

Text-to-3D Gaussian Splatting with Physics-Grounded Motion Generation

Wenqing Wang, Yun Fu
Northeastern University, USA

360 Huntington Ave, Boston, MA 02115

wang.wenqin@northeastern.edu, yunfu@ece.northeastern.edu

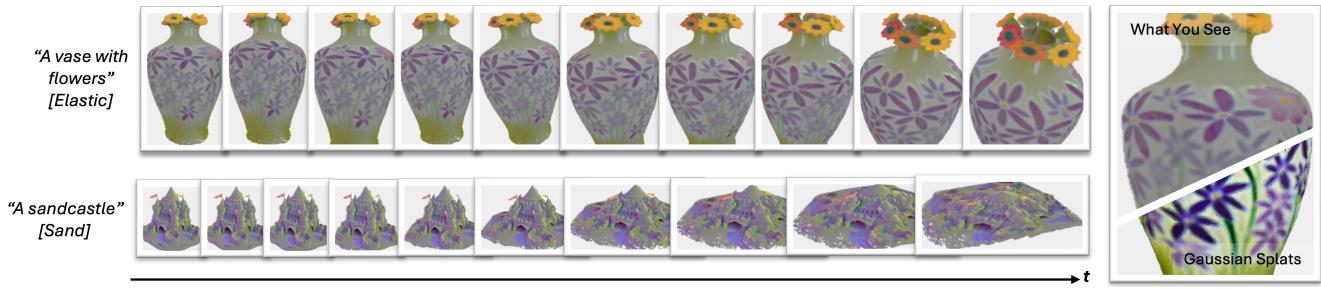


Figure 1. Our framework is a text-to-3D physics-grounded motion-rendering pipeline with high-quality visual appearances and realistic motion.

Abstract

Text-to-3D generation is a valuable technology in virtual reality and digital content creation. While recent works have pushed the boundaries of text-to-3D generation, producing high-fidelity 3D objects with inefficient prompts and simulating their physics-grounded motion accurately still remain unsolved challenges. To address these challenges, we present an innovative framework that utilizes the Large Language Model (LLM)-refined prompts and diffusion priors-guided Gaussian Splatting (GS) for generating 3D models with accurate appearances and geometric structures. We also incorporate a continuum mechanics-based deformation map and color regularization to synthesize vivid physics-grounded motion for the generated 3D Gaussians, adhering to the conservation of mass and momentum. By integrating text-to-3D generation with physics-grounded motion synthesis, our framework renders photo-realistic 3D objects that exhibit physics-aware motion, accurately reflecting the behaviors of the objects under various forces and constraints across different materials. Extensive experiments demonstrate that our approach achieves high-quality 3D generations with realistic physics-grounded motion.

1. Introduction

Text-to-3D modeling has demonstrated remarkable achievements in creating highly realistic 3D representations of ob-

jects. Recently, several works have made great progress in generating delicate 3D objects using text-to-image priors [8, 14, 51, 55]. Additionally, other works have strides in producing the motion of the given 3D objects [9, 20, 27, 33, 57]. Despite these advancements, current methods face challenges in synthesizing realistic 3D objects from inefficient text prompts and accurately simulating their physics-grounded motion.

3D Gaussian Splatting [21] has become a prominent technique in the domain of neural rendering, due to its remarkable ability to render delicate details, point-based representation, and rapid rendering speed. Several works have leveraged 3D GS to generate photo-realistic 3D models from the text prompts [5, 21, 28, 30, 56, 60]. A notable work GSGEN [6] integrates 3D GS with diffusion priors to produce 3D objects with highly realistic structures and visual fidelity. Other works adopt 3D Gaussian representations to model dynamic motion [10, 17, 23, 62]. Xie *et al.* introduces a remarkable framework PhysGaussian [52] that utilizes the physics models that describe the materials’ behaviors to guide the 3D GS to simulate the object motion. These works have laid a robust foundation for the integration of text-to-3D generation and 3D-to-motion simulations.

However, current works have not fully explored techniques for producing high-quality 3D models with realistic, physics-grounded motion from text prompts. In addition, existing text-to-3D frameworks are often guided by text-to-2D image generation models, which have limited text-

understanding ability. This limitation can lead to unsatisfied 3D generations when given poorly written text prompts. To overcome these challenges, we introduce a new framework that enables text-to-3D generation of physics-grounded motion with the aid of LLM-based prompt refinement. To achieve this, we utilize an LLM to refine the input text prompts. Then, we adopt 3D Gaussians as our 3D object representations and use the 3D (shape) diffusion prior and 2D (image) diffusion prior to guide the 3D GS to create photorealistic 3D models with reasonable geometric shapes and realistic appearances. Furthermore, we simulate physics-grounded motion on the generated 3D Gaussians by using a continuum mechanics-based deformation map to deform the Gaussian kernels. Additionally, we introduce a color regularization technique to ensure that the rendered objects maintain accurate and consistent colors. As a result, our framework generates high-quality 3D objects that exhibit physics-grounded motion. In conclusion, our main contributions include:

- We present an innovative framework for synthesizing high-quality 3D objects with realistic, physics-based motion derived from text prompts.
- We leverage an LLM to refine text prompts and diffusion priors to guide the generation of geometrically accurate and visually appealing 3D models.
- We utilize a continuum mechanics-based deformation map combined with a color regularization technique to produce realistic 3D object motion with accurate colors.

2. Related Work

2.1. Neural Rendering

Recent breakthroughs in neural rendering have significantly impacted novel view synthesis. Rendering with radiance fields gained considerable interest due to their remarkable ability to synthesize novel views and their significant promise for advancing 3D generative tasks. Building on this foundation, Neural Radiance Fields (NeRF) [32] revolutionizes volumetric rendering by leveraging neural networks to encode 3D scenes, achieving impressive rendering results. Subsequent works have emerged to improve NeRF in tasks such as 3D scene reconstruction [3, 41, 53, 61], in-the-wild scene handling [4, 31, 45], training speed optimization [7, 25, 49], and rendering quality improvement [11, 16, 48]. However, NeRF poses a computational challenge because of the extensive sampling required along each ray, leading to slow rendering speed and high memory consumption. To overcome these challenges, a point-based rendering method 3D Gaussian Splatting [21] is introduced to represent scenes with 3D Gaussians and render with a fast rasterization method. This enables it to achieve both rapid rendering and high-quality generation. In our proposed framework, we leverage 3D Gaussians to represent

the 3D objects and utilize its point-based nature to generate consistent shapes and realistic motion.

2.2. Text-to-3D Generation

As a groundbreaking approach in generative AI, text-to-3D generation enables synthesizing 3D models directly from the input text prompts. With the recent advancement in diffusion models, a wave of advancements in text-to-3D generation utilize diffusion priors to guide the 3D generation to be aligned with the text prompt description [6, 8, 14, 26, 40, 46, 54]. DreamBooth3D [40] proposes an efficient optimization strategy that leverages the 3D consistency of NeRF with the 2D diffusion prior. DreamFusion [37] utilizes a score distillation sampling loss to align 2D diffusion prior with the generated images during optimization. Magic3D [29] employs a coarse-to-fine optimization process, incorporating diffusion priors to accelerate NeRF’s optimization and improve the quality of generated results. Building on these foundational works, our approach also utilizes diffusion priors to guide 3D generation, focusing on producing high-quality 3D objects with realistic appearances and well-defined shapes.

2.3. The Material Point Method

The Material Point Method (MPM) is a computational framework designed to simulate material behaviors by integrating both particle and grid-based approaches [44]. The MPM represents materials as a set of particles that are mapped onto a grid to compute and simulate the material deformations, which enables the simulation of diverse material properties [12, 24, 58, 59]. Owing to these advantages, we utilize the MPM to generate the physics-grounded motion for our 3D objects.

3. Method

We introduce a framework for synthesizing 3D models with physics-grounded motion using LLM-refined prompts, diffusion priors, and a deformation map (Figure 2). We initially employ an LLM to refine the prompt into a more explicit, detailed, and logically coherent form. For efficient and high-quality 3D model generation, we utilize 3D Gaussian splatting as our object representation. To address challenges such as the Janus problem and to improve the accuracy of the generated 3D object’s shape and appearance, we incorporate guidance from both a 3D shape diffusion prior and a 2D image diffusion prior. Subsequently, a deformation map grounded in continuum mechanics is applied to the 3D Gaussian kernels, enabling realistic motion rendering that adheres to the principles of mass and momentum conservation. This section presents an in-depth explanation of the proposed framework.

