PRISM: ARTICULATED OBJECT MODELING WITH PRECISE STRUCTURAL CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

Articulated objects are fundamental to robotics, simulation, and interactive virtual environments, where generated assets must satisfy not only visual realism but also structural and functional validity. Although recent generation methods have made substantial progress through 2D diffusion priors, sparse-view reconstruction, and native 3D generative models, their semantic or appearance-level conditions leave part topology and articulation underdetermined. Consequently, visually plausible assets can exhibit incorrect connectivity, joint placement, or functional organization. We present Prism, an Analytic-Concept-guided framework for articulated object modeling with strict structural control, together with an automated Analytic Concept construction module that instantiates the executable concepts used for conditioning. Prism uses Analytic Concept as a procedural control interface that specifies part geometry, topological relations, articulation constraints, and affordance functions in an executable form. Given a target brief, procedural templates, and a small set of valid instances, the construction module infers parameter dependencies, proposes an instance in dependency order, and repairs only the parameters implicated by failed execution checks. A validated concept and an appearance reference image are then converted into a multi-modal prior representation from concept-rendered images, concept-derived language descriptions, and image-based appearance features. Adaptive modulation and cross-attention integrate these complementary modalities before a conditional diffusion transformer maps the resulting representation into a latent space. The predicted latent is then decoded into a mesh that preserves the specified articulated organization while maintaining realistic geometric details. We evaluate Prism on PartNet-Mobility and downstream robotic manipulation tasks. Under their respective input conditions, comparisons with general-purpose and articulated-generation baselines show improved geometric quality and structural fidelity. Controlled condition ablations further show that gains depend on how Prism integrates the explicit concept with appearance evidence, rather than on structural specification alone. Evaluating automatically constructed concepts across unseen categories remains a separate question.

1 INTRODUCTION

Articulated assets are increasingly important for robotics, virtual reality, augmented reality, and interactive simulation. Unlike static objects, articulated objects must represent not only external geometry and appearance, but also part organization, articulation layout, and affordance-related structure. For example, a generated cabinet may appear visually plausible, but it can still be functionally invalid if the number of doors is incorrect, the handles are misplaced, or the movable parts do not follow the intended articulation. Manually creating such assets is time-consuming, costly, and difficult to reuse Mo et al. (2021). Therefore, generating articulated assets from images, text, or other conditions has become an active research direction.

Existing generation methods can be broadly grouped into three paradigms. The first paradigm lifts pre-trained 2D diffusion models into 3D representations through score distillation Poole et al. (2023); Lin et al. (2023); Tang et al. (2024). The second paradigm reconstructs objects from sparse-view images Mildenhall et al. (2021); Chan et al. (2022); Hong et al. (2024). The third paradigm directly models the distribution of objects in 3D space, conditioned on text or images Li et al. (2025); Chen et al. (2025b); Xiang et al. (2025). These methods have achieved substantial progress in visual realism

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.

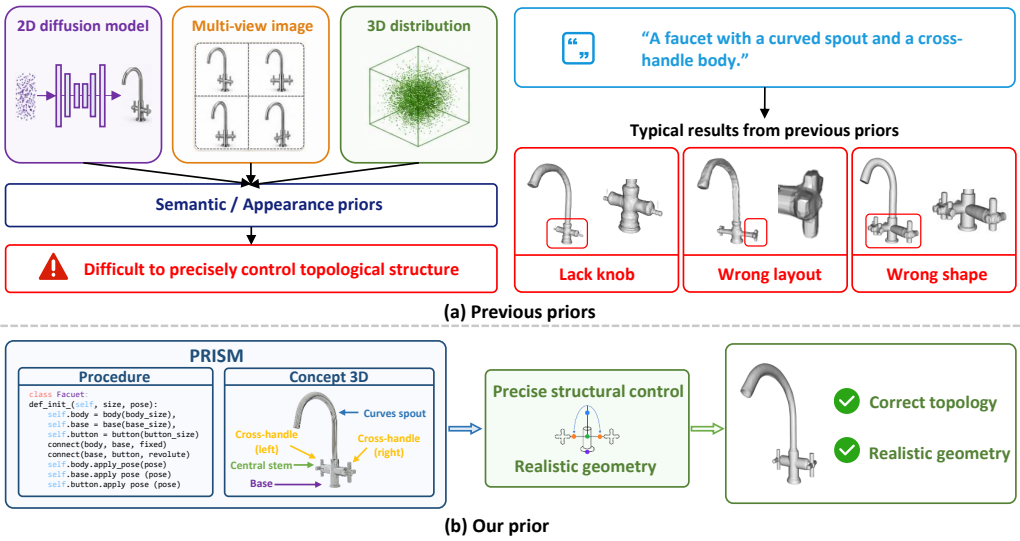


Figure 1: Comparison between the proposed method and previous generation routes. (a) Previous types of priors have limited ability to control topological structures. (b) Compared with previous methods, Prism can better preserve both geometric details and topological structures.

and geometric detail. However, most of them rely primarily on semantic or appearance-oriented conditions, which are insufficient for precise control over articulated structures. As illustrated in Figure 1, such methods may generate plausible geometry while failing to preserve specific structural requirements, such as part connectivity, relative configuration, articulation layout, or the exact number of repeated components.

This limitation is especially consequential for articulated objects because their usability depends on fine-grained structural correctness. A single image or natural-language prompt can suggest category and appearance, yet the part hierarchy, metric layout, joint type, and interaction region often remain ambiguous. Methods that infer these properties from visual evidence can therefore propagate an early structural error into an otherwise convincing shape. Coarse inputs such as categories or connectivity graphs reduce ambiguity but leave geometric and functional relations incompletely specified. Precise generation thus requires both an explicit account of the intended structure and a mechanism that carries it through to the final geometry.

To address this challenge, we introduce Analytic Concept as a control interface for articulated objects generation. Analytic Concept, derived from ConceptFactory Sun et al. (2024b), represents an articulated object through procedural definitions of parts geometric, topological relations, articulation constraints, and affordance-related functions. As shown in Figure 2, an Analytic Concept can be instantiated as conceptual geometry, rendered into concept images, and converted into natural-language descriptions. Compared with semantic prompts or coarse graphs, Analytic Concept provides a more precise and reusable way to specify how functional parts are arranged, connected, and used.

Prism focuses on articulated objects whose functional organization can be specified procedurally. ConceptFactory composes reusable part templates and provides an optimizer to fit selected parts to observed geometry Sun et al. (2024b); assembling a compatible template and configuring its instances nevertheless remain distinct tasks. Our automated construction module addresses repeated instantiation once a compatible assembly exists. Given that template, a target brief, and valid examples, it derives parameter dependencies and repairs candidates using executable checks of geometry, joints, and affordances. Primitive and template design remain human-authored, while the accepted instance enters the unchanged generator.

