Part-Tokenised Graph–Field Flow Matching for Coherent Physics-Grounded 3D Asset Generation

Jongmin Yu[1], Hyeontaek Oh[1], Zhongtian Sun[2],
Anoushka Harit[3], and Jinhong Yang[1,4,†]
[1] ProjectG.AI, Daejeon, Republic of Korea.
[2] School of Computing, University of Kent, Kent, CT2 7NZ, United Kingdom.
[3] Cancer Research UK, University of Cambridge, Cambridge CB2 0RE, United Kingdom
[4] Department of Medical Information Technology, Inje University, Kimhae, Republic of Korea.

***Abstract*— Physically grounded 3D asset generation aims to recover simulation-ready objects from a single image by combining geometric fidelity with physically meaningful attributes, including absolute scale, density, affordance, and articulation. We propose Part-Tokenised Graph–Field Flow Matching (PTGFFM), a two-stage framework that augments a frozen geometry-centric structural prior with a trainable part-aware physics branch. PTGFFM represents physical attributes as an unordered set of latent part tokens with existence prediction, and decodes them into dense physical fields and a kinematic graph conditioned on structured 3D features. To improve robustness to long-tailed object sizes, we regress absolute scale in a stabilised asinh space. A conditional flow-matching model is then trained to generate the physical token set from noise. On PhysXNet, PTGFFM improves several physical-property metrics over prior baselines while maintaining geometric fidelity comparable to the underlying structural prior. These results suggest that part-tokenised physical generation is a promising direction for simulation-ready 3D asset synthesis.**

I. Introduction

3D asset generation has recently advanced rapidly, driven by scalable generative models for single-view and multi-view 3D reconstruction [12]. Recent approaches [14], [10], [7], [3], [4] can synthesise visually compelling geometries and textures, but remain largely limited to appearance-oriented modelling, leaving important physical attributes underexplored. Such omissions pose a critical bottleneck for robust deployment in robotics, simulation, and embodied AI, where size, material, affordance, and kinematic structure fundamentally govern interaction and feasibility. Current image-to-3D pipelines [16], [15], [17], [1], [10], [7], [3], [4] often produced assets that are visually plausible yet lack the intrinsic physical properties required for rigorous physical simulation.

Recently, PhysX-3D [2] took an important step towards this goal by introducing PhysXNet and PhysXGen. PhysXNet is a physics-annotated 3D dataset with part-level supervision, which supports the evaluation of both geometric quality and physical-property prediction. PhysXGen is an end-to-end framework for generating physics-grounded 3D assets. It injects physical information into a pre-trained 3D structural space. Specifically, it jointly modelled the correlations between geometry and physical properties while preserving geometric fidelity via a frozen structural prior. This paradigm highlights both the importance and the difficulty of physical 3D generation.

Motivated by these insights, we propose a new framework, Part-Tokenised Graph-Field Flow Matching (PTGFFM), which aims to generate physically grounded 3D assets from a single image. Similar to PhysXGen, PTGFFM basically decomposes the problem into a frozen structural branch and a trainable physics branch. The structural branch leverages a powerful pre-trained geometry-centric generator, such as TRELLIS [16], to produce high-fidelity geometry.

The main difference between the proposed PTGFFM and PhysXGen lies in how physical information is represented and decoded. Rather than modelling physical attributes in a single monolithic latent, we represent them as an unordered set of latent part tokens with explicit existence prediction, and decode these tokens into dense physical fields and a kinematic graph conditioned on structured 3D features. In other words, while prior work such as PhysXGen constructs a dual-branch architecture yielding dense spatial predictions, PTGFFM diverges by encoding physical properties into a permutation-invariant, discrete set of abstract part tokens. This design is intended to improve part-level reasoning and variable-cardinality modelling whilst remaining compatible with a frozen structural prior.

Our contributions can be summarised as follows. First, we propose PTGFFM, a unified framework that couples part-aware physics generation with a frozen structural prior. Second, we introduce a Graph-Field decoder that grounds latent part tokens into dense physical fields and kinematic graphs conditioned on structured 3D features. Third, we adopt conditional flow matching to generate physical tokens from noise, enabling high-quality and diverse physics-conditioned asset synthesis. Extensive experiments demonstrate that our method achieves strong performance across multiple physical attributes while retaining impressive geometric fidelity.

