mode contributes to the observed motion over time.

D.3 Damping Recovery

The natural frequencies determine how quickly the structure oscillates, but they do not determine how quickly those oscillations decay. This loss of mechanical energy is represented by the modal damping ratio ζ. We use the shared-damping model introduced in Appendix C.1,

$$\ddot{q}_i(t) + 2\zeta\omega_i\dot{q}_i(t) + \omega_i^2 q_i(t) = \widetilde{F}_{\mathrm{modal},i}(t).$$

Recovering damping while an unknown external force is still acting is ambiguous: a decreasing oscillation could be caused by physical dissipation, by a changing input force, or by both. We therefore estimate ζ from a free-decay interval whenever one is available. In this interval the external excitation has ended, so the measured reduction in amplitude is attributed to the intrinsic damped dynamics of the structure.

For an underdamped mode, the free response is

$$q_i(t) \approx c_i + e^{-\zeta\omega_i t}\left[a_i \cos(\omega_{d,i}t) + b_i \sin(\omega_{d,i}t)\right],$$

where

$$\omega_{d,i} = \omega_i\sqrt{1-\zeta^2}.$$

The coefficients a_i and b_i determine the amplitude and phase of mode i at the beginning of the fitted interval, while c_i accounts for a small residual offset around the equilibrium configuration.

Rather than estimating an unrelated damping value for every mode, we use a single shared parameter ζ for the retained modal system. The mode-specific quantities a_i, b_i, and c_i are free to vary, while the same ζ must explain the rate of decay across all reliable modes. Combining several modes makes the estimate less sensitive to noise in any individual trajectory.

The free-decay interval is selected after the detected end of the applied excitation, or directly from the known force timing when that timing is available.

As an additional consistency check, damping can also be estimated from the decay envelope of successive modal peaks. If A_{ik} is the magnitude of the k-th peak of mode i, then approximately

$$\log A_{ik} = \beta_i - \zeta\omega_i t_{ik}.$$

Thus, after taking the logarithm, exponential amplitude decay becomes approximately linear in time. We use a robust fit to this relation to reduce sensitivity to noisy peak detections.

Modes with insufficient oscillations or an unstable decay estimate are excluded from the damping fit. The final $\hat{\zeta}$ is then fixed during the subsequent force-recovery stage.

D.4 Force Recovery

Once the modal trajectories, natural frequencies, and damping ratio have been recovered, the modal equation can be evaluated in the opposite direction: rather than predicting motion from a known force, we use the observed motion to infer the force that must have produced it.

For each retained mode,

$$\widetilde{F}_{\mathrm{modal},i}(t) = \ddot{q}_i(t) + 2\hat{\zeta}\hat{\omega}_i\dot{q}_i(t) + \hat{\omega}_i^2 q_i(t).$$

Equivalently,

$$\widetilde{\mathbf{F}}_{\text{modal}}(t) = \ddot{\mathbf{q}}(t) + 2\hat{\zeta}\Omega\dot{\mathbf{q}}(t) + \Omega^2\mathbf{q}(t),$$

where

$$\Omega = \text{diag}(\hat{\omega}_1, \ldots, \hat{\omega}_r).$$

Because this expression contains first and second temporal derivatives, directly differentiating noisy modal trajectories would strongly amplify measurement errors. We therefore smooth the recovered $\hat{\mathbf{q}}(t)$ before computing $\dot{\mathbf{q}}(t)$ and $\ddot{\mathbf{q}}(t)$.

By construction,

$$\widetilde{\mathbf{F}}_{\text{modal}}(t) = \Phi^T\widetilde{\mathbf{F}}(t), \qquad \widetilde{\mathbf{F}}(t) = \frac{\mathbf{F}(t)}{\rho}.$$

The quantity recovered directly from the modal dynamics is therefore the projection of the density-normalized physical force onto the retained modal subspace.

This distinction is important. The structural system has $6N$ force degrees of freedom, whereas only $r \ll 6N$ modal coordinates are retained. Consequently, the mapping

$$\widetilde{\mathbf{F}}(t) \longrightarrow \Phi^T\widetilde{\mathbf{F}}(t)$$

is many-to-one: many different physical force fields can excite exactly the same retained modal response. Therefore, an arbitrary full spatial force cannot be uniquely reconstructed from the observed motion alone.

If only reproduction of the measured motion is required, the recovered

$$\widetilde{\mathbf{F}}_{\text{modal}}(t)$$

is already sufficient. Recovering an interpretable physical excitation requires additional prior knowledge about the type of force that could have acted on the object.

Restricted physical force models. We describe the admissible force using a low-dimensional set of physically meaningful spatial patterns,

$$V = [\tilde{\mathbf{F}}_1, \ldots, \tilde{\mathbf{F}}_K] \in \mathbb{R}^{6N\times K}. \tag{40}$$

For example, a global directional load uses $K = 3$ patterns corresponding to uniform pushes along the global X, Y, and Z translational degrees of freedom, with the rotational degrees of freedom set to zero, while a point-force family restricts the support of the patterns to a selected node or region.