Based on this motivation, we propose Prism, an Analytic-Concept-guided framework for articulated object generation, preceded by automated Analytic Concept construction. The construction module converts a target brief into a validated instantiation of an available procedural template; Prism then uses that program for structural control and an appearance reference image for shape-detail

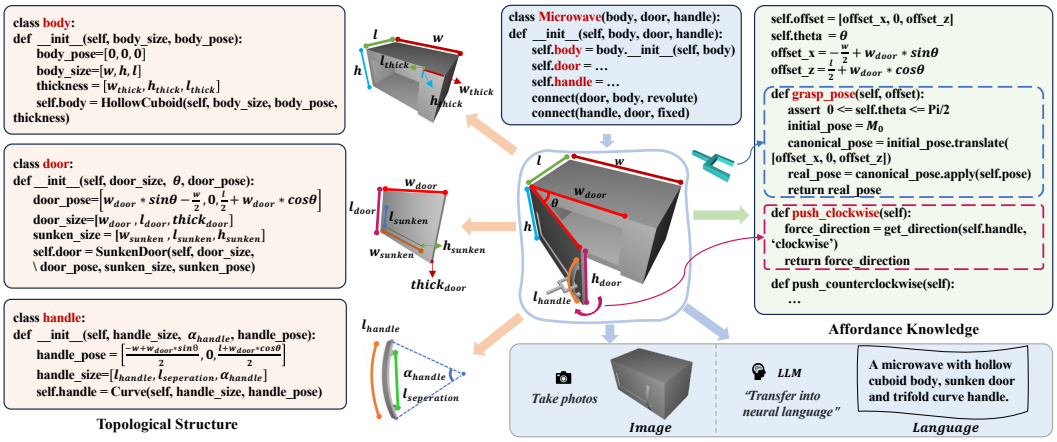


Figure 2: Illustration of Analytic Concept. Analytic Concept describes topological structures and affordance knowledge through procedural language. It can be instantiated as a corresponding 3D model, rendered into images with selected camera parameters, or converted into natural language via LLM.

and appearance cues. The downstream generator comprises Prior Representation Construction, Shape Encoding, and 3D Space Mapping. Concept-rendered images and concept-derived language descriptions encode structural information, while image-based features encode visual details. These features are fused through AdaLN Perez et al. (2018) modulation and cross-attention. Shape Encoding supplies real-object latents for supervision, and a conditional diffusion transformer maps the fused representation into the pre-trained 3D latent space.

The Analytic Concept resolves the structural ambiguity left by semantic and visual cues, while the learned fusion links its rendered and textual descriptions to the appearance reference before latent generation. This coupling matters because a structural specification alone does not determine how faithfully a mesh generator will realize it. Prism is therefore suited to interaction-oriented assets, where topology, articulation, and affordance consistency must coexist with realistic geometry.

Our contributions are summarized as follows. (1) We formulate executable Analytic Concepts as an interface for precise control of part geometry, topology, articulation, and affordances. (2) We introduce an automated construction module that derives argument dependencies from template evidence and couples dependency-ordered instantiation with scoped repair under executable geometric, articulation, and affordance checks. (3) We propose a multi-modal generator that maps concept-rendered images, concept-derived text, and appearance features into a pre-trained 3D latent space. (4) We evaluate the concept-conditioned generator on visual generation and robotic manipulation; these experiments do not by themselves establish the generalization or efficiency of automatically constructed concepts.

2 Related Work

2.1 3D Asset Generation

3D generation includes score-distillation optimization (Poole et al., 2023; Lin et al., 2023), sparse-view reconstruction (Siddiqui et al., 2024; Xie et al., 2024; Wang et al., 2023; Li et al., 2024a), and native 3D generators (Zheng et al., 2023; Chen et al., 2025c; Sanghi et al., 2022). Recent latent models such as Dora (Chen et al., 2025b) and Trellis (Xiang et al., 2025) improve detail and scale, but their semantic or visual conditions do not uniquely determine part topology, joints, or functional interfaces.

2.2 Articulated Generation

Articulated reconstruction estimates geometry and motion from point sequences, implicit fields, or multi-view observations (Mu et al., 2021; Tseng et al., 2022; Liu et al., 2023; Jiang et al., 2022;

Heppert et al., 2023). Generative methods use connectivity graphs or categories (Lei et al., 2023; Liu et al., 2024b), while URDFormer (Chen et al., 2024) regresses simulation assets from images. When articulation must be inferred from incomplete observations, errors in part segmentation, connectivity, or joint estimates can constrain the validity of the resulting asset. Prism conditions generation on an executable concept that explicitly represents structure and function, and its automated construction module expands valid instances of this interface.

3 METHOD

3.1 OVERVIEW

As shown in Figure 3, automated Analytic Concept construction instantiates and verifies a procedural specification. Prism fuses its rendered and textual representations with an appearance image, then maps the fused condition into a latent mesh representation through Prior Representation Construction, Shape Encoding, and 3D Space Mapping.

3.2 ANALYTIC CONCEPT FOR PRECISE STRUCTURAL CONTROL

An Analytic Concept composes elementary geometry templates into higher-level part templates with explicit spatial and functional relations (Appendix A.5). A compatible composite template specifies the part organization, joints, and affordances; continuous dimensions and poses and discrete repeat counts determine an instance. Prism renders the resulting program into conceptual images and verbalizes it for a text encoder; an aligned real-world image supplies appearance detail. The learned conditioning pathway must preserve the specified organization in the generated shape. Reported generator results use existing ConceptFactory annotations, whereas automated construction is defined for additional instances of an available composite template.

3.3 AUTOMATED ANALYTIC CONCEPT CONSTRUCTION

The quality of concept-conditioned generation depends on the validity and coverage of its input concepts. ConceptFactory already assists parameter fitting for a selected part when a target mesh is available Sun et al. (2024b). Our setting is different: given a target brief b, a compatible, pre-assembled template T, and a small evidence set D_T of valid parameter instances, the module seeks an object-level assignment that jointly satisfies dimensions, poses, part counts, articulation, and affordance relations. It returns an executable instantiation $C^* = T(x^*)$, or abstains. Template and seed preparation remain human work; ConceptFactory reports about seven minutes per observed object for its distinct assisted workflow Sun et al. (2024b) (Appendix A.5.3). The accepted instance supplies Prism's existing conditioning interface without modifying the generator.

Template-grounded constraint induction. The construction module reads the template signature and construction code together with D_T. It organizes arguments into a dependency graph: parent dimensions precede openings and attached parts; part counts precede repeated-element spacing; and supporting geometry precedes joint frames and interaction regions. It then derives a constraint set $\mathcal{Q}_T$ containing declared domains, inequalities observed in the construction logic, and cross-part compatibility predicates. These constraints are attached to parameter scopes. Consequently, a failure concerning a handle attachment can expose the handle pose and its supporting panel dimensions without opening unrelated parameters for revision. The examples provide feasible scale and co-variation evidence, while the executable template remains the authority on syntax and construction.