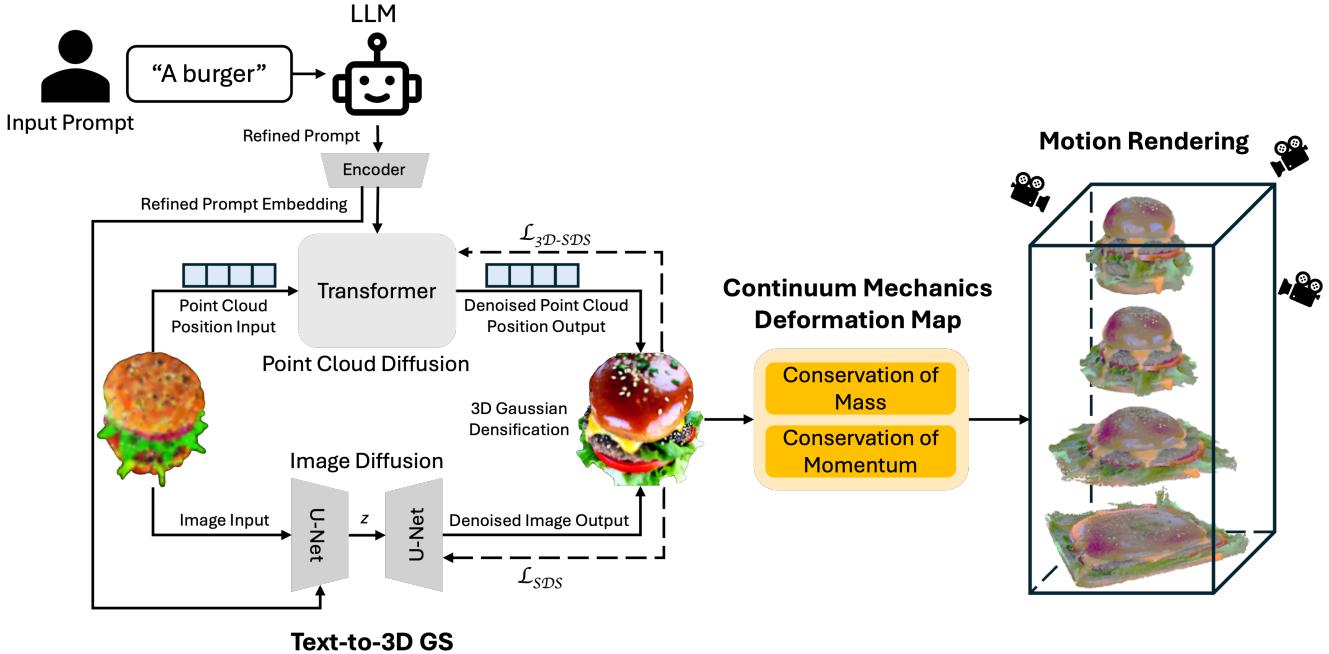


Figure 2. **Pipeline overview.** Our framework first leverages an LLM to refine the text prompt. Next, it employs a 3D geometry diffusion prior and a 2D image diffusion prior for guiding the 3D GS process, producing the high-quality 3D object. Finally, a deformation map based on continuum mechanics is applied to synthesize the physics-grounded motion of the 3D object.

3.1. 3D Gaussian Splatting

3D Gaussian Splatting achieves high-quality scene reconstruction with fast training and rendering speeds [21]. As a point-based rendering approach, it represents a scene using 3D Gaussians, defined by their position (x_i), covariance matrix σ_i , opacity α_i , and spherical harmonic coefficients c_i as $G(x) = e^{-\frac{1}{2}(x)^T \Sigma^{-1}(x)}$. To render a scene, GS first projects the 3D Gaussians into 2D space. To achieve fast rendering, GS employs a tile-based rasterization strategy, which sorts the projected 2D Gaussians based on their depth in view space. Each screen tile is processed by a thread block that loads the Gaussians into shared memory and computes the final pixel colors via alpha-blending:

$$C = \sum_{i \in N} c_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j), \quad (1)$$

where α indicates the opacity, c_i represents the color of each point i , and N denotes the total number of tile Gaussians. To produce high-quality radiance field representation, GS conducts optimization using L_1 and D -SSIM loss: $L_{GS} = (1 - \lambda)L_1 + \lambda L_{D-SSIM}$, and it adaptively controls the density of the 3D Gaussians through pruning and densifying processes [21]. To leverage its fast rendering speed and high-quality rendering ability, we integrate 3D Gaussian Splatting into our framework to generate 3D Gaussians as our object representation. We further extend the GS kernel to incorporate time-dependency in x_i and σ_i , enabling

physics-grounded motion and demonstrating the potential of 3D GS for generative tasks.

3.2. LLM-Prompt Refinement

Text-to-3D generation often produces suboptimal results when the input prompt is vague, overly complex, or involves intricate logical relationships. This limitation arises primarily from the constrained text comprehension capabilities of the guidance models used in the process. Typically, 3D generation models rely on 2D content generation frameworks, including methods like diffusion models [35, 42]. These 2D generation models in turn depend on classifier guidance models like CLIP’s text encoder [39]. These classifier guidance models lack advanced natural language understanding capabilities and are trained on datasets with simple textual descriptions that do not contain complex logic or detailed relational information. Hence, the visual concepts they encode are limited, restricting text-to-3D models to perform effectively only with simple prompts. Furthermore, when the prompts are too vague or brief, the 2D generation models may not have enough context to provide accurate or detailed guidance images for the 3D generation models.

However, we notice that Large Language Models have showcased exceptional abilities in text comprehension, processing, and refinement, owing to their Transformer-based architecture and mechanisms like self-attention and contextual embeddings [2, 34, 47]. Therefore, we leverage an LLM, ChatGPT-4, to refine text prompts for improved text-

to-3D generation. Following the LLM prompt engineering practices in the community [50], the revision instruction prompts that we give to the LLM are composed of the **context** and **task** components.

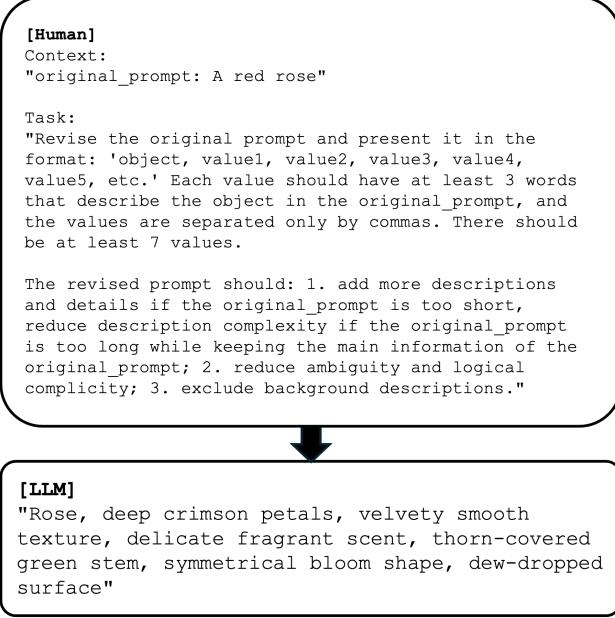


Figure 3. LLM-prompt refinement of a vague text prompt.

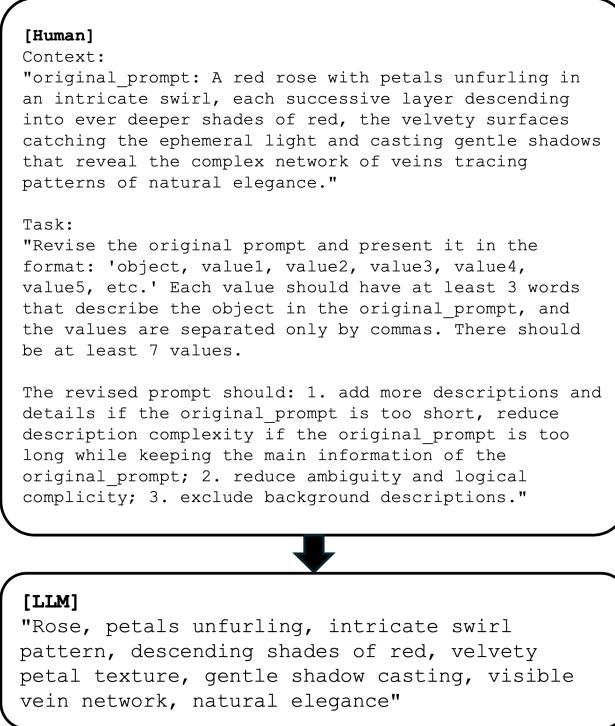


Figure 4. LLM-prompt refinement of a complex text prompt.