† denotes the corresponding author.

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP)-Innovative Human Resource Development for Local Intellectualisation program grant funded by the Korea government (MSIT) (IITP-2026-RS-2024-00436773)

II. Part-Tokenised Graph–Field Flow Matching

A. Architectural details

The framework comprises a frozen structural branch and a trainable physics branch. The structural branch predicts a structured 3D latent that encodes the object geometry, whereas the physics branch predicts part-aligned physical attributes and articulation conditioned on this latent. This design is intended to preserve the geometry provided by the pre-trained structural prior whilst enriching it with physically interpretable information.

First, the structural branch extracts an image feature v from an input image I and converts it into a structured 3D latent grid $z_s \in \mathbb{R}^{N\times d_s}$:

$$v = \mathcal{E}_{vis}(I), \qquad z_s = \mathcal{G}_{str}(v). \tag{1}$$

In our work, $\mathcal{E}_{vis}$ and $\mathcal{G}_{str}$ are defined by DINOv2 [13] and TRELLIS [16], respectively. In practice, structured latent (z_s) is represented as a set of latent vectors attached to the active subset of a 3D voxel grid, i.e., voxels that intersect the object surface, following the structured latent paradigm used by TRELLIS. N is the number of active 3D sites (and varies per instance), and we maintain the corresponding spatial coordinates to define neighbourhood relations and spatial decoding operations over the latent sites. Prior work on 3D generation often focuses mainly on geometry [6], [9], [8], [17], [1]. Our key premise is to retain the geometric reliability of a pre-trained structured prior and attach a compact, interpretable, part-aligned physical representation that can be generated efficiently using flow-matching-based conditional sampling

The physics branch describes the object using a small set of *part tokens*. Each token functions as a compact latent embedding that encodes the distinct physical properties and spatial assignments of an individual structural component. We follow the physical representation of PhysXGen [2]. Input annotations include the absolute scale $P_{scale} \in \mathbb{R}^3$, affordance priorities $P_{aff} \in \mathbb{R}^{K\times 1}$, densities $P_\rho \in \mathbb{R}^{K\times 1}$, and kinematic information P_{kin}.

A physical encoder $\mathcal{E}_{phy}$ maps these annotations to K fixed token slots:

$$Z_{\text{phy}} = \mathcal{E}_{phy}(P_{scale}, P_{aff}, P_\rho, P_{kin}), \tag{2}$$

where $Z_{\text{phy}} \in \mathbb{R}^{K\times d_t}$ and d_t indicates the dimensionality of Z_{phy}. We use a fixed number of slots even though a real object may have fewer parts. To handle this, each token predicts an existence score

$$\hat{o}_k = \sigma(h_{\text{exist}}(Z_{\text{phy},k})), \tag{3}$$

which tells the model whether the slot k is actually needed or not. We adopt a fixed number of part-token slots K and treat the tokens as an unordered set, mirroring the set-prediction formulation where a model emits a fixed number of hypotheses and learns permutation invariance through bipartite matching. The existence score ($\hat{o}_k$) plays the role of a learned "present vs. absent" indicator, allowing the model to represent objects with fewer than K parts while keeping the architecture and training objective unchanged.

The key challenge is that tokens are abstract, while the object shape lives on a 3D grid. Our Graph–Field decoder $\mathcal{D}_{phy}$ bridges these two views. Conditioned on the tokens and the structural grid, it predicts

$$(\alpha, \hat{F}_{phy}, \hat{P}_{kin}, \hat{P}_{scale}) = \mathcal{D}_{phy}(Z_{\text{phy}}, z_s). \tag{4}$$

Here, $\alpha \in \mathbb{R}^{K\times N}$ is a soft assignment matrix, $\hat{F}_{phy} = (\hat{\rho}, \hat{a})$ is a two-channel physical field on the 3D grid, $\hat{P}_{kin}$ contains the predicted articulation graph and its parameters, and $\hat{P}_{scale} \in \mathbb{R}^3$ is the global scale prediction. The matrix α is especially important: for every site n, we normalise the tokens so that $\sum_{k=1}^{K} \alpha_{k,n} = 1$. In simple terms, $\alpha_{k,n}$ says how much token k is responsible for location n.