Constrained instantiation and execution audit. Conditioned on b, the construction module proposes parameters in a topological order of the dependency graph. The template is executed to obtain parts, their hierarchy, and any declared kinematics and affordances. An audit function V_T evaluates successful construction and rendering, non-degenerate geometry, required connectivity, containment and attachment relations, joint-frame and motion-range validity, collision over sampled articulation states, consistency between affordance annotations and their functional parts, and the parts and functions explicitly requested in b. Its output is a set of typed residuals rather than a single natural-language judgment. At iteration k, only parameters in the union of scopes associated with

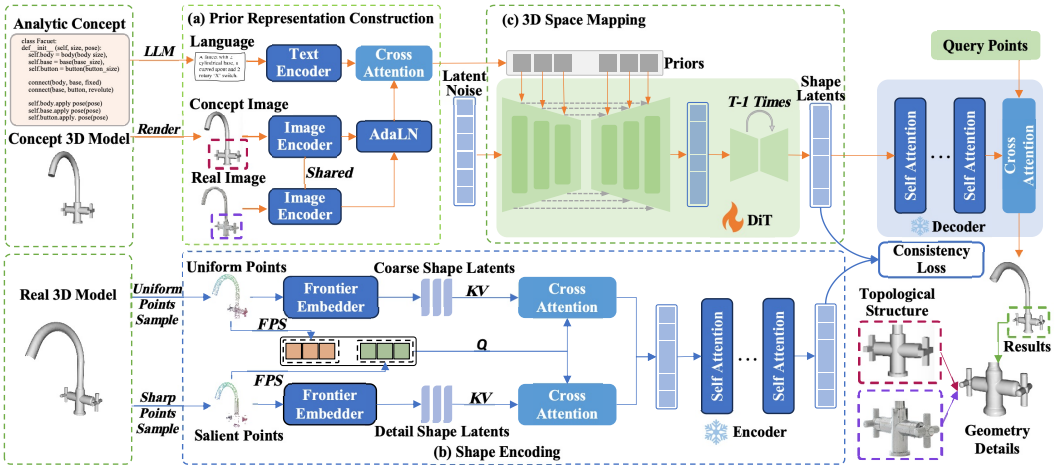


Figure 3: Overall architecture of Prism. Blue arrows indicate processes performed only during training. (a) In Prior Representation Construction, we learn a novel prior representation that integrates structural concepts and shape details through AdaLN modulation and cross-attention. (b) In Shape Encoding, we sample uniform points and salient points from real objects and integrate them through cross-attention. We then use a pre-trained VAE encoder to obtain shape latents. (c) In 3D Space Mapping, we feed the prior representation into a DiT as the condition and learn a mapping to the latent space under the supervision of the shape latents produced in (b). Finally, we use a pre-trained VAE decoder to generate articulated objects with controllable topological structure and high-quality appearance from the latent features produced by the DiT.

failed predicates may change:

$$x_{\mathcal{S}_k}^{(k+1)} = A\Big(b, T, D_T, \mathcal{Q}_T, x^{(k)}, V_T(x^{(k)})\Big), \quad \mathcal{S}_k = \bigcup_{r \in V_T(x^{(k)})} \text{scope}(r), \tag{1}$$

where A denotes the constrained proposal-and-repair operator; parameters outside $\mathcal{S}_k$ remain fixed and discrete alternatives are enumerated when necessary. A candidate is accepted only when all hard residuals, including explicit brief requirements, vanish:

$$C^* = T(x^{(k)}) \quad \text{if} \quad V_T(x^{(k)}) = \varnothing; \qquad C^* = \perp \quad \text{after } K \text{ unsuccessful audits.} \tag{2}$$

The accepted program is rendered into C_i and verbalized into C_t, preserving Prism's original conditioning interface. This design targets repeated instance construction within the coverage of an authored template library. New primitives, unsupported part assemblies, and genuinely new category templates still require human design.

3.4 Prism Framework

3.4.1 Prior Representation Construction

Prior Representation Construction aims to encode two types of information: the precise articulated structure specified by Analytic Concept and the appearance details provided by a rendered real-world image. Let the validated Analytic Concept be denoted as C. In the reported generator experiments, C is supplied by ConceptFactory Sun et al. (2024b); the construction stage can instead instantiate C^* from an available template, subject to the checks in Section 3.3. Based on C, we construct a corresponding conceptual model C_m and render it from selected camera views to obtain conceptual images C_i. We also convert the procedural description into a natural-language description C_t using a Large Language Model. The concept representations C, C_m, C_i, and C_t describe the same underlying structure from different modalities; more details are provided in Appendix A.8.

To provide appearance and shape-detail cues, we use a rendered image R_i, generated by VLM, and encode it with a pre-trained image encoder γ_i:

$$F_r = \gamma_i(R_i). \tag{3}$$

We then encode the conceptual image C_i with the same image encoder, while using AdaLN (adaptive layer normalization) modulation introduced in Li et al. (Li et al., 2024a) to inject the real-image feature into the concept-image encoding process:

$$F_v = \gamma_i(C_i, ModLN(F_r)), \tag{4}$$

where $ModLN$ denotes the modulation operation. This produces a visual representation F_v that preserves the concept-specified layout while incorporating appearance cues from R_i.

For the language modality, we use a pre-trained text encoder γ_t to encode the concept-derived description:

$$F_t = \gamma_t(C_t), \tag{5}$$

where C_t describes the structural and affordance information contained in the Analytic Concept. Finally, we fuse the visual representation F_v and language representation F_t through cross attention:

$$F_m = \text{CrossAttn}(F_t, F_v, F_v). \tag{6}$$

The resulting representation F_m serves as the condition for generation, carrying both analytic structural control and appearance information.

3.4.2 Shape Encoding

Shape Encoding constructs shape latents from real objects for supervision. To represent both global geometry and fine-grained details, we sample two point-cloud sets from each real object. Uniform point sampling is used to capture the coarse shape, while sharp-point sampling focuses on salient geometric details. A frontier embedder maps these two point clouds into coarse shape latents and detail shape latents, respectively. We then use cross attention to integrate them and self attention to obtain the final shape latents. The detailed process is described in Appendix A.6.

3.4.3 3D Space Mapping

Given the fused prior representation F_m, 3D Space Mapping learns to generate a latent that can be decoded into a mesh. We employ a learnable DiT (Peebles & Xie, 2023) conditioned on F_m. The shape latents produced by Shape Encoding are used as supervision, so the DiT learns to map the Analytic-Concept-guided representation into the pre-trained latent space.

This mapping should be understood as geometric realization of the analytic specification. The Analytic Concept provides structural information, including topology and articulation-related constraints; the DiT learns to produce a latent representation whose decoded geometry matches real shape distributions while preserving the specified structure through conditioning. Finally, the pre-trained VAE decoder converts the predicted latent into a mesh.

3.5 Training and Inference Process

Training Process. During training, the pre-trained VAE encoder and decoder are frozen. The trainable components are the conditional DiT and the fusion modules used in Prior Representation Construction, including the AdaLN modulation and cross-attention layers. We use the shape latent G obtained from Shape Encoding as supervision, and train the DiT with the fused representation F_m as the condition. The conditional DiT is optimized with the following denoising objective:

$$\mathcal{L}_{\text{DiT}} := \mathbb{E}_{G,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta\left(G_t, t, F_m\right)\|_2^2\right], \tag{7}$$

where ϵ_θ represents the DiT, which is built on a UNet-like transformer, t is sampled from $\{1, \ldots, T\}$, and G_t is the noisy version of G_0. We also adopt classifier-free guidance (CFG) (Ho & Salimans, 2022) by randomly dropping conditions during training.