Context component. This component contains the origi-

nal text prompt that needs refinement. As mentioned above, some of the issues regarding the original prompts include vagueness, description complexity, and logical complicity. For example, a prompt might simply describe an object with minimal details, such as “a red rose” (Figure 3). However, this prompt lacks sufficient context and fails to specify details about the rose beyond its color, which can result in generated outputs that are vague or insufficiently detailed. Furthermore, prompts might also be overly lengthy and intricate (Figure 4), making it challenging for models to accurately interpret and generate the intended output.

Task component. The task component provides instructions for refining the original prompt. Specifically, it guides the LLM to revise prompts in a comma-separated format, detailing the characteristics of the target object to improve clarity and interpretability for guidance models. For short and under-described prompts, the revision should include additional details and elaborations to provide richer context. However, if the original prompt is lengthy with complex descriptions, it should simplify the description while preserving the core information. Overall, the revised prompt should address any vagueness and logical inconsistencies. In addition, to emphasize the target object for text-to-3D generation, refined prompts should omit background descriptions.

3.3. Text-to-3D GS

To synthesize shape-accurate and visually appealing 3D objects, we leverage 3D Gaussians as our 3D representation using the Gaussian Splatting method. This approach is driven by its point-based structure, capacity to generate high-quality rendering outcomes, and rapid rendering. Building on the work of [6], we iteratively optimize 3D shapes and visual appearances by integrating guidance from diffusion priors. Since 3D Gaussians are created from the point cloud produced by methods such as Structure from Motion, we utilize the 3D diffusion prior from a text-to-3D-point-cloud diffusion framework, *Point-E* [36]. This guides the generation of 3D Gaussians to produce plausible 3D shapes, mitigating the Janus problem—an issue where the model overfits to specific views, resulting in artifacts such as multiple faces or inaccurate geometry. To achieve this, we employ a 3D Score Distillation Sampling (SDS) loss [1] to guide the shape optimization process:

$$L_{\text{shape}} = \mathbb{E}_{\epsilon_I, t} \left[w_I(t) \|\epsilon_\phi(\hat{I}; y, t) - \epsilon_I\|_2^2 \right] + \mathbb{E}_{\epsilon_X, t} \left[w_X(t) \|\epsilon_\psi(x_t; y, t) - \epsilon_X\|_2^2 \right] \cdot \lambda_{3D} \quad (2)$$

where x_t denotes the noisy Gaussian positions and \hat{I} represents the generated image, w and ϵ are the weighting function and Gaussian noise.

To improve the visual quality of the generated 3D models, we refine and densify the 3D Gaussians using a 2D diffusion prior derived from a pre-trained 2D image diffusion

mode, *Stable-Diffusion* [42]. In this process, the Gaussians gradually improve their visual details and appearances. The loss function of this appearance refinement process is:

$$L_{\text{appearance}} = \mathbb{E}_{\varepsilon, I, t} \left[w_I(t) \|\varepsilon_\phi(\hat{I}_t; y, t) - \varepsilon_I\|_2^2 \right] \cdot \lambda_{\text{SDS}} \quad (3)$$

where \hat{I} is the generated image and λ_{SDS} denotes the SDS loss weight. By iteratively optimizing the 3D Gaussians using both the 3D shape diffusion prior and the 2D image diffusion prior, this method enables the synthesis of 3D models that are consistent in shape and visually compelling.

3.4. Physics-Grounded Deformation Map

To generate physics-grounded motion in 3D Gaussians, we incorporate continuum mechanics into our framework. Inspired by the work of [52], we employ a deformation map $\phi(X, t)$ to describe the motion of a particle's position x_i at the time t . Local transformations, such as rotation and stretch at any position, are defined using the deformation map gradient as $F(X, t) = \nabla_X \phi(X, t)$. This can be decomposed as $F = F^E F^P$, where F^E is the elastic part and F^P is the plastic part. To be grounded in continuum mechanics, the updates of the deformation map ϕ conform to the principles of mass and momentum conservation [44].

Under the mass conservation principle, the material mass should remain constant over time, regardless of how the region deforms:

$$\int_{R_\varepsilon^t} \rho(x, t) \equiv \int_{R_\varepsilon^0} \rho(\phi^{-1}(x, t), 0), \quad (4)$$

where $\rho(x, t)$ denotes the material density field and $R_\varepsilon^t = \phi(R_\varepsilon^0, t)$ is the region within the undeformed material space.

According to the conservation of momentum principle, the momentum of any material region should remain unchanged before and after the deformation:

$$\rho(x, t) \dot{v}(x, t) = \nabla \cdot \tau(x, t) + f^{\text{ext}}, \quad (5)$$

where $\tau = \frac{1}{\det(F)} K(F^E)(F^E)^T$ with the Kirchhoff stress tensor $K = \frac{\partial \Psi}{\partial F}$ with a strain energy density $\Psi(F)$, and K depends on the materials' elasticity models. The term $\nabla \cdot \tau(x, t)$ represents the internal forces within the material, while f^{ext} denotes the external force.

To obtain the deformation map $\phi(X, t)$ while satisfying the mass and momentum conservation, we leverage the MPM [44]. This transforms the continuum into discrete Lagrangian particles carrying quantities such as velocity v_i , deformation gradient F_i , and position x_i . To achieve the two-way transfer of information between these Lagrangian particles and Eulerian grids, we utilize B-spline kernels with C^1 degree of continuity [44]. During the time step t^n to t^{n+1} , the mass of Lagrangian particles remains constant under the

mass conservation. With momentum conservation, the momentum of the particles also remains unchanged:

$$\frac{m_j}{\Delta t} (v_j^{n+1} - v_j^n) = - \sum_i V_i^0 \frac{\partial \Psi}{\partial F} (F_i^{E,n})^T \nabla \beta_{j,i}^n + f_j^{\text{ext}}, \quad (6)$$

where i denotes the Lagrangian particles and j represents the Eulerian grid; $m = \rho V$ is mass; β is the B-spline kernel function; V is volume. To update the Lagrangian particles' positions, the updated Eulerian grid velocity v_j^{n+1} is transferred onto v_i^{n+1} , then the particles' positions are updated as $x_i^{n+1} = x_i^n + \Delta t v_i^{n+1}$. The elastic deformation gradients of the particles are updated as $F_i^{E,n+1} = (I + \Delta t \sum_j v_j^{n+1} \nabla (\beta_{j,i}^n)^T) F_i^{E,n}$. Depending on the material-specific plasticity model, $F_i^{E,n+1}$ is adjusted by a mapping as $M : F_i^{E,n+1} \mapsto F_i^{E,n+1}$. Please refer to the supplementary document for detailed information on the plasticity mapping functions.

To apply the deformation map $\phi(X, t)$ to generate physics-grounded motion, we employ 3D Gaussians to represent the discrete particles. Under the assumption that particles undergo local affine transformations, which ensures the deformed Gaussian kernel remains Gaussian in the world space, the deformation map is approximated with the first-order Taylor expansion as $\tilde{\phi}_i(X, t) = x_i + F_i(X - X_i)$. The deformed Gaussian kernel then becomes:

$$\begin{aligned} G_i(x, t) &= e^{-\frac{1}{2}(\tilde{\phi}^{-1}(x, t) - X_i)^T \Sigma_i^{-1} (\tilde{\phi}^{-1}(x, t) - X_i)} \\ &= e^{-\frac{1}{2}(x - x_i)^T (F_i \Sigma_i F_i^T)^{-1} (x - x_i)} \end{aligned} \quad (7)$$

Given the 3D Gaussians with $\{X_i, \Sigma_i, \alpha_i, c_i\}$, the deformation map $\phi(X, t)$ deforms them to $\{x_i(t), \sigma(t), \alpha_i, c_i\}$, where $x_i(t) = \phi(X_i, t)$ and $\sigma_i(t) = F_i(t) \Sigma_i F_i(t)^T$.

To ensure consistent and accurate RGB color values during rendering, we regularize the RGB values converted from spherical harmonics c_i through a color regularization process that includes the normalization and clamping steps.

Normalization. The normalization step rescales the RGB values to fit within the $[0, 1]$ range. This is achieved by subtracting the minimum value of the original RGB tensor and dividing by the difference between the maximum and minimum values, as follows:

$$s_{\text{norm}} = \frac{s - s_{\min}}{s_{\max} - s_{\min}}, \quad (8)$$

where s represents the original RGB tensor, and s_{\min} , s_{\max} are the minimum and maximum values in s , and s_{norm} is the normalized RGB tensor.