We model the physical force as a time-varying combination of these patterns,

$$\tilde{\mathbf{F}}(t) = V\mathbf{a}(t), \tag{41}$$

where $\mathbf{a}(t) \in \mathbb{R}^K$ contains the strength of each pattern at time t. Substituting this model into the projection identity $\tilde{\mathbf{F}}_{\text{modal}}(t) = \Phi^\top\tilde{\mathbf{F}}(t)$ gives

$$\Phi^\top V\mathbf{a}(t) \approx \tilde{\mathbf{F}}_{\text{modal}}(t). \tag{42}$$

The matrix $\Phi^\top V \in \mathbb{R}^{r\times K}$ describes how strongly each force pattern excites each retained mode. Since the number of patterns is small, $K \ll r$, this system is overdetermined and we solve it independently at each frame by linear least squares,

$$\hat{\mathbf{a}}(t) = \arg\min_{\mathbf{a}} \left\|\Phi^\top V\mathbf{a} - \tilde{\mathbf{F}}_{\text{modal}}(t)\right\|_2^2. \tag{43}$$

The problem is well posed as long as $\Phi^\top V$ has full column rank, i.e., each force pattern excites the retained modes in a distinguishable way. The recovered physical force is then

$$\hat{\tilde{\mathbf{F}}}(t) = V\hat{\mathbf{a}}(t). \tag{44}$$

Because the fit is performed per frame, no temporal profile of the force needs to be assumed; the temporal structure of $\hat{\mathbf{a}}(t)$ is inherited from the smoothed modal trajectories used to compute $\tilde{\mathbf{F}}_{\text{modal}}(t)$.

Finally, because the kinematic observations identify force only relative to density, the recovered quantity remains

$$\hat{\widetilde{\mathbf{F}}}(t) = \frac{\widehat{\mathbf{F}}(t)}{\rho}.$$

An absolute force in Newtons can be obtained only if ρ is known independently.

D.5 Effective Force Calibration

The force recovered above is the effective force required by the reduced beam model to reproduce the observed motion. This distinction becomes relevant when the observations originate from a different physical discretization, such as the volumetric FEM simulator used in our synthetic experiments.

Even when both systems represent the same object and use consistent physical units, their discretizations need not have identical compliance. A given physical force can therefore produce somewhat different displacement magnitudes in the volumetric model and in the reduced beam representation. If this systematic difference is ignored, an otherwise correct recovered force pattern may appear to have the wrong magnitude simply because the two forward models respond differently to the same load.

We account for this mismatch using a single global calibration factor c_F:

$$\widetilde{\mathbf{F}}_{\mathrm{beam}}(t) = c_F \widetilde{\mathbf{F}}_{\mathrm{source}}(t).$$

c_F is obtained as the least-squares scalar that maps the known training force to the force recovered by our method on the training sequence; it is the only use of the training force.

The important restriction is that c_F is estimated only once, using the training excitation. It is then frozen together with the recovered intrinsic parameters. We do not fit a new force scale for each novel interaction.

This makes the novel-force experiment substantially stronger than reconstructing the training sequence alone. If the calibration factor were re-estimated for every novel force, it could absorb part of the model error. Keeping it fixed tests whether the force mapping learned from the original interaction transfers to a genuinely different excitation.

D.6 Forward Re-Simulation and Validation

The final stage verifies that the recovered quantities define a self-consistent dynamical model rather than merely providing a direct decomposition of the observed trajectories.

For the training sequence, the recovered system contains

$$\widehat{E/\rho}, \qquad \hat{\zeta}, \qquad \Phi, \qquad \widehat{\widetilde{\mathbf{F}}}(t),$$

together with the initial modal state when required.

We insert these quantities back into the modal equations,

$$\ddot{q}_{\mathrm{rec},i}(t) + 2\hat{\zeta}\hat{\omega}_i \dot{q}_{\mathrm{rec},i}(t) + \hat{\omega}_i^2 q_{\mathrm{rec},i}(t) = \widehat{\widetilde{F}}_{\mathrm{modal},i}(t),$$

and integrate the system forward in time. For general time-dependent forces we use Newmark–β integration.

For sequences that start from rest, the re-simulation is initialized with $\mathbf{q}(0) = \mathbf{0}$ and $\dot{\mathbf{q}}(0) = \mathbf{0}$; otherwise, the initial modal state is taken from $\hat{\mathbf{q}}(t_0)$.

The reconstructed structural displacement is

$$\mathbf{u}_{\mathrm{rec}}(t) = \Phi \mathbf{q}_{\mathrm{rec}}(t),$$

and the corresponding predicted observations are

$$\mathbf{y}_{\mathrm{rec}}(t) = H \mathbf{u}_{\mathrm{rec}}(t).$$

This step is intentionally different from directly reconstructing $\mathbf{u}(t) = \Phi\hat{\mathbf{q}}(t)$ from the fitted modal coordinates. In the latter case, the measured motion is inserted directly into the reconstruction. In forward re-simulation, only the recovered physical parameters, force, and initial conditions are provided; the subsequent motion must emerge from the dynamical equations themselves. Agreement with the observations therefore tests the complete recovered model.