Inference Process. During inference, we provide a validated Analytic Concept and a rendered image as inputs. The concept may be supplied by an existing annotation or instantiated by the automated construction module; in the latter case, generation proceeds only after the execution audit accepts it. Prism first constructs the fused representation F_m, then samples a latent using the trained conditional DiT. The pre-trained VAE decoder decodes this latent into an occupancy field and converts it into a mesh. Since the structural information is supplied through Analytic Concept and encoded in F_m, the generated asset preserves the specified articulated organization while using the learned latent space to produce realistic geometry.

Table 1: Quantitative comparison with baseline methods on Partnet-Mobility dataset, where Co represents concept priors and Se represents semantic priors.

Method	Co	Se	FID↓	CD×100↓	Uni3D-R↑	Uni3D-C↑
InstantMesh	✗	✓	23.72	3.16	0.20	0.12
CraftsMan	✗	✓	17.61	2.27	0.25	0.18
Trellis	✗	✓	13.35	1.83	0.27	0.21
Prism	✓	✓	**9.15**	**0.92**	**0.34**	**0.30**

Table 2: Success rates on average of manipulation tasks. Entries for Where2Act and Where2Explore is reported separately.

Method	Where2 Act	Where2 Explore	Manip LLM
Original	26.4 / 8.4	29.2 / 14.3	**33.4 / 19.0**
Prism w/o C_i	21.2 / 6.4	23.7 / 9.5	28.7 / 14.3
Prism	**28.2 / 9.2**	**33.0 / 15.6**	32.0 / 18.5

Table 3: Quantitative results on PartNet-Mobility dataset for motion parameters and geometry performance.

Metric	Type Err (↓)	Ang Err (↓)	Pos Err (↓)	FID (↓)	CD (↓)
Singapo	0.08	0.010	0.12	26.32	4.23
Freeart3D	0.04	0.005	0.05	23.17	2.71
Prism	**0.00**	**0.001**	**0.01**	**7.24**	**0.78**

4 Experiments

4.1 Implementation Details

We evaluate Prism on PartNet-Mobility (Xiang et al., 2020) with ConceptFactory-derived Analytic Concepts. After excluding low-quality meshes with thin structures, holes, or missing textures, the data contain approximately 3,000 training and 1,000 test objects across 16 categories, following a 3:1 split. These exclusions bound the range of geometry addressed by the reported results; category counts and preprocessing are detailed in Appendix A.7.1. We adopt Dora-VAE (Chen et al., 2025b) as the pre-trained model which contains 1.3 billion parameters. We freeze the VAE model during training. Our DiT has comprising 1.04 billion parameters and conditions on image features extracted by DINOv2 (Oquab et al., 2023) and text features extracted by CLIP(ViT-L-14) (Radford et al., 2021). Our training is conducted on 4 A100 GPUs for 500 epochs with a batch size of 32 and a learning rate of 5×10^{-5}.We set CFG with a drop rate of 0.1 and set CFG strength to 3 and sampling steps to 50 at the inference stage. We employ Flash-Attention-v2, mixed-precision training with FP16 and gradient checkpointing to optimize memory usage and training efficiency. For manipulation experiments, we adopt SAPIEN (Xiang et al., 2020) as simulation environment, using a Franka Panda gripper as the robot actuator. All quantitative generator results below use the existing ConceptFactory-derived concepts and measure conditional generation rather than automated concept construction. More implementation details can be found in Appendix A.7.1.

Condition alignment and evaluation. For each object, the concept program is instantiated to produce the part layout and a conceptual mesh. We render C_i from a selected camera view and render the corresponding real object with the same camera parameters before obtaining its appearance image R_i. The procedural program is converted to C_t with GPT-4o (Hurst et al., 2024); the appearance transformation uses DALL-E 3 (Betker et al.). This shared view is needed so that differences between C_i and R_i mainly concern appearance and shape detail rather than a camera mismatch. The shape encoder samples $32{,}768$ uniform points and the same number of sharp points per mesh, using a 10° dihedral threshold in the main configuration. FID measures distributional image quality, CD measures surface discrepancy, and Uni3D-C compares generated assets with concept images. Uni3D-C is a representation-based proxy for structural consistency, so we complement it with part and joint inspection and manipulation tests. For articulated comparisons, axis angle error is reported in degrees, revolute-axis position error in meters, and type error records an incorrect joint class. The same image resolution and train/test partition are used for the baselines; detailed preprocessing and simulation settings appear in Appendix A.7.1.

Comparison protocol. The general 3D and articulated baselines are fine-tuned on the same training split and evaluated on the same test objects with their prescribed input modalities. Prism additionally

receives an Analytic Concept, so the cross-method comparison measures performance at each method's intended interface rather than attributing every difference to the generator architecture. The concept-conditioned ablations retain Prism's backbone, data, and evaluation protocol while changing the available concept and appearance channels. Together, the comparisons assess the value of the full structural-control pipeline and the extent to which its conditioning design converts a fixed structural specification into accurate geometry.

4.2 Automated Construction Evaluation

The automated module is assessed separately from the generator benchmark, which uses existing concepts. The proposed protocol distinguishes interpolation, held-out parameter regions, unseen combinations of observed part attributes, and held-out categories with compatible templates; unsupported categories measure library coverage. Human configuration, independent parameter sampling, and a single-pass language-model proposal receive the same brief, template code, seed examples, and time budget. ConceptFactory's existing optimizer fits selected parts to an observed mesh and therefore addresses a different input setting Sun et al. (2024b). An independent checker would assess executable-instance rate, geometric and functional validity, coverage, diversity, and abstention alongside human and compute cost. Accepted concepts would enter the same frozen Prism generator for downstream evaluation. Appendix A.2 specifies the controlled ablations and accounting for template and seed preparation; construction-specific numerical results are not included in the current benchmark.

4.3 Main Results

We assess appearance with FID and Chamfer Distance (CD), structural consistency with Uni3D (Zhou et al., 2023) similarity to concept images, and affordance alignment through part pose, grasp area, grasp pose, and downstream manipulation. The evaluation focuses on human-made articulated objects from indoor scenes.

4.3.1 Qualitative Comparison

We compare with InstantMesh (Xu et al., 2024a), CraftsMan (Li et al., 2025), Trellis (Xiang et al., 2025), Singapo Liu et al. (2024a), and Freeart3D Chen et al. (2025a) on PartNet-Mobility (Xiang et al., 2020). Image-conditioned baselines receive 256×256 images. In the evaluated examples, Prism more consistently preserves repeated parts, connections, and functional layouts while retaining geometric detail; full comparisons appear in Appendix Figures 4 and 5.

4.3.2 Quantitative Comparison

Tables 1 and 3 report appearance similarity (Uni3D-R), concept consistency (Uni3D-C), joint type, angle and position errors, FID, and CD. Relative to the strongest general 3D baseline in Table 1, Prism reduces FID from 13.35 to 9.15 and CD from 1.83 to 0.92, while increasing Uni3D-C from 0.21 to 0.30. Relative to the strongest articulated baseline in Table 3, it reduces FID from 23.17 to 7.24 and CD from 2.71 to 0.78. These results show the benefit of an explicit structural interface for the complete generation pipeline. Joint errors have a narrower interpretation: because the concept supplies joint structure, Prism's low errors measure fidelity to specified articulation, whereas image-based baselines must also estimate that structure from their inputs.