Clamping. While the normalization step adjusts the RGB values to $[0, 1]$, numerical precision issues can sometimes cause minor deviations, leading to values outside this range. For instance, these issues might arise when rounding errors accumulate to allow some values to exceed the $[0, 1]$

range, and when normalization does not precisely yield values within $[0, 1]$. To address this, a clamping step is applied to enforce strict adherence to $[0, 1]$. The clamping operation is defined as:

$$s_{\text{clamp}} = \min(\max(s_{\text{norm}}, 0), 1). \quad (9)$$

With the RGB values regularized, a GS rasterizer is used to render the deformed Gaussian kernels. This process produces high-quality 3D objects with realistic physics-grounded motion.

4. Experiments

In this section, we evaluate the effectiveness of our proposed framework through comprehensive experiments. We provide both qualitative and quantitative evaluations, along with our detailed ablation study results.

4.1. Implementation Details

Setup. We utilize Pytorch to implement the 3D Gaussian Splatting, adhering to the optimization pipeline from [21]. To generate physics-grounded motion, we build upon the MPM [44, 52]. Our experiments are conducted on an Nvidia RTX 3090 GPU.

Metrics. We utilize the LAION aesthetic score [17], which evaluates the aesthetic quality of a video on a scale from 0 to 10. Furthermore, we employ the CLIP score [39] to measure the prompt consistency of a video, which is the average cosine similarity between input prompt and all video frames. In addition, we adopt the Mean Opinion Score (MOS) for the human study evaluations on the generated videos.

Method comparison. As the first approach to utilize text for generating 3D objects with physics-grounded motion, there is no existing method for direct comparison. Due to the generative nature of our framework, the ground truth of the deformed scenes is also unavailable. To evaluate our method, we provide both qualitative and quantitative results of our framework and compare them against the results of the relevant 3D-to-motion methods [15, 38], using the 3D models provided by our approach.

4.2. Qualitative Evaluation

We showcase the qualitative performance of our framework by creating high-quality 3D objects featuring diverse physics-based dynamics. For each material dynamics type, we present an example illustrating the deformation motion of a 3D object, as shown in Figure 5. Please review the supplementary document for the material-specific elasticity and plasticity dynamics models.

Our framework demonstrates the following dynamics: **elastic, fracture, jelly, metal, and sand**. Elastic and jelly dynamics refer to the behavior of objects where the rest

shape remains unchanged despite undergoing deformation. Metal dynamics cause the object to undergo permanent deformation when its stress reaches a certain threshold, governed by von Mises plasticity model [43]. Fracture dynamics simulate particle separation into clusters under significant deformation. Sand dynamics, modeled by Druker-Prager plasticity model [22], captures granular level frictional effects among particles. As illustrated in Figure 5, our results demonstrate the capability of our framework to synthesize 3D objects with high-quality appearances, accurate shapes, and realistic physics-grounded motion for the given text prompts and material types.

Qualitative comparisons. Since our framework is the first to bridge text-to-3D synthesis with physics-grounded motion, there are no existing works available for direct qualitative comparison. Therefore, we evaluate the qualitative performance of our framework in comparison to the 3D-to-motion methods DreamPhysics [15] and Feature-Splatting [38], using our provided 3D models. Their corresponding results are shown in Figure 5 and Figure 6. Compared to the results of our framework, we observe that the results produced by DreamPhysics have under-saturated colors with weaker and less realistic motion. For instance, its fracture motion for the sponge object fails to achieve actual fracturing; instead, the object merely shrinks. Additionally, DreamPhysics’s generated metal motion for the can object lacks a realistic metallic movement when force is applied, particularly at the top of the can. Feature-Splatting, on the other hand, generates over-saturated and often inaccurate colors, with its motions being nearly imperceptible. In addition, it struggles with video segmentation in certain cases, such as the elastic burger example, which prevents successful motion generation. Moreover, it can only generate motion for elastic and sand material types, which has less versatility compared to our framework.

Criteria	Ours	DreamPhysics	Feature-Splatting
Appearance	2.87	2.58	2.36
Shape Accuracy	3.22	3.07	2.82
Motion Quality	3.04	2.67	2.61

Table 1. User study MOS results. Best results are in **bold**.

User study. A user study is conducted to evaluate the human-perceived quality of the synthesized 3D object motion videos. In the study, we recruit 20 participants to evaluate the videos. For each material type—elastic, fracture, jelly, metal, and sand—we generate corresponding 3D object motion videos using our framework and DreamPhysics. In addition, we employ Feature-Splatting to generate its elastic and sand video results. To measure participant evaluations of the generated videos, we adopt the Mean Opinion Score, with ratings ranging from 1 (Bad) to 5 (Excellent). Participants are instructed to rate each video based on 3 cri-

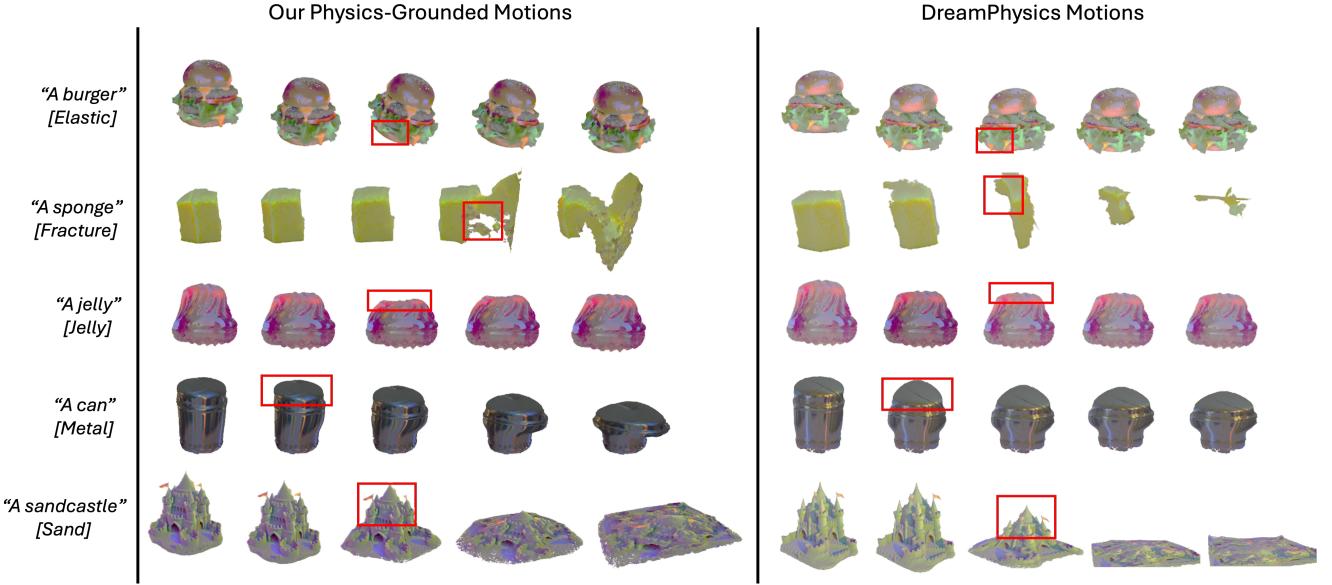


Figure 5. **Our results and DreamPhysics results.** We present our text-to-3D physics-grounded motion results and the results generated by DreamPhysics using the 3D models provided by our framework.

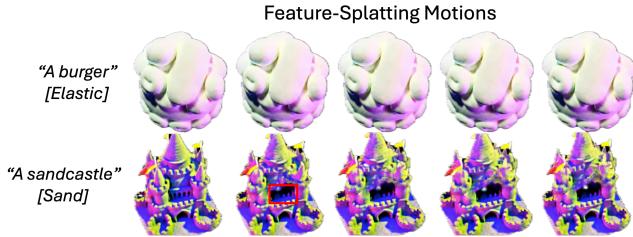


Figure 6. **Feature-Splatting results.** Results generated by Feature-Splatting using the 3D models provided by our framework.

teria: 1) visual appearance; 2) shape accuracy; 3) motion quality.

We observe from the MOS results (Table 1) that: 1) Our framework outperforms the other methods in visual appearance, shape accuracy, and video quality. 2) The MOS results are aligned with the evaluation results presented in the paper.