Novel-force simulation. The main advantage of recovering intrinsic physical parameters is that the model is no longer tied to the particular excitation present in the input sequence. Once

$$\widehat{E/\rho}, \qquad \hat{\zeta}, \qquad \Phi,$$

and the training force-calibration factor have been determined, they remain fixed.

A novel physical excitation $\tilde{\mathbf{F}}_{\text{novel}}(t)$ is projected into the same modal basis. When the novel force is specified in the units of an external physical model, such as the FEM simulator in our synthetic experiments, it is first mapped to the beam model using the fixed calibration factor c_F of Appendix D.5:

$$\tilde{\mathbf{F}}_{\text{novel,modal}}(t) = c_F\, \Phi^\top \tilde{\mathbf{F}}_{\text{novel}}(t). \tag{45}$$

For forces defined directly on the beam model, such as the interactive forces applied to real plants, $c_F = 1$. The same damped modal system is then integrated to obtain $\mathbf{q}_{\text{novel}}(t)$. The resulting structural motion is

$$\mathbf{u}_{\text{novel}}(t) = \Phi \mathbf{q}_{\text{novel}}(t).$$

The novel force may differ from the training interaction in spatial location, direction, magnitude, or temporal profile. No material parameter is re-estimated from the novel sequence. Consequently, successful novel-force prediction tests whether the recovered parameters describe transferable physical properties of the object rather than merely fitting the particular motion used for recovery.

E Observation Model

The beam model describes motion through the generalized displacement $\mathbf{u}(t)$, which contains both translational and rotational degrees of freedom at the skeleton nodes. In contrast, our observations consist only of the 3D motion of tracked points. We therefore require a mapping between the internal structural state and the visible motion.

The trajectories may come from the skeleton joints alone, or be augmented with points sampled on the visual geometry, such as mesh surface points or tracked Gaussians, that are linked to the skeleton through skinning. For skeleton joints, H simply selects the translational degrees of freedom of the corresponding nodes. For points on the visual geometry, H is the linearization of the skinning transformation around the rest configuration.

Each observed point is attached to the structural skeleton in the reference configuration using fixed skinning weights. This binding is computed once and remains unchanged throughout the sequence. The observed displacement can then be approximated as

$$\mathbf{y}(t) \approx H\mathbf{u}(t),$$

where H is the observation operator induced by this fixed geometric binding.

Intuitively, H answers the following question: given a deformation of the beam skeleton, how would that deformation appear at the 3D points that we actually observe? It therefore bridges the structural degrees of freedom of the physical model and the measured trajectories.

Using the modal representation $\mathbf{u}(t) \approx \Phi\mathbf{q}(t)$, the observation model becomes

$$\mathbf{y}(t) \approx H\Phi\mathbf{q}(t).$$

This relation is used throughout the recovery procedure. For a structural mode $\boldsymbol{\phi}_i$, the vector $H\boldsymbol{\phi}_i$ describes how that mode appears at the observed points. This allows us to compare predicted structural modes with the spatial vibration patterns extracted from the measured trajectories. The same mapping is later used to recover the temporal modal coordinates $\mathbf{q}(t)$ from the observed motion.

If some tracked points are missing or unreliable at a particular frame, the corresponding observations are simply omitted from the operator and displacement vector for that frame.

F Dataset Construction and Ground-Truth Generation

Tree Generation and Filtering. To evaluate our method, we constructed a synthetic dataset of trees spanning five botanical categories: aspen, oak, platanus, elm, and bald cypress. For each

category, tree images were generated using the OpenAI API. Each generated image was then independently provided to Claude and Gemini for species identification, and was retained only when the predicted category was consistent with the one requested during generation. We further discarded cases for which a suitable 3D geometry could not be obtained for simulation. The resulting dataset contains **72 trees**.

3D Reconstruction and Geometry. Each accepted image was processed using TRELLIS to obtain a 3D Gaussian Splatting representation (Xiang et al., 2025). We converted the resulting 3DGS into a closed surface mesh and subsequently tetrahedralized it using fTetWild (Hu et al., 2020). For the ground-truth simulations, we used the L48 mesh resolution and quadratic ($P2$) tetrahedral finite elements.

Physical Parameters. Material parameters were assigned according to the botanical category of each tree, using green-wood measurements from the USDA *Wood Handbook* and the moisture–density relations described by Simpson (1993) (Ross & USDA Forest Service., 2010). The corresponding base parameters for each category were

Species	E [GPa]	ρ [kg/m^3]
Aspen	6.80	724.2
Oak	9.08	988.88
Platanus	7.30	1021.2
Elm	7.95	873.98
Bald cypress	8.10	1033.20

To account for natural within-tree variability, E and ρ were modeled as smoothly varying spatial fields with moderate correlated variation. Spatial correlation was defined along the tree structure, such that nearby regions along the same branch tend to have similar material properties, while geometrically close points on different branches are not necessarily correlated. Under our isotropic material approximation, the Poisson ratio was fixed to $\nu = 0.35$ throughout the dataset, using a representative value consistent with values reported for wood in the USDA Wood Handbook (Ross & USDA Forest Service., 2010).