4.3.3 Results on Manipulation

In SAPIEN (Xiang et al., 2020), we train Where2Act Mo et al. (2021), Where2Explore Ning et al. (2023), and ManipLLM Li et al. (2024b) with original or Prism-generated assets. Prism improves both reported success measures for Where2Act ($26.4/8.4$ to $28.2/9.2$) and Where2Explore ($29.2/14.3$ to $33.0/15.6$); ManipLLM remains slightly stronger with original assets. Removing C_i degrades every policy, linking structural conditioning to downstream utility rather than appearance alone. Pose and grasp alignment and additional simulations appear in Appendix Figures 6 and 7a.

In qualitative SAPIEN trials, each target is placed at the pose specified by its Analytic Concept, and a two-finger Franka Panda acts on the annotated grasp region. Additional scenes include mugs

Table 4: Generator ablations on PartNet-Mobility. Lower FID and CD are better; higher Uni3D-C indicates greater similarity to the Analytic Concept. A dash denotes an unreported metric. The construction-module ablations are specified separately and have no measured scores in this table.

Condition	FID ↓	CD×100 ↓	Uni3D-C ↑
C_i only	18.72	1.74	0.22
$C_i + R_i$	10.96	0.97	0.27
$C_i + C_t$	23.38	3.29	–
C_t only	14.86	1.54	0.24
Full $C_i + R_i + C_t$	**9.15**	**0.92**	**0.30**
Sharp points, 1°	12.70	1.09	0.29
Sharp points, 30°	11.64	1.04	0.28
Sharp points, 10°	**9.15**	**0.92**	**0.30**

and bottles grasped at different parts, a knife held by its handle, and two joint mechanisms: a microwave door with a revolute joint and a storage drawer with a prismatic joint. These interactions examine whether the generated mesh preserves the concept's contact and motion constraints; the visual evidence appears in Appendix Figures 12 and 14. Because the grasp poses and joint trajectories are prescribed by the concept rather than estimated from the generated mesh, these cases specifically probe whether generated surfaces and moving parts remain compatible with the supplied specification. They provide qualitative evidence and are not additional success-rate measurements.

4.4 Ablation Study

Prior representation. We ablate conceptual images C_i, concept text C_t, and appearance images R_i under the same generator and training data. Table 4 shows that the full condition achieves the lowest FID and CD and the highest concept similarity. Removing text while retaining concept images and appearance images increases FID from 9.15 to 10.96 and reduces Uni3D-C from 0.30 to 0.27. Concept images alone yield FID 18.72, indicating that an explicit structural cue requires complementary appearance information to produce high-quality geometry. Conversely, the full model improves on $C_i + R_i$ while retaining the same concept images, consistent with the fusion pathway extracting useful information from the concept's language description. The $C_i + C_t$ condition has FID 23.38 and CD 3.29; its Uni3D-C was unavailable in the original evaluation and is therefore left unreported. Additional visual comparisons appear in Appendix A.7.4.

Construction mechanism. The proposed ablations isolate dependency induction, executable feedback, and scoped repair under matched budgets. These components respectively address cross-part compatibility, invalid geometry or motion, and regressions during revision. Validity must be interpreted together with coverage because frequent abstention may otherwise inflate acceptance quality. Their quantitative effects remain unmeasured here (Appendix A.2).

Sharp point sampling. The shape encoder samples high-curvature regions using a dihedral-angle threshold. In Table 4, 10° outperforms 1° and 30° on all three reported metrics. The lower threshold admits too few salient edges, whereas the higher threshold weakens the emphasis on fine interfaces such as joints and knobs. The complete threshold study, including Uni3D-R and visual results, appears in Appendix A.7.4.

5 Conclusion

Prism uses executable Analytic Concepts and learned multi-modal fusion to generate articulated assets with controlled structure and detailed geometry. Its upstream module constructs valid instances of available templates through dependency-aware proposals and execution checks. Experiments with existing concepts show strong geometric and structural performance; condition ablations support the integration of structural and appearance cues. Construction coverage, cost, and fragile topology remain open questions.

ETHICS STATEMENT

There are no ethical problems.

REFERENCES

James Betker, Gabriel Goh, Li Jing, † TimBrooks, Jianfeng Wang, Linjie Li, † LongOuyang, † JuntangZhuang, † JoyceLee, † YufeiGuo, † WesamManassra, † PrafullaDhariwal, † CaseyChu, † YunxinJiao, and Aditya Ramesh. Improving image generation with better captions.

Irving Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987.

Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16123–16133, 2022.

Chuhao Chen, Isabella Liu, Xinyue Wei, Hao Su, and Minghua Liu. Freeart3d: Training-free articulated object generation using 3d diffusion. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–13, 2025a.

Rui Chen, Jianfeng Zhang, Yixun Liang, Guan Luo, Weiyu Li, Jiarui Liu, Xiu Li, Xiaoxiao Long, Jiashi Feng, and Ping Tan. Dora: Sampling and benchmarking for 3d shape variational auto-encoders. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 16251–16261, 2025b.

Zhaoxi Chen, Jiaxiang Tang, Yuhao Dong, Ziang Cao, Fangzhou Hong, Yushi Lan, Tengfei Wang, Haozhe Xie, Tong Wu, Shunsuke Saito, et al. 3dtopia-xl: Scaling high-quality 3d asset generation via primitive diffusion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 26576–26586, 2025c.

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

Kevin Ellis, Maxwell Nye, Yewen Pu, Felix Sosa, Josh Tenenbaum, and Armando Solar-Lezama. Write, execute, assess: Program synthesis with a repl. *Advances in Neural Information Processing Systems*, 32, 2019.

Yaroslav Ganin, Sergey Bartunov, Yujia Li, Ethan Keller, and Stefano Saliceti. Computer-aided design as language. *Advances in Neural Information Processing Systems*, 34:5885–5897, 2021.

Haoran Geng, Helin Xu, Chengyang Zhao, Chao Xu, Li Yi, Siyuan Huang, and He Wang. Gapartnet: Cross-category domain-generalizable object perception and manipulation via generalizable and actionable parts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7081–7091, 2023.

Yang Hai, Rui Song, Jiaojiao Li, Mathieu Salzmann, and Yinlin Hu. Rigidity-aware detection for 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8927–8936, 2023.

Nick Heppert, Muhammad Zubair Irshad, Sergey Zakharov, Katherine Liu, Rares Andrei Ambrus, Jeannette Bohg, Abhinav Valada, and Thomas Kollar. Carto: Category and joint agnostic reconstruction of articulated objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21201–21210, 2023.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. LRM: Large reconstruction model for single image to 3d. In *The Twelfth International Conference on Learning Representations*, 2024.

Ishtiaque Hossain, I-Chao Shen, and Oliver van Kaick. Approximating procedural models of 3d shapes with neural networks. In *Computer Graphics Forum*, pp. e70024. Wiley Online Library, 2025.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

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

Mohammad Sadil Khan, Sankalp Sinha, Talha Uddin, Didier Stricker, Sk Aziz Ali, and Muhammad Zeshan Afzal. Text2cad: Generating sequential cad designs from beginner-to-expert level text prompts. *Advances in Neural Information Processing Systems*, 37:7552–7579, 2024.