We implement α as a differentiable correspondence between a fixed-size token set and the structured latent sites. Concretely, for each site n, we compute unnormalized assignment logits ($s_{k,n}$) from token- and site-level features (e.g., via dot-product attention or an MLP over ($[Z_{\text{phy},k}, z_{s,n}, x_n]$)), and apply a per-site softmax to enforce ($\sum_k \alpha_{k,n} = 1$). This design yields permutation-invariant part assignment while remaining fully differentiable, enabling end-to-end training of both token attributes and their spatial support.

From each token we decode part-level attributes $\hat{p}_k = (\hat{\rho}_k, \hat{a}_k)$ for density and affordance. The final physical field on the 3D grid is then a weighted mixture:

$$\hat{\rho}_n = \sum_{k=1}^{K} \alpha_{k,n}\hat{\rho}_k, \qquad \hat{a}_n = \sum_{k=1}^{K} \alpha_{k,n}\hat{a}_k. \tag{5}$$

This makes the model straightforward to interpret: The soft assignment matrix projects the discrete token attributes onto the continuous spatial grid via a learned interpolation mechanism. As a result, the output contains both continuous fields (such as density over space) and discrete structure (such as which parts are linked by joints).

B. Training and Inference

To effectively coordinate the entire framework, we decouple the training process into two distinct stages. The first stage focuses on representation learning for the physics tokens and their spatial decoding, whilst the second stage trains the generative model to sample these tokens from noise.

Stage I-Learning the Physical Latent and Decoder: Given an image I, we first compute v and z_s using the frozen structural branch. Let x_{phy} denote the aligned ground-truth physical annotations on this structured grid: $x_{\text{phy}} = [P_{scale}, P_{aff}, P_\rho, P_{kin}]$. We then encode x_{phy} into latent tokens with the encoding part of the physical encoder $\mathcal{E}_{phy}$. For token k, the encoder predicts a Gaussian distribution

$$\mathcal{E}^{enc}_{phy}(t_k \mid x_{\text{phy}}) = \mathcal{N}(\mu_k, \text{diag}(\sigma_k^2)), \tag{6}$$

and we sample $t_k = \mu_k + \sigma_k \odot \epsilon_k$ with $\epsilon_k \sim \mathcal{N}(0, I)$. Stacking these samples gives the target token set $Z_{\text{phy},0} \in \mathbb{R}^{K\times d_t}$ used in Stage II. A KL term keeps the latent space close to a standard Gaussian so that it can later be sampled smoothly.

Because tokens form an unordered set, token 1 does not always correspond to the same semantic part among different

objects. We therefore match predicted tokens with ground-truth parts using bipartite (Hungarian) matching [5]. A good match should cover the right region in space, predict the right density and affordance, and be confident that the token exists. We use the cost

$$C(i,k) = \mathrm{CE}(\alpha_{k,\cdot}, m_{i,\cdot}) + \|\hat{\rho}_k - \rho_i\|_2^2 + \|\hat{a}_k - a_i\|_2^2 - \log(\hat{o}_k), \tag{7}$$

where $m_{i,\cdot}$ is the voxel mask of part i, (ρ_i, a_i) are its density and affordance labels, and $\mathrm{CE}(\cdot)$ indicates cross-entropy function.

After matching, we apply standard supervision terms: a mask loss for the soft assignments, an existence loss for active versus inactive tokens, regression losses for density and affordance, and kinematic losses for edge existence, joint type, and continuous joint parameters. We also predict the absolute object scale. We regress scale in an inverse-hyperbolic-sine space because $(\mathrm{asinh}(\cdot))$ behaves similarly to a logarithm for large positive values while remaining well-behaved near zero, which can stabilize regression when object scales span multiple orders of magnitude:

$$L_{\mathrm{scale}} = \left\| \mathrm{asinh}(\hat{P}_{\mathrm{scale}}/c) - \mathrm{asinh}(P_{\mathrm{scale}}/c) \right\|_2^2, \tag{8}$$

with $c = 100$ in all experiments. It is empirically selected on the basis of the transition point of the object scale distribution in the PhysXNet dataset. Note that because log-like transformations (including (asinh)) can be sensitive to the units or scaling of the target variable, we fix a single constant (c) across all experiments and report it explicitly for reproducibility.