Ground-Truth Simulation. Ground-truth motion was generated using our own implementation of a standard volumetric finite element model with linear isotropic elasticity. The base of each tree was clamped, and the dynamics were represented using its low-frequency natural modes with a uniform damping ratio of $\zeta = 0.05$. To generate the dynamic observation sequence, each tree was driven by a short uniform body-acceleration pulse in the $+x$ direction, equivalent to a distributed body force, followed by free decay. The force magnitude was adapted to the size of each tree in order to maintain a comparable deformation scale across examples. The sampling rate was adapted to the characteristic time scale of each tree's dynamics, while the sequence duration was chosen to capture both the applied force and the subsequent free-decay motion.

Novel-Force Ground Truth. For the generalization experiments, each tree is additionally simulated with the same FEM model under two forces not seen during reconstruction: (i) a distributed force orthogonal ($90°$) to the training force, and (ii) a point force applied at a single skeleton node. For the point force, we first identify the set $\mathcal{G}_{\text{train}}$ of modes that together account for 90% of the training-motion share (Appendix G). We then consider candidate point-force locations whose response excites at least one mode outside this set. In practice, most candidate locations satisfy this condition, and the final node is selected arbitrarily among them. These simulations are used only for evaluation. The generalization experiments use 22 of the 72 trees, selected at random.

Dynamic 3D Observations. The time-varying 3D positions were exported directly from the FEM simulation. In addition, trajectories were extracted for the skeleton joints and points sampled from the 3DGS representation. These trajectories were used directly as input to the reconstruction pipeline, without additional noise injection or temporal filtering.

G Evaluation Metrics

This appendix defines the quantitative metrics used in our experiments. We evaluate three complementary aspects of the recovered model: physical-parameter accuracy, agreement of the reconstructed trajectories with the reference motion, and agreement of their modal content.

G.1 Physical-Parameter Recovery Metrics

Let

$$q = \left(\frac{E}{\rho}\right)_{\mathrm{GT}}, \qquad \hat{q} = \widehat{\left(\frac{E}{\rho}\right)}$$

denote the ground-truth and recovered stiffness-to-density ratios. In our synthetic data, the ground-truth scalar is computed from the FEM material field as

$$q = \frac{\frac{1}{V}\sum_{v=1}^{V} E_v}{\frac{1}{V}\sum_{v=1}^{V} \rho_v},$$

where V is the number of mesh vertices.

Absolute Log Distance Error (ALDE).

$$\mathrm{ALDE} = \left|\log_{10} \frac{\hat{q}}{q}\right|.$$

A value of zero corresponds to exact recovery.

Absolute Log Relative Error (ALRE). We normalize the logarithmic error by the magnitude of the ground-truth value:

$$\mathrm{ALRE} = \frac{\mathrm{ALDE}}{|\log_{10} q|}.$$

All quantities are in SI units, with q measured in $\mathrm{m}^2\mathrm{s}^{-2}$. Lower is better for both metrics.

G.2 Trajectory Metrics

Let U and $\hat{U}$ denote the reference and predicted displacement trajectories, evaluated over the same frames and observed skeleton nodes.

Trajectory Relative L_2 Error.

$$\mathrm{RelL2} = \frac{\|\hat{U} - U\|_F}{\|U\|_F},$$

where the Frobenius norm aggregates over time, skeleton nodes, and the three spatial coordinates. Lower is better.

Cosine Similarity.

$$\mathrm{Cosine} = \frac{\langle \mathrm{vec}(\hat{U}), \mathrm{vec}(U) \rangle}{\|\hat{U}\|_F \, \|U\|_F}.$$

This is an uncentered cosine similarity with range $[-1, 1]$. Higher is better.

For the parameter-recovery experiment, both metrics are evaluated on a fixed 20% held-out set of skeleton joints over all frames, with displacements measured relative to the recovered skeleton rest configuration. For the novel-force experiment, they are evaluated over all skeleton nodes and all frames, with the FEM ground-truth trajectory expressed relative to its first frame.

G.3 MODAL METRICS

To compare the modal content of the predicted and ground-truth motion, we use the first $K = 30$ FEM eigenmodes, interpolated from the volumetric mesh to the non-root skeleton nodes. Let

$$\Psi = [\psi_1, \dots, \psi_K] \in \mathbb{R}^{3(N-1)\times K}$$

denote this skeleton-space modal basis. For a displacement u_t at frame t, its modal coordinates are obtained by least squares,

$$q_t = \arg\min_q \|\Psi q - u_t\|_2^2.$$

Because restricting the FEM modes to the skeleton changes their Euclidean magnitudes, we define the displacement-weighted modal coefficient $c_{j,t} = \|\psi_j\|_2 \, q_{j,t}$. We accumulate the squared contribution over the free-decay portion of the sequence,

$$E_j = \sum_{t \geq t_{\mathrm{rel}}} c_{j,t}^2,$$

where t_{rel} is the first frame after the force pulse has ended, and define the normalized modal share

$$S_j = \frac{E_j}{\sum_{k=1}^K E_k}.$$

We compute these quantities separately for the ground-truth and predicted motion, denoted S_j^{GT} and S_j^{pred}. E_j is a displacement-based quantity used for evaluation and should not be interpreted as mechanical energy.