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

Jiahui Lei, Congyue Deng, Bokui Shen, Leonidas Guibas, and Kostas Daniilidis. Nap: Neural 3d articulation prior. *arXiv preprint arXiv:2305.16315*, 2023.

Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. In *ICLR*, 2024a.

Weiyu Li, Jiarui Liu, Hongyu Yan, Rui Chen, Yixun Liang, Xuelin Chen, Ping Tan, and Xiaoxiao Long. Craftsman3d: High-fidelity mesh generation with 3d native diffusion and interactive geometry refiner. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5307–5317, 2025.

Xiaoqi Li, Mingxu Zhang, Yiran Geng, Haoran Geng, Yuxing Long, Yan Shen, Renrui Zhang, Jiaming Liu, and Hao Dong. Manipllm: Embodied multimodal large language model for object-centric robotic manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18061–18070, 2024b.

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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 300–309, 2023.

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

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

Jiayi Liu, Hou In Ivan Tam, Ali Mahdavi-Amiri, and Manolis Savva. Cage: Controllable articulation generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17880–17889, 2024b.

Liu Liu, Wenqiang Xu, Haoyuan Fu, Sucheng Qian, Qiaojun Yu, Yang Han, and Cewu Lu. Akb-48: A real-world articulated object knowledge base. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14809–14818, 2022.

Liqian Ma, Jiaojiao Meng, Shuntao Liu, Weihang Chen, Jing Xu, and Rui Chen. Sim2real^2: Actively building explicit physics model for precise articulated object manipulation. *arXiv preprint arXiv:2302.10693*, 2023.

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. *Communications of the ACM*, 65(1):99–106, 2021.

Kaichun Mo, Leonidas J Guibas, Mustafa Mukadam, Abhinav Gupta, and Shubham Tulsiani. Where2act: From pixels to actions for articulated 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6813–6823, 2021.

Jiteng Mu, Weichao Qiu, Adam Kortylewski, Alan Yuille, Nuno Vasconcelos, and Xiaolong Wang. A-sdf: Learning disentangled signed distance functions for articulated shape representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13001–13011, 2021.

Chuanruo Ning, Ruihai Wu, Haoran Lu, Kaichun Mo, and Hao Dong. Where2explore: Few-shot affordance learning for unseen novel categories of articulated objects. *Advances in Neural Information Processing Systems*, 36:4585–4596, 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Chuer Pan, Brian Okorn, Harry Zhang, Ben Eisner, and David Held. Tax-pose: Task-specific cross-pose estimation for robot manipulation. In *Conference on Robot Learning*, pp. 1783–1792. PMLR, 2023.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023.

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

Pradyumna Reddy, Michael Gharbi, Michal Lukac, and Niloy J Mitra. Im2vec: Synthesizing vector graphics without vector supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7342–7351, 2021.

Aditya Sanghi, Hang Chu, Joseph G Lambourne, Ye Wang, Chin-Yi Cheng, Marco Fumero, and Kamal Rahimi Malekshan. Clip-forge: Towards zero-shot text-to-shape generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18603–18613, 2022.

A Seff, W Zhou, N Richardson, and RP Adams. Vitruvion: A generative model of parametric cad sketches. In *The International Conference on Learning Representations (ICLR)*, 2022.

Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. MVDream: Multi-view diffusion for 3d generation. In *The Twelfth International Conference on Learning Representations*, 2024.

Yawar Siddiqui, Tom Monnier, Filippos Kokkinos, Mahendra Kariya, Yanir Kleiman, Emilien Garreau, Oran Gafni, Natalia Neverova, Andrea Vedaldi, Roman Shapovalov, et al. Meta 3d assetgen: Text-to-mesh generation with high-quality geometry, texture, and pbr materials. *Advances in Neural Information Processing Systems*, 37:9532–9564, 2024.

Jianhua Sun, Yuxuan Li, Jiude Wei, Longfei Xu, Nange Wang, Yining Zhang, and Cewu Lu. Arti-pg: A toolbox for procedurally synthesizing large-scale and diverse articulated objects with rich annotations. *arXiv preprint arXiv:2412.14974*, 2024a.

Jianhua Sun, Yuxuan Li, Longfei Xu, Nange Wang, Jiude Wei, Yining Zhang, and Cewu Lu. Conceptfactory: Facilitate 3d object knowledge annotation with object conceptualization. *Advances in Neural Information Processing Systems*, 37:75454–75467, 2024b.

Jianhua Sun, Yuxuan Li, Longfei Xu, Jiude Wei, Liang Chai, and Cewu Lu. Discovering conceptual knowledge with analytic ontology templates for articulated objects. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 14681–14689, 2025.

Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. In *ICLR*, 2024.

Wei-Cheng Tseng, Hung-Ju Liao, Lin Yen-Chen, and Min Sun. Cla-nerf: Category-level articulated neural radiance field. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 8454–8460. IEEE, 2022.

Tengfei Wang, Bo Zhang, Ting Zhang, Shuyang Gu, Jianmin Bao, Tadas Baltrusaitis, Jingjing Shen, Dong Chen, Fang Wen, Qifeng Chen, et al. Rodin: A generative model for sculpting 3d digital avatars using diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4563–4573, 2023.

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

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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 21469–21480, 2025.

Desai Xie, Sai Bi, Zhixin Shu, Kai Zhang, Zexiang Xu, Yi Zhou, Sören Pirk, Arie Kaufman, Xin Sun, and Hao Tan. Lrm-zero: Training large reconstruction models with synthesized data. *Advances in Neural Information Processing Systems*, 37:53285–53316, 2024.

Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024a.

Xiang Xu, Karl DD Willis, Joseph G Lambourne, Chin-Yi Cheng, Pradeep Kumar Jayaraman, and Yasutaka Furukawa. Skexgen: Autoregressive generation of cad construction sequences with disentangled codebooks. In *International Conference on Machine Learning*, pp. 24698–24724. PMLR, 2022a.

Xiang Xu, Joseph Lambourne, Pradeep Jayaraman, Zhengqing Wang, Karl Willis, and Yasutaka Furukawa. Brepgen: A b-rep generative diffusion model with structured latent geometry. *ACM Transactions on Graphics (TOG)*, 43(4):1–14, 2024b.

Zhenjia Xu, Zhanpeng He, and Shuran Song. Universal manipulation policy network for articulated objects. *IEEE robotics and automation letters*, 7(2):2447–2454, 2022b.

Harry Zhang, Ben Eisner, and David Held. Flowbot++: Learning generalized articulated objects manipulation via articulation projection. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 1222–1241. PMLR, 06–09 Nov 2023.

Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)*, 43(4):1–20, 2024.

Chao Zhao, Chunli Jiang, Lifan Luo, Shuai Yuan, Qifeng Chen, and Hongyu Yu. Learning thin deformable object manipulation with a multi-sensory integrated soft hand. *IEEE Transactions on Robotics*, 2025.

Xin-Yang Zheng, Hao Pan, Peng-Shuai Wang, Xin Tong, Yang Liu, and Heung-Yeung Shum. Locally attentional sdf diffusion for controllable 3d shape generation. *ACM Transactions on Graphics (ToG)*, 42(4):1–13, 2023.

Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. *arXiv preprint arXiv:2310.06773*, 2023.

Yifeng Zhu, Zhenyu Jiang, Peter Stone, and Yuke Zhu. Learning generalizable manipulation policies with object-centric 3d representations. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 3418–3433. PMLR, 06–09 Nov 2023.

A APPENDIX

A.1 LLM USE STATEMENT

Generative AI tools were used to derive natural-language descriptions from procedural Analytic Concepts and to assist with the proposed automated Analytic Concept construction methodology. Generative AI tools also assisted with writing and editing this manuscript. The authors are responsible for verifying AI-assisted content and for the final text, claims, and artifacts.

A.2 AUTOMATED CONSTRUCTION EVALUATION PROTOCOL

The proposed evaluation distinguishes interpolation within observed parameter ranges, held-out parameter regions, unseen combinations of observed part attributes, and held-out categories with a compatible composite template. Programs sharing a template family remain in one partition to avoid near-duplicate leakage. Categories requiring a new template are recorded separately as a measure of library coverage and are outside the module's supported construction task.

Within a supplied template, human configuration from the brief, independent sampling within marginal parameter ranges, a single-pass language-model proposal, and the complete module receive the same target brief, template code, seed-example budget, and wall-clock limit. ConceptFactory's differentiable parameter optimizer fits selected parts to an observed mesh; without that mesh it addresses a different input setting and is documented as the source workflow rather than treated as an equivalent brief-conditioned baseline Sun et al. (2024b). An independent checker, unavailable to the repair process, would assess executable-instance rate and geometric, articulation, affordance, and brief-compatibility predicates. Coverage and diversity among accepted instances would be reported alongside abstention, audit iterations, compute time, and human correction time; template-authoring and seed-example effort would be recorded separately.

Controlled ablations would remove dependency induction, executable feedback, or scoped repair while retaining the same proposal model and budget. This separates cross-part compatibility, error detection, and repair stability; validity must be interpreted together with coverage because frequent abstention can inflate acceptance quality. Accepted concepts from every method would enter the same frozen Prism generator for topology consistency, CD, joint-error, and manipulation evaluation. No construction-specific numerical results are included in the present generator benchmark.

A.3 MAIN-PAPER QUALITATIVE PANELS

The following panels are placed in the appendix to satisfy the ICLR main-text page limit.

A.4 MORE RELATED WORK

A.4.1 MANIPULATION TASK

Object manipulation encompasses a range of tasks centered on enabling embodied agents to interact appropriately with objects. Pick-and-place is a fundamental task in manipulation Liu et al. (2023). Some methods concentrate on pose estimation of known objects Pan et al. (2023); Hai et al. (2023), while others choose to create universal policies for novel objects Zhu et al. (2023); Xu et al. (2022b). Besides this, some researchers focuses on improving the intelligence of robots to handle more complex tasks, such as manipulating deformable and articulated objects Zhang et al. (2023); Ma et al. (2023); Zhao et al. (2025). For instance, Where2Act (Mo et al., 2021) predicts per-pixel action likelihoods and manipulation proposals to guide interaction. Where2Explore (Ning et al., 2023) adopts few-shot learning for object manipulation, which measures affordance similarity across object categories to transfer affordance knowledge to novel objects. Additionally, GAPartNet (Geng et al., 2023) with semantic and affordance labels has been released, accompanied by a manipulation pipeline that leverages the concept of actionable parts. The effectiveness of manipulation using these methods is largely contingent on the depth of understanding regarding affordances in objects.

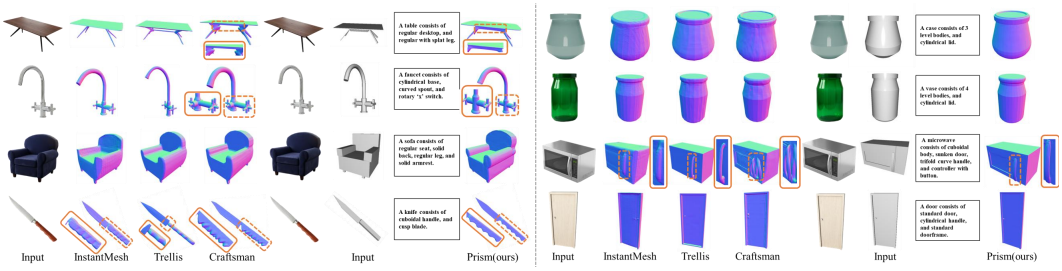


Figure 4: Qualitative comparisons with baseline methods on generation results.

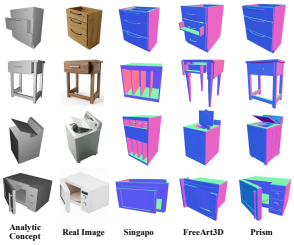


Figure 5: Qualitative comparisons with articulated generation baselines.

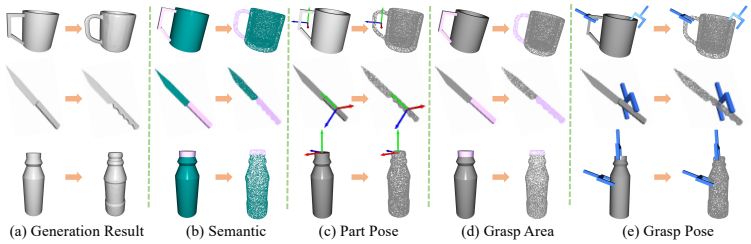


Figure 6: Visualization of generated results conditioned on Analytic Concept. Our methods can effectively control the topological structures along with part pose, grasp area, and grasp pose specified by Analytic Concept.

A.4.2 Neurosymbolic Synthesis

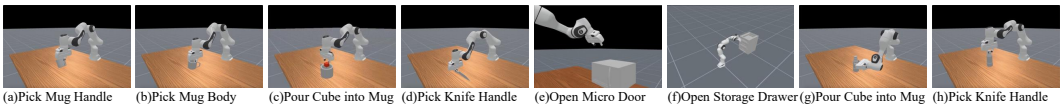
Neurosymbolic synthesis has long been applied to represent graphical content, providing interpretable parameters, stochastic variability, and high-quality outputs. However, designing procedural models from scratch remains challenging. Advances in AI have enabled neurosymbolic models to combine the strengths of AI and symbolic programs for representing, generating, and manipulating visual data. Transformer-based methods (Ganin et al., 2021; Seff et al., 2022; Khan et al., 2024) leverage large-scale sketch datasets for engineering sketch generation, with advances in controllability through image or sketch conditioning, codebook-based geometry/topology separation. Recent research (Ellis et al., 2019; Reddy et al., 2021; Xu et al., 2024b) tackles inferring 2D shape programs from images or sketches using reinforcement learning, differentiable rendering or diffusion model to capture structural regularities. Another line of work (Xu et al., 2022a; Hossain et al., 2025) employs neurosymbolic models to generate shapes by leveraging transformer-based architectures specialized for CAD operations. Although procedural models have achieved promising results across various domains, they remain limited to shape modeling and overlook the conceptual nature of the generated objects. Our work leverages structural concept for generation.