To improve spatial coherence, we impose an edge-aware smoothness prior on the decoded physical field. This term encourages adjacent 3D sites to produce similar density and affordance predictions, while assigning smaller penalties across sharp geometric discontinuities inferred from occupancy, signed distance, or the structural latent. As a result, the model suppresses spurious local fluctuations without blurring across semantically meaningful part boundaries. The complete Stage I objective is

$$\begin{aligned} L_{\mathrm{vae}} = {} & \lambda_{\mathrm{mask}} L_{\mathrm{mask}} + \lambda_{\mathrm{exist}} L_{\mathrm{exist}} + \lambda_{\rho} L_{\rho} + \lambda_a L_a \\ & + \lambda_{\mathrm{kin}} L_{\mathrm{kin}} + \lambda_{\mathrm{scale}} L_{\mathrm{scale}} + \lambda_{\mathrm{coh}} L_{\mathrm{coh}} + \lambda_{\mathrm{KL}} L_{\mathrm{KL}}. \end{aligned} \tag{9}$$

Stage II-Generating part tokens by flow matching: Once Stage I has defined a useful token space, we learn how to generate tokens directly from noise. The idea is straightforward: start from random noise, then learn how to gradually move it towards a valid token set that matches the input image and 3D structure. We use conditional flow matching for this.

The generation process is formulated as an initial-value problem, in which a neural vector field, parameterised by an ordinary differential equation, maps a simple Gaussian prior to the complex empirical data distribution over valid physical token sets.

Let $\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$ be noise and let $Z_{\mathrm{phy},0}$ be the Stage I target tokens. For a random time $t \sim \mathcal{U}(0,1)$, we place ourselves on the straight line between them:

$$Z_{\mathrm{phy},t} = (1-t) Z_{\mathrm{phy},0} + t\boldsymbol{\epsilon}. \tag{10}$$

A neural vector field f_θ then learns which direction to move at each time step, conditioned on the image and structure features $c \doteq (v, z_s)$. The training objective is

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}_{Z_{\mathrm{phy},0}, \boldsymbol{\epsilon}, t} \left\| f_\theta(Z_{\mathrm{phy},t}, t; c) - (\boldsymbol{\epsilon} - Z_{\mathrm{phy},0}) \right\|_2^2. \tag{11}$$

Intuitively, f_θ learns the velocity that turns noise into a meaningful set of part tokens.

In other words, this is a conditional flow-matching objective: we define a probability path between data tokens and Gaussian noise using a linear interpolant, and regress a neural vector field toward the path's velocity. This training reduces to a simple regression problem for learning a continuous-time generative flow, which can be sampled by integrating an ODE with standard numerical solvers rather than running a long Markov chain.

Inference. At test time, the structural branch first produces the 3D shape latent z_s and the geometric asset $\hat{\mathcal{X}}$. The physics branch starts from Gaussian noise $Z_{\mathrm{phy},1} \sim \mathcal{N}(0, I)$ and integrates the reverse-time ODE from $t = 1$ to $t = 0$, yielding the final tokens Z_{phy}. Tokens with low existence scores $\hat{o}_k < \tau_o$ are discarded, and the remaining assignment weights are renormalised. We then decode the surviving tokens together with z_s. For kinematics, we enforce a valid directed forest at decode time by allowing at most one parent per token and removing cycles, which guarantees zero invalid-graph rate by construction. The final output is a 3D asset that includes geometry, continuous physical fields, an articulation graph, and the global physical scale.