Target-Mode Energy. This metric measures how much of the predicted motion lies in the modes that characterize the ground-truth response. The target modes are always selected from the ground truth, and the reported value is always computed from the prediction. For the training reconstruction, the target set $\mathcal{G}_{\mathrm{train}}$ is the smallest set of highest-share modes in the FEM training response whose cumulative share $\sum_j S_j^{\mathrm{GT}}$ reaches at least 90%, and we report

$$S_{\mathrm{target}} = \sum_{j \in \mathcal{G}_{\mathrm{train}}} S_j^{\mathrm{pred}}. \tag{46}$$

For a novel-force experiment, we identify the dominant newly excited mode outside the training set from the FEM response to that force,

$$j^\star = \arg\max_{j \notin \mathcal{G}_{\mathrm{train}}} S_j^{\mathrm{GT}}, \tag{47}$$

and report $S_{\mathrm{target}} = S_{j^\star}^{\mathrm{pred}}$. The range is $[0, 1]$, and higher is better.

In Figure 3c, we additionally report, under the novel force, the share of the motion that remains in the training modes, $\sum_{j \in \mathcal{G}_{\mathrm{train}}} S_j$, for both the prediction and the ground truth. Unlike S_{target}, this quantity should match the ground-truth value rather than be maximized.

Modal Distance Error. We compare the full normalized modal distributions of prediction and ground truth using the total-variation distance

$$\mathrm{MDE} = \frac{1}{2} \sum_{j=1}^K \left| S_j^{\mathrm{GT}} - S_j^{\mathrm{pred}} \right|, \tag{48}$$

which is dimensionless, lies in $[0, 1]$, and is zero for identical distributions.

H Implementation Details

This appendix briefly summarizes the main implementation choices and numerical parameters used throughout the reconstruction pipeline. Unless stated otherwise, the same settings are used across all experiments.

The model assumes a fixed Poisson ratio of $\nu = 0.35$. We compute a fixed maximum number of the lowest modes of the beam model, which serves as an upper bound on the modal basis; from these, only the modes below the Nyquist limit and visible at the observed points are retained (Appendix D.2).

During spectral processing, the trajectories undergo linear detrending, RMS normalization, and Hann windowing before the FFT is computed. No zero-padding is used. Only valid and sufficiently reliable observations are included in the reconstruction, and short missing intervals are filled by interpolation.

Observed modes are first matched to the model modes using MAC. When the spatial correspondence is ambiguous, an order-preserving frequency-matching procedure is used. The modal coordinates at each frame are recovered using weighted ridge regression, together with rank and conditioning checks to avoid unstable solutions.

Damping is represented using a single shared damping ratio ζ. Before force recovery, the modal trajectories are temporally smoothed in order to reduce noise amplification during numerical differentiation. Force recovery is solved by per-frame linear least squares, with a rank check on $\Phi^T V$ to ensure each force pattern is distinguishable.

When a global calibration factor is used to relate the force in the beam model to the force in an external physical model, it is estimated once from the training excitation and then kept fixed for all novel-force simulations.

The re-simulations use the recovered physical parameters without re-estimation during novel-force evaluation.

I Skeleton Extraction and Refinement

We extract a compact tree skeleton directly from the Gaussian centers of the reconstructed 3DGS representation. The extraction is topology-aware: it preserves the tree connectivity while avoiding spurious connections between branches that are spatially close but topologically distinct.

We first construct a symmetric k-nearest-neighbor graph over the Gaussian centers and remove edges that are unusually long relative to the median nearest-neighbor distance. This pruning reduces spurious connections between nearby but topologically distinct branches. The graph is then rooted at the trunk base, using all points within a thin base cross-section as source points. We compute the geodesic distance from this base using multi-source Dijkstra. Compared with a single root point, this gives more stable level sets around the hollow surface representation of the trunk.

The Gaussian centers are partitioned into bins according to their geodesic distance from the root. Within each bin, connected components of the pruned graph are identified, and each sufficiently large component is represented by a skeleton node at the mean position of its points. Small disconnected components are absorbed into nearby significant components to suppress spurious nodes caused by surface sampling artifacts. Adjacent level-set components are then connected and converted into a rooted tree.

The resulting dense skeleton is simplified by recursively removing short, insignificant side branches, simplifying degree-two chains while preserving the root, junctions, and branch tips, and merging very short bones near junctions.

Branch radii. The local radius r_e of each beam element is estimated from the mean distance of the level-set component's Gaussian centers to their centroid; for each beam element $e = (i, j)$, $r_e = (r_i + r_j)/2$. Because bending stiffness scales with r_e^4, radius errors directly affect the recovered E/ρ (see below).