Method	LAION \uparrow	CLIP Score \uparrow	Resolution \uparrow	Generation Time (min) \downarrow
DreamPhysics [15]	3.77	0.269	800x800	3.58
Feature-Splatting [38]	1.98	0.268	512x512	6.57
Ours	3.80	0.278	1958x1090	1.7

Table 2. Results of quantitative evaluation. Best results are in **bold**.

4.3. Quantitative Evaluation

We conduct quantitative comparison with other 3D-to-motion methods [15, 38] using the 3D models generated by our framework. As presented in Table 2, our method surpasses other frameworks across multiple metrics, including

the mean LAION score, CLIP score, video resolution, and generation time. These results highlight that our framework can achieve higher aesthetic quality, better prompt-video consistency, improved visual quality, and faster generation time. In contrast, despite using the 3D models generated by our framework, both DreamPhysics and Feature-Splatting exhibit worse performance across all metrics. This indicates the superiority and efficacy of our framework in producing high-quality text-to-3D motion videos.

4.4. Ablation Studies

Ablation studies are conducted on our proposed framework to demonstrate the necessity of **1) LLM-prompt refinement**, **2) 3D diffusion prior guidance**, **3) RGB color regularization**. The presented ablation results demonstrate the contribution of each of these components in improving the overall performance of our system. From Table 3, we observe that each proposed component plays an important role in ensuring both high aesthetic quality and strong prompt-video consistency. Overall, it shows that 3D diffusion prior guidance produces a larger impact on the LAION score, which indicates its importance in obtaining a high aesthetic quality. On the other hand, LLM-prompt refinement has a greater effect on the CLIP score, reflecting its ability to achieve semantically aligned generation results.

LLM-prompt refinement. Figure 7 illustrates that without LLM-prompt refinement (LLM-PR), the generated object suffers from inaccuracies in object details. For instance, in the result generated without LLM-PR, the coloration of the salmon is inconsistent with its natural appearance, and the rice grains are inaccurately positioned on top of the salmon,

reducing the overall realism and coherence of the scene. Quantitative ablation results in Table 3 also shows that the absence of LLM-PR leads to reduced LAION score and CLIP score, which reflects the critical role of LLM-PR in achieving a high aesthetic quality and prompt-video consistency.

3D diffusion prior guidance. The results in Figure 8 demonstrate that the incorporation of the 3D diffusion prior as guidance (3D Guidance) enables the Gaussian Splatting to generate the 3D object with a more accurate geometrical shape, compared to the results generated without this guidance. Furthermore, as presented in Table 3, the absence of the 3D diffusion prior guidance results in lower LAION and CLIP scores. This indicates the importance of 3D Guidance in improving both the aesthetic quality and the coherence between the input prompt and the resulting video.

Color regularization. Our results in Figure 9 indicate that the absence of the color regularization (CR) leads to the inaccurately rendered colors of the 3D object, compared to the results generated with the color regularization. This is also reflected in Table 3, because the LAION score and CLIP score in the results generated without CR are noticeably reduced. This shows that CR plays a significant role in obtaining accurate color rendering, which also ensures a high aesthetic quality and prompt consistency of the generated video.

Method	LAION \uparrow	CLIP Score \uparrow
w/o LLM-PR	3.62	0.256
w/o 3D Guidance	3.29	0.263
w/o CR	3.48	0.266
Full	3.80	0.278

Table 3. Ablation study results. Best results are in **bold**.

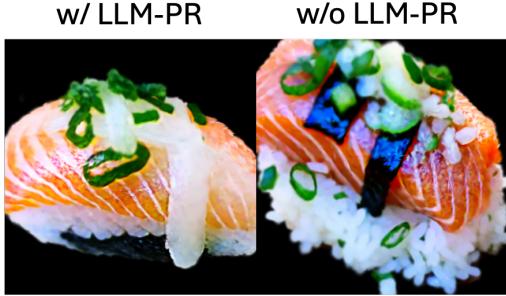


Figure 7. The impact of adopting LLM-prompt refinement. Prompt: *A salmon nigiri*.

5. Discussion

Limitation. Our framework does not currently support rendering the interaction of 3D object surfaces with light, so it

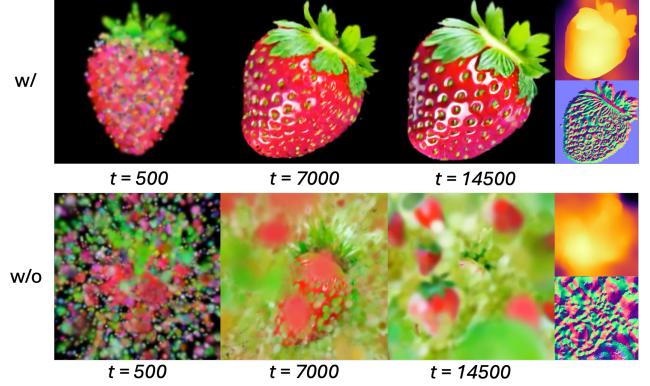


Figure 8. The impact of employing the 3D diffusion prior as GS shape guidance. Prompt: *A strawberry*.

w/ CR w/o CR



Figure 9. The effects of applying the color regularization on the RGB values. Prompt: *A vase with flowers*.

cannot generate the effects such as reflections or shadows. Additionally, our framework only supports the motion simulation of limited material types. Future work could explore integrating advanced relighting techniques and expanding the range of material types to enhance the framework’s versatility and realism.

Conclusion. In this paper, we present an innovative framework for text-to-3D motion generation based on physics, facilitating the creation of high-quality 3D objects with realistic, physics-aware movements, effectively integrating generative modeling with physics-driven motion simulation. Our framework integrates four innovative components: 1) **LLM-prompt refinement** to ensure the accurate 3D generation for the prompt; 2) **diffusion prior guidance** for steering the generative process toward the result with accurate shape and high-quality visual appearance; 3) **continuum mechanics-based deformation mapping** to model realistic physical interactions and deformations of the generated 3D object; 4) **color regularization** for consistent and accurate color rendering. This unified pipeline integrates natural language processing, generative modeling, and physics simulation to redefine the boundaries of 3D content creation, which paves the way for transformative applications across diverse industries such as filmmaking, virtual/augmented reality, gaming, and beyond.