III. Experiments

A. Dataset and implementation details

We evaluate our approach on the PhysXNet dataset [2], which provides 26K physics-annotated 3D objects with part-level supervision. We follow the standard split of 24,000 samples for training, 1,000 for validation, and 1,000 for testing. We use the same splits proposed by Cao *et al.* [2]. Performance is assessed along two axes: physical property prediction and geometric fidelity. For geometric evaluation, we report rendered-view PSNR for appearance, as well as Chamfer Distance (CD) and F-score (FS) for geometry. For physical properties, we evaluate absolute scale, material (via density), affordance, kinematics, and function descriptors. kinematic quality is summarised by coverage (COV) and minimum matching distance (MMD), where the pairwise distance is the instantiation distance of [7]. Higher COV and lower MMD indicate better agreement between the predicted and reference articulation distributions. Following [2], absolute scale is measured by Euclidean distance; density/material and affordance are evaluated using PSNR on rendered property maps; kinematics is evaluated using instantiation distance [11]; and function descriptors are evaluated using PSNR computed on cosine-similarity score maps.

TABLE I

QUANTITATIVE COMPARISON OF EXISTING METHODS ON THE TEST SETS OF PHYSXNET DATASET. THERE ARE TWO TYPES OF EVALUATIONS: STRUCTURAL AND PHYSICAL PROPERTY EVALUATIONS. PHYSPRE REPRESENTS A SEPARATE PHYSICAL PROPERTY PREDICTOR AFTER TRELLIS.

Methods	Structural geometry			Physics properties					
	PSNR ↑	CD ↓	F-Score ↑	Absolute scale ↓	Material ↑	Affordance ↑	Kinematics COV ↑	Kinematics MMD ↓	Description ↑
TRELLIS [16]	24.31	13.2	76.9	–	–	–	–	–	–
TRELLIS + PhysPre	24.31	13.2	76.9	13.21	8.63	7.23	0.24	0.12	6.55
PhysXGen [2]	**24.53**	12.7	**77.3**	7.24	13.01	11.30	0.33	0.08	10.11
PTGFFM (Ours)	24.51	**12.6**	77.2	**6.92**	**13.96**	**12.02**	**0.35**	**0.08**	**11.08**

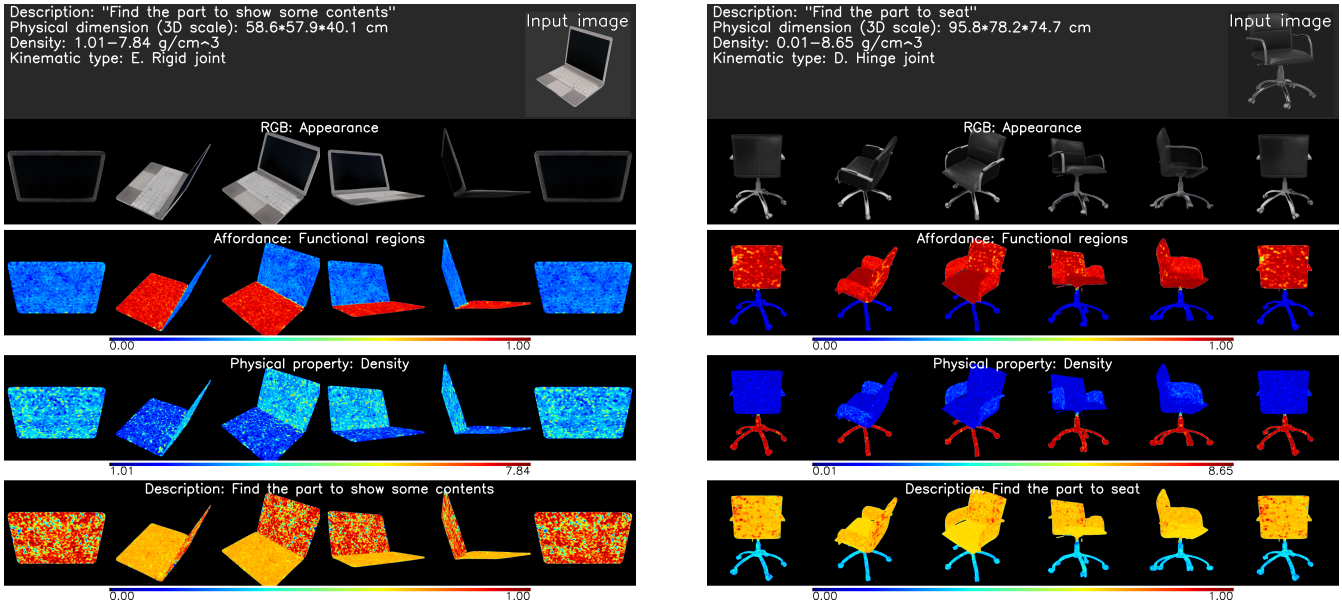

Fig. 1. The visualisation results of the proposed PTGFFM.