Sensitivity to skeleton and radius errors. Our method is particularly sensitive to the accuracy of the extracted skeleton and branch radii, because these quantities directly determine the mass and stiffness matrices of the beam model. For a cylindrical branch, mass scales with the cross-sectional area, $A \propto r^2$, whereas bending stiffness depends on the second moment of area, $I \propto r^4$. Consequently, even a small error in the estimated radius produces a substantially larger error in bending stiffness, and therefore in the predicted natural frequencies and mode shapes. The skeleton geometry is similarly important: branch lengths, orientations, and connectivity determine how the local beam matrices are constructed, rotated, and assembled into the global system (Appendix B), with the bending terms additionally scaling as $1/L$, $1/L^2$, or $1/L^3$. Since our method recovers E/ρ by matching the observed dynamics to this geometry-dependent model, geometric inaccuracies cannot in general be separated from material-property errors. A uniform relative error in the radii mainly rescales the bending frequencies and is largely absorbed into the recovered E/ρ, whereas spatially non-uniform errors also distort the mode shapes and cannot be compensated by a single material scale. Accurate skeleton and radius estimation is therefore critical for reliable physical parameter recovery and subsequent simulation.

Skinning. Each Gaussian is assigned to the nearest bone among those incident to its geodesically assigned skeleton node, and is bound to the bone's two endpoint joints with linear weights $(1-t, t)$ based on its projection along the segment. The same weights define the observation operator H (Appendix E) and are used to render simulated motion.

Manual refinement. Each extracted skeleton is visually inspected overlaid on the reconstructed tree. When the automatic result does not sufficiently follow the geometry, we refine it with a custom interactive skeleton-editing tool, correcting misplaced joints and missing or spurious branches before the skeleton is used for physical modeling and simulation. Manual refinement was required for 19 of the 72 trees. For real videos, the skeleton is instead built from manual first-frame annotations (Appendix J).

J Reconstruction from Real Monocular Videos

To apply our framework to real captures, we construct its static geometry and dynamic observations from a monocular video. The reconstruction front end combines a single canonical surface with temporally persistent motion, producing a reference Gaussian representation, a rooted structural graph, skinning weights, and corresponding 3D trajectories. This stage estimates only geometry and kinematics; material and force identification are performed afterwards by DynaBeam.

Segmentation, Cameras, and Depth. Given an image sequence $\{I_t\}_{t=0}^{T-1}$, we obtain foreground masks using SAM 2 (Ravi et al., 2025), initialized with manually specified foreground prompts in the first frame. We use VGGT-Ω (Wang et al., 2026) to initialize camera calibration and to provide depth estimates with confidence scores. For long sequences, depth inference is performed in overlapping windows that share a reference frame. Each window is scale-aligned to the reference using stationary background regions, so that moving foliage does not affect the alignment. Camera poses are then refined by robust reprojection fitting of stationary-background correspondences to the reference frame, and are held fixed during plant-motion optimization.

Canonical Surface and Gaussian Representation. We generate a textured reference mesh from the masked first frame using Pixal3D (Li et al., 2026a). Its conditioning camera, image crop, and coordinate conventions provide the mapping to the reference image. We transfer the mesh to the reconstruction camera using a projection-preserving mapping and align its depth scale using reliable pixels on the stationary pot. We sample anisotropic Gaussians on the mesh surface, retaining each sample's source triangle and barycentric coordinates. Gaussian colors are initialized from the mesh texture, and their orientations are aligned with the local surface. Colors, opacities, and bounded scale adjustments are first fitted to renders of the textured mesh. These synthetic views supervise the conversion between representations and do not provide additional observations of the real plant. To reduce residual first-frame silhouette errors, we optimize a bounded, smooth image-plane deformation of the canonical surface while preserving its camera-space depth. This correction is applied consistently to the mesh and its Gaussian samples. Visible surface colors are additionally updated

from the reference photograph. The resulting canonical geometry and Gaussian identities remain fixed throughout temporal fitting.

Guide-Assisted Structural Graph. We construct a rooted tree from manually specified first-frame branch paths, junctions, root location, and approximate image-space radii. The paths are resampled into beam segments and lifted to 3D using nearby rendered depths of the canonical mesh, with depth regularization along the graph. For a guide node with depth z_i, annotated pixel radius r_i^{px}, and focal lengths f_x, f_y, its approximate radius in reconstruction coordinates is

$$r_i = \frac{r_i^{\mathrm{px}}\, z_i}{\sqrt{f_x f_y}}. \tag{49}$$

For an edge $e = (i, j)$, we use

$$L_e = \|\mathbf{v}_j^0 - \mathbf{v}_i^0\|_2, \qquad r_e = \frac{r_i + r_j}{2}. \tag{50}$$

The graph is checked to be connected and acyclic. Each Gaussian and mesh vertex is assigned to a nearby beam, with fixed skinning weights on its two endpoints. The pot and structural root are treated as stationary.