References

- [1] Thiemo Alldieck, Nikos Kolotouros, and Cristian Sminchisescu. Score distillation sampling with learned manifold corrective, 2024. 4
- [2] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. 3
- [3] Hansheng Chen, Jiatao Gu, Anpei Chen, Wei Tian, Zhuowen Tu, Lingjie Liu, and Hao Su. Single-stage diffusion nerf: A unified approach to 3d generation and reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2416–2425, 2023. 2
- [4] Xingyu Chen, Qi Zhang, Xiaoyu Li, Yue Chen, Ying Feng, Xuan Wang, and Jue Wang. Hallucinated neural radiance fields in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12943–12952, 2022. 2
- [5] Zilong Chen, Feng Wang, Yikai Wang, and Huaping Liu. Text-to-3d using gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21401–21412, 2024. 1
- [6] Zilong Chen, Feng Wang, Yikai Wang, and Huaping Liu. Text-to-3d using gaussian splatting, 2024. 1, 2, 4
- [7] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12882–12891, 2022. 2
- [8] Lihe Ding, Shaocong Dong, Zhanpeng Huang, Zibin Wang, Yiyuan Zhang, Kaixiong Gong, Dan Xu, and Tianfan Xue. Text-to-3d generation with bidirectional diffusion using both 2d and 3d priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5115–5124, 2024. 1, 2
- [9] Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. Momask: Generative masked modeling of 3d human motions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1900–1910, 2024. 1
- [10] Zhiyang Guo, Wengang Zhou, Li Li, Min Wang, and Houqiang Li. Motion-aware 3d gaussian splatting for efficient dynamic scene reconstruction, 2024. 1
- [11] Yuqi Han, Tao Yu, Xiaohang Yu, Di Xu, Binge Zheng, Zonghong Dai, Changpeng Yang, Yuwang Wang, and Qionghai Dai. Super-nerf: View-consistent detail generation for nerf super-resolution. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–14, 2024. 2
- [12] Lucy Harris, Dongfang Liang, Songdong Shao, Taotao Zhang, and Grace Roberts. Mpm simulation of solitary wave run-up on permeable boundaries. *Applied Ocean Research*, 111:102602, 2021. 2
- [13] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. 2
- [14] Fangzhou Hong, Jiaxiang Tang, Ziang Cao, Min Shi, Tong Wu, Zhaoxi Chen, Shuai Yang, Tengfei Wang, Liang Pan, Dahua Lin, and Ziwei Liu. 3dtopia: Large text-to-3d generation model with hybrid diffusion priors, 2024. 1, 2
- [15] Tianyu Huang, Haoze Zhang, Yihan Zeng, Zhiliu Zhang, Hui Li, Wangmeng Zuo, and Rynson W. H. Lau. Dreamphysics: Learning physical properties of dynamic 3d gaussians with video diffusion priors, 2024. 6, 7, 2
- [16] Xudong Huang, Wei Li, Jie Hu, Hanting Chen, and Yunhe Wang. Refsr-nerf: Towards high fidelity and super resolution view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8244–8253, 2023. 2
- [17] Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4220–4230, 2024. 1, 6
- [18] Chenfanfu Jiang, Craig Schroeder, Andrew Selle, Joseph Teran, and Alexey Stomakhin. The affine particle-in-cell method. *ACM Trans. Graph.*, 34(4), 2015. 1
- [19] Chenfanfu Jiang, Craig Schroeder, Joseph Teran, Alexey Stomakhin, and Andrew Selle. The material point method for simulating continuum materials. In *ACM SIGGRAPH 2016 Courses*, New York, NY, USA, 2016. Association for Computing Machinery. 1
- [20] Roy Kapon, Guy Tevet, Daniel Cohen-Or, and Amit H. Bermano. Mas: Multi-view ancestral sampling for 3d motion generation using 2d diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1965–1974, 2024. 1
- [21] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering, 2023. 1, 2, 3, 6
- [22] Gergely Klár, Theodore Gast, Andre Pradhana, Chuyuan Fu, Craig Schroeder, Chenfanfu Jiang, and Joseph Teran. Drucker-prager elastoplasticity for sand animation. *ACM Trans. Graph.*, 35(4), 2016. 6, 1
- [23] Junghe Lee, Donghyeong Kim, Dogyoone Lee, Suhwan Cho, and Sangyoun Lee. Crim-gs: Continuous rigid motion-aware gaussian splatting from motion blur images, 2024. 1
- [24] Zhengda Lei, Bisheng Wu, Shengshen Wu, Yuanxun Nie, Shaoyi Cheng, and Chongyuan Zhang. A material point-finite element (mpm-fem) model for simulating three-dimensional soil-structure interactions with the hybrid contact method. *Computers and Geotechnics*, 152:105009, 2022. 2
- [25] Ruilong Li, Hang Gao, Matthew Tancik, and Angjoo Kanazawa. Nerfacc: Efficient sampling accelerates nerfs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 18537–18546, 2023. 2
- [26] Weiyu Li, Rui Chen, Xuelin Chen, and Ping Tan. Sweet-dreamer: Aligning geometric priors in 2d diffusion for consistent text-to-3d, 2023. 2

- [27] Han Liang, Jiacheng Bao, Ruichi Zhang, Sihan Ren, Yuecheng Xu, Sibei Yang, Xin Chen, Jingyi Yu, and Lan Xu. Omg: Towards open-vocabulary motion generation via mixture of controllers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 482–493, 2024. 1
- [28] Yixun Liang, Xin Yang, Jiantao Lin, Haodong Li, Xiaogang Xu, and Yingcong Chen. Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6517–6526, 2024. 1
- [29] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation, 2023. 2
- [30] Xian Liu, Xiaohang Zhan, Jiaxiang Tang, Ying Shan, Gang Zeng, Dahua Lin, Xihui Liu, and Ziwei Liu. Humangaussian: Text-driven 3d human generation with gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6646–6657, 2024. 1
- [31] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7210–7219, 2021. 2
- [32] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis, 2020. 2
- [33] Aymen Mir, Xavier Puig, Angjoo Kanazawa, and Gerard Pons-Moll. Generating continual human motion in diverse 3d scenes. In *2024 International Conference on 3D Vision (3DV)*, pages 903–913, 2024. 1
- [34] Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, Johannes Heidecke, Pranav Shyam, Boris Power, Tyna Eloundou Nekoul, Girish Sastry, Gretchen Krueger, David Schnurr, Felipe Petroski Such, Kenny Hsu, Madeleine Thompson, Tabarak Khan, Toki Sherbakov, Joanne Jang, Peter Welinder, and Lilian Weng. Text and code embeddings by contrastive pre-training, 2022. 3
- [35] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models, 2022. 3
- [36] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts, 2022. 4
- [37] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion, 2022. 2
- [38] Ri-Zhao Qiu, Ge Yang, Weijia Zeng, and Xiaolong Wang. Feature splatting: Language-driven physics-based scene synthesis and editing, 2024. 6, 7, 2
- [39] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 3, 6
- [40] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, Yuanzhen Li, and Varun Jampani. Dreambooth3d: Subject-driven text-to-3d generation, 2023. 2
- [41] Fabio Remondino, Ali Karami, Ziyang Yan, Gabriele Mazzacca, Simone Rigon, and Rongjun Qin. A critical analysis of nerf-based 3d reconstruction. *Remote Sensing*, 15(14), 2023. 2
- [42] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 3, 5
- [43] Arunabha M. Roy and Suman Guha. A data-driven physics-constrained deep learning computational framework for solving von mises plasticity. *Engineering Applications of Artificial Intelligence*, 122:106049, 2023. 6, 2
- [44] D Sulsky. A particle method for history-dependent materials. *Computer Methods in Applied Mechanics and Engineering*, 118(1–2):179–196, 1994. 2, 5, 6, 1
- [45] Jiaming Sun, Xi Chen, Qianqian Wang, Zhengqi Li, Hadar Averbuch-Elor, Xiaowei Zhou, and Noah Snavely. Neural 3d reconstruction in the wild. In *ACM SIGGRAPH 2022 Conference Proceedings*, New York, NY, USA, 2022. Association for Computing Machinery. 2
- [46] Junshu Tang, Tengfei Wang, Bo Zhang, Ting Zhang, Ran Yi, Lizhuang Ma, and Dong Chen. Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22819–22829, 2023. 2
- [47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. 3
- [48] Chen Wang, Xian Wu, Yuan-Chen Guo, Song-Hai Zhang, Yu-Wing Tai, and Shi-Min Hu. Nerf-sr: High quality neural radiance fields using supersampling. In *Proceedings of the 30th ACM International Conference on Multimedia*, page 6445–6454, New York, NY, USA, 2022. Association for Computing Machinery. 2
- [49] Peng Wang, Yuan Liu, Zhaoxi Chen, Lingjie Liu, Ziwei Liu, Taku Komura, Christian Theobalt, and Wenping Wang. F2-nerf: Fast neural radiance field training with free camera trajectories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4150–4159, 2023. 2
- [50] Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C. Schmidt. A prompt pattern catalog to enhance prompt engineering with chatgpt, 2023. 4
- [51] Zike Wu, Pan Zhou, Xuanyu Yi, Xiaoding Yuan, and Hanwang Zhang. Consistent3d: Towards consistent high-fidelity text-to-3d generation with deterministic sampling prior. In *Proceedings of the IEEE/CVF Conference on Computer Vi-*