For training, we optimise Stage I and Stage II models separately using AdamW with a learning rate of 1×10^{-4}. We use $K = 262144$ token slots with token dimension $d_t = 8$. The loss weights in Eq. (9) are (λ_{mask}, λ_{exist}, λ_{ρ}, λ_{a}, λ_{kin}, λ_{scale}, λ_{coh}, λ_{KL} = (1, 1, 0.01, 0.1, 1, 1, 1, 1e-6). The batch size is 4. At inference, the existence threshold is τ_o = 0.5. Our framework is trained on 2 NVIDIA A6000 GPUs.

IV. COMPARISON WITH EXISTING SOTA METHODS

Table I contrasts our approach with (i) TRELLIS, (ii) TRELLIS + PhysPre, and (ii) PhysXGen [2]. The TRELLIS, TRELLIS + PhysPre, and PhysXGen rows are taken directly from Cao *et al.* [2], which uses the same test split and evaluation protocol; PTGFFM is evaluated under the same protocol for a controlled comparison. Overall results in Table I, our model improves physical-property metrics whilst remaining competitive on geometry.

Compared with TRELLIS + PhysPre, our method improves every physical indicator by a notable margin: the absolute-scale error decreases from 13.21 to 6.92, material and affordance scores rise from 8.63/7.23 to 13.96/12.02, and the kinematic statistics better match the target distribution (COV increases from 0.24 to 0.35, whilst MMD decreases from 0.12 to 0.08). Relative to PhysXGen, we further reduce scale error (6.92 vs. 7.24) and improve material, affordance, and description scores (13.96, 12.02, and 11.08, respectively), whilst matching the best kinematic MMD (0.08) and slightly increasing coverage (COV 0.35 vs. 0.33). These results indicate that the proposed architectural changes strengthen the coupling between structure and physics, leading to more reliable physical predictions.

On the structural metrics, PTGFFM remains comparable to PhysXGen (PSNR 24.51 vs. 24.53; F-score 77.2 vs. 77.3) whilst achieving a slightly lower Chamfer distance (12.6 vs. 12.7). Compared with TRELLIS + PhysPre, PTGFFM improves CD and maintains higher PSNR and F-score. Overall, geometric quality is preserved, while gains in physical-property fidelity are consistent across categories. On geometry, PTGFFM remains comparable to the TRELLIS structural prior and to PhysXGen. The observed differences are small and are not the main focus of this work; the primary result is the improvement in several physical-property metrics whilst retaining the geometry provided by the structural prior.

Fig. 1 shows examples of the outputs of the PTGFFM. PTGFFM generates high-fidelity, simulation-ready 3D assets by successfully coupling detailed physical attributes with explicit spatial geometry. The RGB appearance rows confirm that the trainable physics branch preserves the high-quality geometric priors of the frozen structural branch across multiple viewing angles. The affordance and density maps

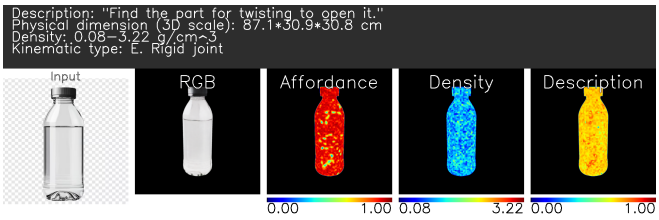

Fig. 2. Visualisation of a transparent object (a water bottle), illustrating a representative failure case.

demonstrate the framework's ability to ground abstract part tokens into dense, continuous physical fields aligned with the object's structure. Global properties are likewise well-captured; the framework predicts precise absolute 3D scales, such as the $58.6 \times 57.9 \times 40.1$ cm dimensions for the laptop, and identifies discrete kinematics including hinge or rigid joint types. These qualitative results underscore that the proposed architecture strengthens the coupling between structure and physics, yielding reliable, interaction-ready digital assets suitable for robotics and embodied AI.