Temporally Persistent Motion. We adapt the low-dimensional motion representation of Shape of Motion (Wang et al., 2025). A canonical point $\mathbf{x}_p^0$ follows a shared deformation field

$$\mathbf{x}_p(t) = \mathcal{D}_t(\mathbf{x}_p^0) = R_p(t)\,\mathbf{x}_p^0 + \mathbf{b}_p(t). \tag{51}$$

Point transformations are obtained by blending a compact set of rigid-motion bases using fixed spatial weights initialized from mesh-geodesic neighborhoods. Rotations are blended in a continuous six-dimensional representation and converted back to valid rotation matrices. The temporal rotation-vector and translation parameters of each basis are represented by cubic B-splines. We enforce identity deformation at the reference frame and reserve a stationary basis for the pot and root. AllTracker (Harley et al., 2025) supplies long-range 2D correspondences from multiple query frames. Tracks reliably visible in the reference frame are attached to the canonical mesh through its rasterized triangle identities and barycentric coordinates. Visibility and foreground masks determine which observations contribute to fitting. Tracked pixels are also lifted using the estimated depth and cameras to obtain candidate 3D displacement observations. These are accepted only when supported by depth confidence, distance from depth discontinuities, and spatial and temporal consistency checks; unreliable lifted depth jumps are rejected rather than treated as motion.

Motion Optimization. We optimize the temporal motion parameters using

$$\begin{aligned}\mathcal{L} = {} & \lambda_{\mathrm{rgb}}\mathcal{L}_{\mathrm{rgb}} + \lambda_{\mathrm{mask}}\mathcal{L}_{\mathrm{mask}} + \lambda_{\mathrm{track}}\mathcal{L}_{\mathrm{track}} + \lambda_{\mathrm{flow2D}}\mathcal{L}_{\mathrm{flow2D}} \\ & + \lambda_{\mathrm{flow3D}}\mathcal{L}_{\mathrm{flow3D}} + \lambda_{\mathrm{temp}}\mathcal{L}_{\mathrm{temp}} + \lambda_{\mathrm{struct}}\mathcal{L}_{\mathrm{struct}} + \lambda_{\mathrm{attach}}\mathcal{L}_{\mathrm{attach}}.\end{aligned} \tag{52}$$

The image terms measure RGB and silhouette agreement, including a penalty for Gaussian projections outside the foreground mask. The tracking terms enforce confidence-weighted agreement with both tracked image positions and their inter-frame displacements. The 3D-flow term uses only the accepted lifted observations. Temporal regularization penalizes acceleration and excessive translational or angular motion. Structural regularization encourages local rigidity and preserves graph-edge lengths. The attachment term penalizes disagreement between the dense motion field and the deformation induced by the skeleton through its fixed skinning weights. Additional penalties on the largest residuals discourage localized jumping artifacts that average losses could otherwise hide. Canonical positions, Gaussian scales and opacities, camera parameters, and spatial motion weights are fixed during temporal optimization. A separate, bounded color-only refinement is performed on a short initialization sequence, and the refined appearance is then frozen for the full-length reconstruction.

Skeleton-Driven Dense Replay. We evaluate the fitted motion field at the structural nodes to obtain joint positions $\mathbf{v}_j(t)$ and rest-relative rotations $R_j(t)$. These joint trajectories are the observations passed to DynaBeam. The dense representation is animated using only these joint poses and the fixed skinning weights:

$$\mathbf{x}_g^{\text{skin}}(t) = \sum_{j \in \mathcal{N}(g)} w_{gj} \left[R_j(t)(\mathbf{x}_g^0 - \mathbf{v}_j^0) + \mathbf{v}_j(t) \right], \tag{53}$$

where $\mathcal{N}(g)$ contains the two joints influencing Gaussian g, and $\sum_j w_{gj} = 1$. To rotate each anisotropic covariance, we project the weighted joint-rotation blend onto $\mathrm{SO}(3)$:

$$\overline{R}_g(t) = \mathrm{Proj}_{\mathrm{SO}(3)}\Big(\sum_j w_{gj} R_j(t) \Big), \qquad \Sigma_g(t) = \overline{R}_g(t)\, \Sigma_g^0\, \overline{R}_g(t)^\top. \tag{54}$$

Gaussian colors, opacities, and covariance eigenvalues remain unchanged during replay. Consequently, joint poses simulated by DynaBeam under novel forces can directly replace the reconstructed poses, without rerunning the motion reconstruction.

Implementation and Scope. Our full-length configuration uses 480 consecutive frames, 300,000 Gaussians, and 49 motion bases, including the stationary basis. A 32-frame initialization is followed by full-sequence optimization with 96 temporal spline controls. Gradients are evaluated over 32-frame windows, but all windows update one shared sequence-wide motion model. The exported trajectories preserve point identities across the sequence. However, motion on unobserved surfaces is inferred through the shared deformation field and has no direct observation confidence. Foreground prompts, structural guides, and approximate radii are manual inputs. Hidden geometry and occluded connectivity remain uncertain, and the first frame is a reference configuration rather than a verified unloaded equilibrium. The reconstruction is calibrated to metric units by measuring the physical size of the pot, so the recovered E/ρ is expressed in physical units. Since E/ρ scales with the square of the geometric scale, its accuracy depends on this calibration: a relative scale error of ϵ leads to a relative error of approximately 2ϵ in E/ρ. We therefore use these reconstructions for qualitative real-video applications, while quantitative physical-parameter evaluation uses the synthetic dataset.