- sion and Pattern Recognition (CVPR)*, pages 9892–9902, 2024. 1
- [52] Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. Physgaussian: Physics-integrated 3d gaussians for generative dynamics, 2024. 1, 5, 6
- [53] Hongyi Xu, Thiendo Alldieck, and Cristian Sminchisescu. H-nerf: Neural radiance fields for rendering and temporal reconstruction of humans in motion. In *Advances in Neural Information Processing Systems*, pages 14955–14966. Curran Associates, Inc., 2021. 2
- [54] Jiale Xu, Xintao Wang, Weihao Cheng, Yan-Pei Cao, Ying Shan, Xiaohu Qie, and Shenghua Gao. Dream3d: Zero-shot text-to-3d synthesis using 3d shape prior and text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20908–20918, 2023. 2
- [55] Zizheng Yan, Jiapeng Zhou, Fanpeng Meng, Yushuang Wu, Lingteng Qiu, Zisheng Ye, Shuguang Cui, Guanying Chen, and Xiaoguang Han. Dreamdissector: Learning disentangled text-to-3d generation from 2d diffusion priors, 2024. 1
- [56] Taoran Yi, Jiemin Fang, Junjie Wang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussians by bridging 2d and 3d diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6796–6807, 2024. 1
- [57] Wanyue Zhang, Rishabh Dabral, Thomas Leimkühler, Vladislav Golyanik, Marc Habermann, and Christian Theobalt. Roam: Robust and object-aware motion generation using neural pose descriptors. In *2024 International Conference on 3D Vision (3DV)*, pages 1392–1402, 2024. 1
- [58] Xiangcou Zheng, Federico Pisanò, Philip J. Vardon, and Michael A. Hicks. Fully implicit, stabilised mpm simulation of large-deformation problems in two-phase elastoplastic geomaterials. *Computers and Geotechnics*, 147:104771, 2022. 2
- [59] Xiangcou Zheng, Shuying Wang, Feng Yang, and Junsheng Yang. Material point method simulation of hydro-mechanical behaviour in two-phase porous geomaterials: A state-of-the-art review. *Journal of Rock Mechanics and Geotechnical Engineering*, 16(6):2341–2350, 2024. 2
- [60] Xiaoyu Zhou, Xingjian Ran, Yajiao Xiong, Jinlin He, Zhiwei Lin, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Gala3d: Towards text-to-3d complex scene generation via layout-guided generative gaussian splatting, 2024. 1
- [61] Yiming Zhou, Zixuan Zeng, Andi Chen, Xiaofan Zhou, Haowei Ni, Shiyao Zhang, Panfeng Li, Liangxi Liu, Mengyao Zheng, and Xupeng Chen. Evaluating modern approaches in 3d scene reconstruction: Nerf vs gaussian-based methods. In *2024 6th International Conference on Data-driven Optimization of Complex Systems (DOCS)*, pages 926–931, 2024. 2
- [62] Ruijie Zhu, Yanzhe Liang, Hanzhi Chang, Jiacheng Deng, Jiahao Lu, Wenfei Yang, Tianzhu Zhang, and Yongdong Zhang. Motions: Exploring explicit motion guidance for deformable 3d gaussian splatting, 2024. 1
- [63] Zeshun Zong, Xuan Li, Minchen Li, Maurizio M. Chiaramonte, Wojciech Matusik, Eitan Grinspun, Kevin Carlberg, Chenfanfu Jiang, and Peter Yichen Chen. Neural stress fields for reduced-order elastoplasticity and fracture. In *SIGGRAPH Asia 2023 Conference Papers*, New York, NY, USA, 2023. Association for Computing Machinery. 1

Text-to-3D Gaussian Splatting with Physics-Grounded Motion Generation

Supplementary Material

6. Theoretical Details

6.1. The Material Point Method

The MPM is a computational physics method to simulate the material behaviors under different physical forces and deformations [44]. The MPM discretizes a material body into a collection of Lagrangian particles, and each particle has a set of quantities such as position x_i^n , mass m_i , velocity v_i^n , Kirchhoff stress tensor K_i^n , deformation gradient F_i^n , and affine momentum A_i^n on particle i at time t^n . At time t^n , let x_j^n , m_j , and v_j^n represent the position, mass, and velocity on grid node j . These grid nodes facilitate the calculation of the deformations and the applied forces on the material body. Due to the conservation of mass, particle mass is invariant. At each time step, the MPM conducts a two-way transfer: 1) Particle-to-Grid; 2) Grid-to-Particle.

Particle-to-Grid Transfer. In this process, the mass and particle momentum are transferred to the grids [52]. The mass m_j^n at a grid node j is computed as:

$$m_j^n = \sum_i w_{ji}^n m_i, \quad (10)$$

where w_{ji}^n is the interpolation weight obtained from a B-spline kernel. Using the APIC momentum transfer method [18], the momentum at the grid node j is updated as:

$$m_j^n v_j^n = \sum_i w_{ji}^n m_i (v_i^n + A_i^n (x_j - x_i^n)). \quad (11)$$

Based on the particles' internal and external forces, the grid velocity v_j^{n+1} at the next time step is updated as:

$$v_j^{n+1} = v_j^n - \frac{\Delta t}{m_j} \sum_i K_i^n \nabla w_{ji}^n V_i^0 + \Delta t g, \quad (12)$$

where g is the gravity acceleration.

Grid-to-Particle Transfer. In this stage, the grid nodes' updated velocities and momentum are transferred back to the particles [19, 52]. The velocity v_i^{n+1} , position x_i^{n+1} , affine momentum A_i^{n+1} , and deformation gradient F_i^{n+1} of particle i at the new time step are updated as:

$$\begin{aligned} v_i^{n+1} &= \sum_j v_j^{n+1} w_{ji}^n, \\ x_i^{n+1} &= x_i^n + \Delta t v_i^{n+1}, \\ A_i^{n+1} &= \frac{12}{\Delta x^2(b+1)} \sum_j w_{ji}^n v_j^{n+1} (x_j^n - x_i^n)^T, \\ \nabla v_i^{n+1} &= \sum_j v_j^{n+1} (\nabla w_{ji}^n)^T, \\ F_i^{n+1} &= M((I + \nabla v_i^{n+1}) F_i^n). \end{aligned} \quad (13)$$

Notation	Meaning	Definition
E	Young's modulus	-
μ	Shear modulus	$\mu = \frac{E}{2(1+\nu)}$
λ	Lamé modulus	$\lambda = \frac{EV}{(1+\nu)(1-2\nu)}$

Table 4. Material Parameters.

Here, b denotes the B-spline degree, and Δx represents the Eulerian grid spacing. The calculation of the deformation adjustment mapping M and the Kirchhoff stress tensor K are detailed in the next subsection.

6.2. Physics Models

In this section, we provide the physics model details for the paper, and we show the relevant material parameters in Table 4. The employed physics models are adopted from [52, 63]. For the given material, the Kirchhoff stress tensor K is mapped by the material's corresponding elasticity model, and the deformation gradient F^E is mapped by the material's specific plasticity model.

Fixed Corotated Elasticity. The Fixed Corotated Elasticity model describes the behaviors of materials that undergo deformations with rotations and small elastic strains [18]:

$$K = 2\mu (F^E - R) (F^E)^T + \lambda (J - 1) J, \quad (14)$$

where $R = UV^T$ and $F^E = U\Sigma V^T$, and J is the determinant of F^E .

St. Venant-Kirchhoff Elasticity. St. Venant-Kirchhoff models materials that return to their original shapes after large deformations [22]:

$$K = U (2\mu \epsilon + \lambda \text{sum}(\epsilon)) V^T, \quad (15)$$

where $\epsilon = \log(\Sigma)$ and $F^E = U\Sigma V^T$.

Drucker-Prager Plasticity. Drucker-Prager Plasticity describes the behaviors of the materials that do not exhibit purely ductile behavior [22]:

$$F^E = UM(\Sigma)V^T, \quad (16)$$

$$M(\Sigma) = \begin{cases} 1 & \text{sum}(\epsilon) > 0 \\ \Sigma & \delta_\gamma \leq 0 \text{ and } \text{sum}(\epsilon) \leq 0 \\ \exp(\epsilon - \delta\gamma \frac{\hat{\epsilon}}{\|\hat{\epsilon}\|}) & \text{otherwise.} \end{cases} \quad (17)$$

Here, M is the deformation adjustment mapping, $\delta\gamma = \|\hat{\epsilon}\| + \alpha \frac{(d\lambda + 2\mu)\text{sum}(\epsilon)}{2\mu}$, $\alpha = \sqrt{\frac{2}{3}} \frac{2\sin\phi_f}{3-\sin\phi_f}$, ϕ_f is the friction angle, and $\hat{\epsilon} = \text{dev}(\epsilon)$.

von Mises Plasticity. von Mises Plasticity models the materials that will permanently deform when the stress reaches a certain threshold value [43]:

$$F^E = UM(\Sigma)V^T, \quad (18)$$

$$M(\Sigma) = \begin{cases} \Sigma & \delta\gamma \leq 0 \\ \exp(\varepsilon - \delta\gamma \frac{\hat{\varepsilon}}{\|\hat{\varepsilon}\|}) & \text{otherwise}, \end{cases} \quad (19)$$

where $\delta\gamma = \|\hat{\varepsilon}\|_F - \frac{KY}{2\mu}$, and KY is the yield stress.