V. Limitation and Discussion

Figure 2 shows a representative failure on a transparent object. Because the structural prior is primarily appearance-driven, refractive boundaries can lead to inaccurate part localisation, which in turn causes leakage in the predicted physical fields and an incorrect rigid-joint prediction. This example indicates that transparent and highly refractive objects remain challenging for the current system.

Consequently, the structural generator fails to resolve accurate geometric boundaries, causing the physics branch to miss fine-grained part decomposition. Additionally, the generated affordance, density, and description maps incorrectly bleed across the entire monolithic structure rather than cleanly isolating the bottle cap. This failure case underscores the critical need to integrate material-aware geometric priors and enhanced visual encoders capable of parsing refractive surfaces, ensuring physical fields remain accurate even when standard visual cues are compromised.

VI. Conclusion

In this paper, we introduced Part-Tokenised Graph–Field Flow Matching (PTGFFM), a two-stage framework for physically grounded 3D asset generation from a single image. By combining a frozen structural prior with a part-aware physics branch, the method predicts dense physical fields, articulation, and absolute scale whilst retaining the geometry provided by the structural prior. Experiments on PhysXNet indicate improvements in several physical-property metrics relative to prior baselines, with geometry remaining broadly comparable to the underlying structural model. Future work will focus on transparent materials, more complex articulated objects, and stronger material-aware structural priors.

References

[1] Z. Cao, Z. Chen, L. Pan, and Z. Liu. Collaborative multi-modal coding for high-quality 3d generation. *arXiv preprint arXiv:2508.15228*, 2025.

[2] Z. Cao, Z. Chen, L. Pan, and Z. Liu. Physx-3d: Physical-grounded 3d asset generation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[3] Z. Cao, F. Hong, T. Wu, L. Pan, and Z. Liu. Large-vocabulary 3d diffusion model with transformer. *arXiv preprint arXiv:2309.07920*, 2023.

[4] Z. Cao, F. Hong, T. Wu, L. Pan, and Z. Liu. Difftf++: 3d-aware diffusion transformer for large-vocabulary 3d generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[5] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers, 2020.

[6] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[7] Z. Chen, J. Tang, Y. Dong, Z. Cao, F. Hong, Y. Lan, T. Wang, H. Xie, T. Wu, S. Saito, et al. 3dtopia-xl: Scaling high-quality 3d asset generation via primitive diffusion. *arXiv preprint arXiv:2409.12957*, 2024.

[8] J. Collins, S. Goel, K. Deng, A. Luthra, L. Xu, E. Gundogdu, X. Zhang, T. F. Y. Vicente, T. Dideriksen, H. Arora, et al. Abo: Dataset and benchmarks for real-world 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21126–21136, 2022.

[9] M. Deitke, D. Schwenk, J. Salvador, L. Weihs, O. Michel, E. VanderBilt, L. Schmidt, K. Ehsani, A. Kembhavi, and A. Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13142–13153, 2023.

[10] F. Hong, J. Tang, Z. Cao, M. Shi, T. Wu, Z. Chen, S. Yang, T. Wang, L. Pan, D. Lin, et al. 3dtopia: Large text-to-3d generation model with hybrid diffusion priors. *arXiv preprint arXiv:2403.02234*, 2024.

[11] J. Lei, C. Deng, B. Shen, L. Guibas, and K. Daniilidis. Nap: Neural 3d articulation prior. *arXiv preprint arXiv:2305.16315*, 2023.

[12] A. Melnik, B. Alt, G. Nguyen, A. Wilkowski, M. Stefańczyk, Q. Wu, S. Harms, H. Rhodin, M. Savva, and M. Beetz. Digital twin generation from visual data: A survey, 2026.

[13] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[14] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

[15] J. Tang, Z. Chen, X. Chen, T. Wang, G. Zeng, and Z. Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18. Springer, 2024.

[16] J. Xiang, Z. Lv, S. Xu, Y. Deng, R. Wang, B. Zhang, D. Chen, X. Tong, and J. Yang. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506*, 2024.

[17] J. Xu, W. Cheng, Y. Gao, X. Wang, S. Gao, and Y. Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024.