Runtime and Supplementary Viewers. Given the reconstructed geometry and trajectories, physical recovery takes 4-5 seconds per plant, and simulating the recovered model runs at 60.3 fps on RTX 3090. This is somewhat longer than the recovery time on the synthetic trees (Table 1), mainly because the real reconstructions contain substantially more tracked Gaussians. The supplementary material contains, for each real plant, a video of the reconstructed and simulated motion, and an interactive HTML viewer. In the viewer, users can apply either point forces by selecting a skeleton node or distributed forces over the entire tree, while choosing the desired force direction and adjusting the force amplitude using a slider.

K Baseline Details

K.1 VoMP

VoMP (Dagli et al., 2025) predicts spatially varying fields $E(\mathbf{x})$, $\rho(\mathbf{x})$, and $\nu(\mathbf{x})$ from a static 3DGS representation. To reduce sensitivity to a single input, we run it on the 3DGS geometry at a set of sampled frames $\mathcal{T}$ and average the resulting estimates. VoMP uses only the geometry and appearance of each frame, not the temporal motion. For the parameter-recovery metrics, we compute at each frame the ratio of the spatial means of the predicted E and ρ and average over frames:

$$\widehat{E/\rho} = \mathrm{mean}_{t \in \mathcal{T}} \left(\frac{\overline{E}_t}{\overline{\rho}_t} \right). \tag{55}$$

VoMP does not estimate damping, forces, or trajectories, and it has no forward simulator of its own. To evaluate its reconstruction, we transfer the fields predicted at each sampled frame to the volumetric FEM mesh used for ground-truth generation, average them per mesh vertex over $\mathcal{T}$, and simulate the training force with this averaged field, supplying damping and force from the ground truth. We

do not use VoMP's original simulator. In a separate validation against an independent MPM reference, its response was effectively stiffer than the predicted parameters imply and did not reproduce the expected natural frequencies. Using the ground-truth FEM simulator for all material predictions isolates differences in the predicted materials from differences between simulation engines.

K.2 PHYSFLOW

The original gradient-based PhysFlow optimization (Liu et al., 2025) did not reliably recover material parameters from an arbitrary initialization in our setting. We keep its fixed-corotated MPM formulation and optical-flow objective; obtaining physically meaningful dynamics additionally required volumetric particle sampling from the tetrahedral mesh, consistent particle volumes, and double-precision simulation. In both variants, ν and the geometry are taken from the ground truth.

Refinement variant. Here PhysFlow is used as a local refinement procedure. We fix $\rho = 1$, so the optimized Young's modulus numerically corresponds to the target ratio E/ρ. We initialize it with the VoMP estimate and perform a short gradient-based refinement around this value. The ground-truth force and damping are not provided. This is closer to the original usage of the method, but introduces a mismatch with the dynamics used to generate our observations. Recovery takes about 30 minutes per tree; due to this runtime, this variant is evaluated on 45 trees.

Oracle variant. This variant evaluates PhysFlow under substantially more favorable conditions. We provide the ground-truth density, external force, and damping used to generate the observed motion, and report $\widehat{E/\rho} = \hat{E}/\rho_{\mathrm{GT}}$. To reduce sensitivity to the unreliable gradients, we first sweep a range of candidate values of E, select the best-performing region according to the PhysFlow objective, and only then apply a short local refinement. Because every candidate requires a full MPM simulation, recovery takes approximately 1.5–5 hours per tree; due to this runtime, this variant is evaluated on 45 of the 72 trees.

The refinement variant thus asks whether PhysFlow can improve a reasonable initial estimate under incomplete knowledge of the dynamics, whereas the oracle variant tests whether it can recover E/ρ when the density, force, damping, and approximate search region are effectively known. Failures in the oracle setting are therefore more diagnostic: they cannot be attributed to unknown excitation, unknown damping, or poor initialization.

K.3 MODALNERF

The original ModalNeRF implementation (Petitjean et al., 2023) is tied to its own scene representation. We implement the same modal principle on our dynamic mesh representation: the observed motion is decomposed into the vibration modes present in the input sequence, and novel responses are simulated within this empirical modal space. This isolates the difference between a basis extracted from observed motion and one derived from a mechanical model. To simulate a novel force, we project the applied force onto ModalNeRF's learned modal basis using the ground-truth FEM mass matrix,

$$\ddot{\mathbf{q}} = (\Phi_e^\top M \Phi_e)^{-1} \Phi_e^\top \mathbf{F}.$$

Because this projection uses the FEM mass model directly, rather than our beam approximation, we do not introduce an additional force-calibration factor. Despite this, ModalNeRF reconstructs the original training motion well. The same uncalibrated projection is used for all novel-force simulations. We also verified that introducing an amplitude calibration analogous to ours does not resolve the novel-force discrepancy, indicating that the dominant limitation lies in the learned modal basis rather than in force magnitude.

K.4 RECONSTRUCTION PROTOCOL

For the reconstruction metrics in Table 1, each method's recovered model is forward-simulated under the training force. DynaBeam uses only its recovered E/ρ, ζ, and F/ρ. VoMP uses its predicted material field, with damping and force supplied from the ground truth. Both PhysFlow variants are re-simulated with the ground-truth force: the refinement variant with its refined E/ρ, and the oracle variant with its recovered E and the ground-truth density, Poisson's ratio, damping, and geometry.