6.3. Score Distillation Sampling

Score Distillation Sampling is a technique proposed in DreamFusion [37] that utilizes the 2D diffusion prior to optimize an image generator based on the probability density distillation. To achieve this, an image generator parameterized by parameters θ is represented as $g(\theta)$. To optimize over parameters θ such that the generated image $x = g(\theta)$ resembles a sampling from the pre-trained frozen 2D diffusion model, the SDS loss gradient for optimizing θ is formulated as:

$$\nabla_{\theta} L_{\text{SDS}}(\phi, x = g(\theta)) \triangleq \mathbb{E}_{t, \varepsilon} \left[w(t) (\hat{\varepsilon}_{\phi}(z_t; y, t) - \varepsilon) \frac{\partial x}{\partial \theta} \right], \quad (20)$$

where $\hat{\varepsilon}_{\phi}(z_t; y, t)$ is the predicted noise by the pre-trained 2D diffusion model with the text prompt y at the time step t , and ε is the true noise at the time step. $\frac{\partial x}{\partial \theta}$ is the derivative of the image generator's generated image with respect to its parameters θ , and $w(t)$ is a weighting function from DDPM [13]. This loss function aligns the scores (or gradients) of the image generator and the 2D diffusion model by optimizing the loss gradients with respect to θ , which can enable the use of the 2D diffusion prior to guide the generation of 3D models efficiently.

7. More Results

7.1. More Text-to-3D Physics Motion Results

We present additional text-to-3D physics-grounded motion results in Figure 10. These results demonstrate that our framework effectively generates 3D objects with high-quality appearances, accurate shapes, and realistic physics-driven motion for the given text prompts and material types. We also include the generated videos in the supplementary materials, please refer to the MP4 files in the videos folder or view them through the videos.html file.

7.2. More Qualitative Comparisons

The additional results of the compared methods, DreamPhysics [15] and Feature-Splatting [38], are presented in Figure 10 and Figure 11, respectively. Compared to our findings, we notice that DreamPhysics generates under-saturated colors, displaying muted and less convincing

Ours	DreamPhysics	Feature-Splatting
3.88	3.71	2.23

Table 5. Mean LAION aesthetic scores of the 5 object videos generated by all methods.

movements. For example, the jelly-like motion it produces for the pancakes is limited to minimal movements. Additionally, DreamPhysics's generated metal-like motion for the can model seems unnatural and less authentic. In contrast, Feature-Splatting produces over-saturated and inaccurate colors, with its generated motion being nearly undetectable.

7.3. More Quantitative Comparisons

The additional quantitative comparison results are shown in Table 5 for the object videos in Figure 10. The table shows that our framework achieves a mean LAION score of 3.88, outperforming the other methods. This indicates that our framework produces videos with higher visual quality compared to the other methods.

8. More Experiments

8.1. LLM-Prompt Refinement

Figure 12 demonstrates that the absence of LLM-Prompt Refinement (LLM-PR) can lead to uneven and over-saturated colors, as well as the lack of shadows, fine textures, and intricate details in the flower petals.

8.2. 3D Diffusion Prior Guidance

The results in Figure 13 illustrate that incorporating the 3D diffusion prior as guidance (3D Guidance) significantly improves the Gaussian Splatting process, enabling it to produce 3D objects with more precise geometrical shapes compared to those generated without this guidance.

8.3. Color Regularization

Our results in Figure 14 indicate that the absence of the color regularization (CR) on RGB values results in inaccurately rendered colors in the generated 3D object, as opposed to the more accurate results achieved with color regularization.

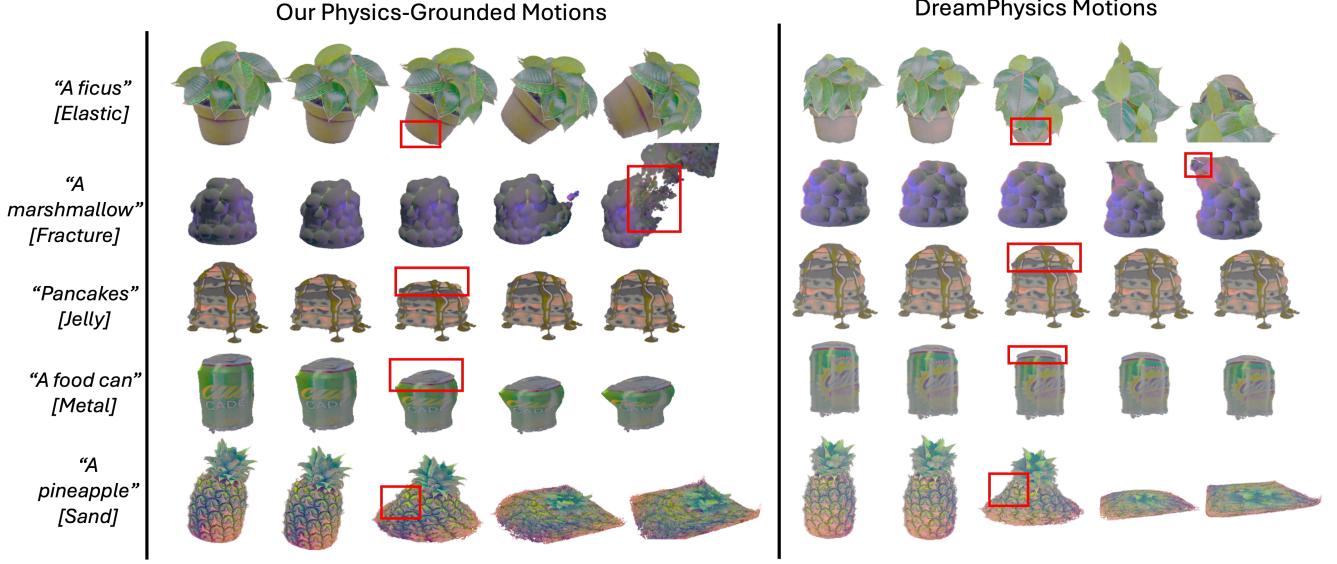


Figure 10. Additional text-to-3D physics-grounded motion results generated by our framework and the results generated by DreamPhysics using the 3D models provided by our framework.

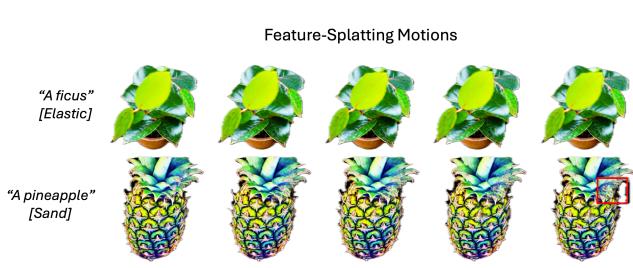


Figure 11. **Feature-Splatting Results.** Results generated by Feature-Splatting using the 3D models provided by our framework.

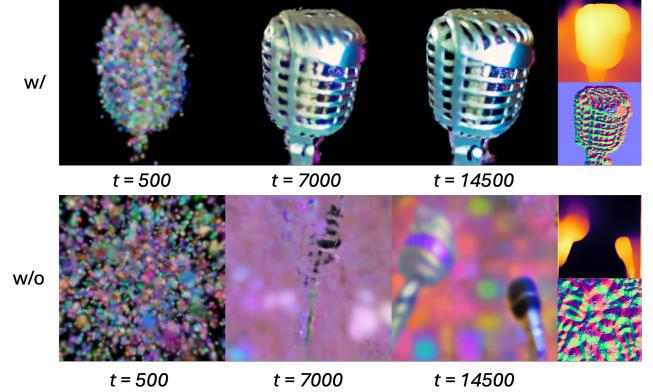


Figure 13. The impact of employing the 3D diffusion prior as GS shape guidance. Prompt: *A microphone*.

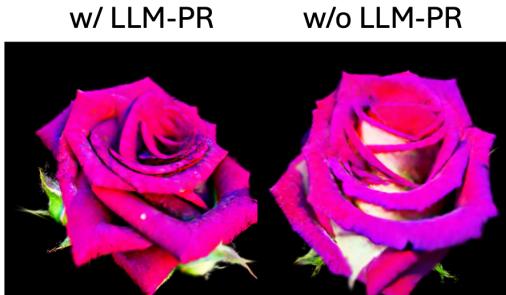


Figure 12. The impact of adopting LLM-prompt refinement. Prompt: *A rose*.

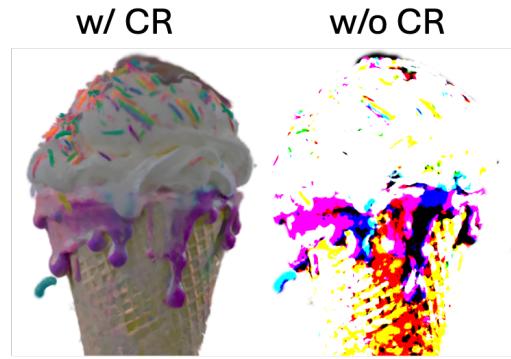


Figure 14. The effects of applying the color regularization on the RGB values. Prompt: *An ice-cream*.