A.5 More Details on Analytic Concept

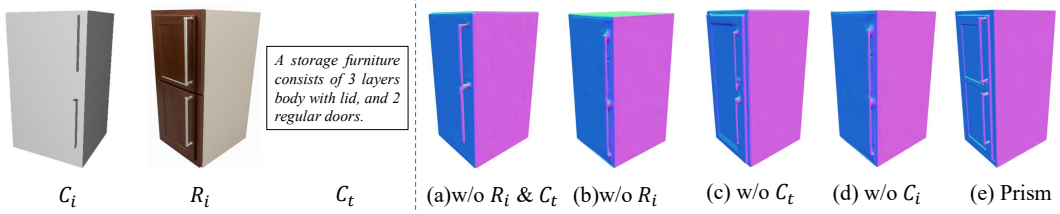
A.5.1 Definition

An Analytic Concept is an executable description of an object's part geometry and spatial relations together with associated functional knowledge. Figure 8 illustrates how a bucket's body and handle can be described through geometric, topological, affordance, and kinematic expressions (Sun et al., 2025). The program provides the structural specification consumed by Prism; its rendered mesh and natural-language description are two representations of the same instance.

The template library has two levels (Sun et al., 2024b;a). Elementary geometry templates parameterize reusable shapes such as cuboids and cylinders. Higher-level concept templates compose these shapes or other concept templates under spatial constraints and can define part semantics and affordance functions. A composite object template assembles the relevant parts and their relations. Its continuous parameters govern dimensions and poses, whereas discrete parameters can govern repeated components. Instantiating this template produces an object-level procedural description; rendering and textual conversion do not change the underlying part organization. The automated



(a) Visualization in the simulation environment.



(b) Visualization of the ablation study on different prior components.

Figure 7: Overall visualization results including simulation and ablation study.

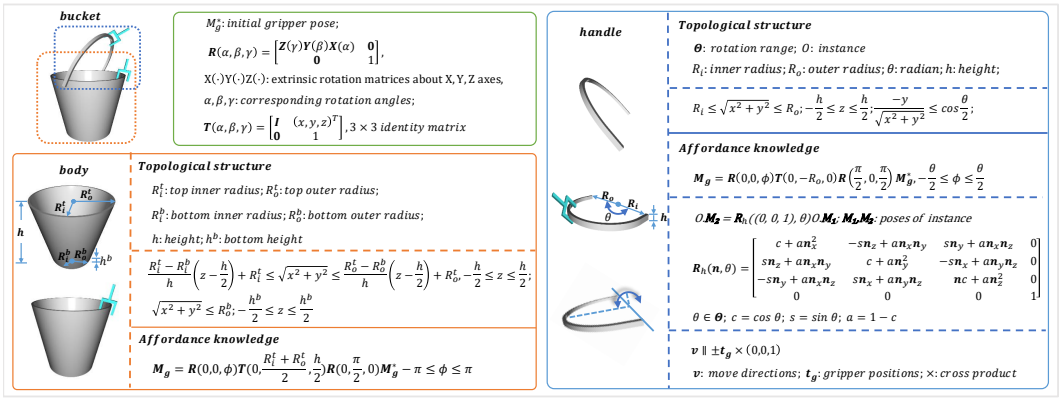


Figure 8: Procedural description of a bucket Analytic Concept. The body (left) specifies geometry, topology, and affordance regions; the handle (right) additionally specifies its kinematic relation.

module in Section 3.3 searches the parameters of an already compatible composite template. It does not create the elementary or higher-level templates, choose an unsupported part decomposition, or add new functional definitions.

A.5.2 Scalability

The two-level library permits reuse of geometric and functional patterns across categories Sun et al. (2024b;a). A handle concept, for example, may contribute to several object assemblies. Such reuse expands the range of instances that can be parameterized without redefining every component, but coverage of a genuinely new part arrangement or function still requires an author to extend the library or assemble a new template. Automated parameter search cannot remove this one-time design effort.

The cost figures in the cited literature refer to different activities. AKB-48 estimates more than 120 minutes and over $100 to commission one manually modeled articulated CAD object, versus about 20 minutes for its real-world scanning route Liu et al. (2022). ConceptFactory reports approximately 7 minutes of human conceptualization per object with its part-fitting optimizer, compared with approximately 10 minutes without that optimizer Sun et al. (2024b). Neither comparison measures Prism's automated construction cost, and neither includes an estimate of the effort to author every new composite template and seed set used here. We therefore treat amortized template reuse as a motivation and specify separate accounting for authoring, correction, and compute time in Appendix A.2.

A.5.3 Manual Construction and Automated Scope

Figure 9 shows the original ConceptFactory workflow for an observed object Sun et al. (2024b). The user chooses a library template for each part, fits shape and pose parameters against the target

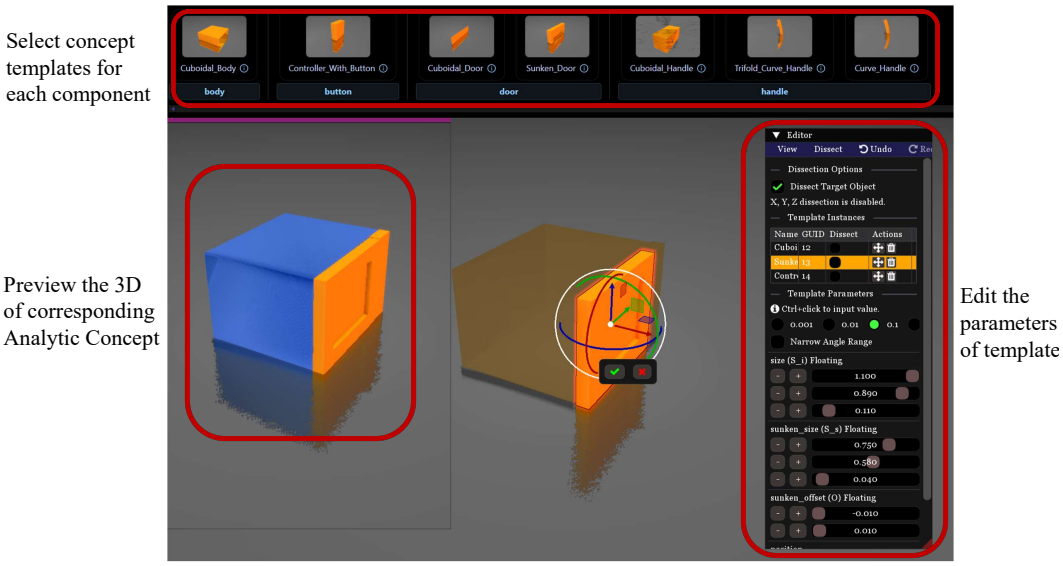


Figure 9: ConceptFactory's manual conceptualization interface Sun et al. (2024b). A user selects templates for observed parts and adjusts their dimensions and spatial transformations, with rendered previews and optional parameter optimization. Prism's automated module instead starts from a compatible composite template and searches for valid object-level instances.

geometry, and assembles the part instances into a procedural description. ConceptFactory already offers a differentiable optimizer for selected part parameters, followed by human correction where needed. Its reported time therefore describes an assisted manual workflow, not wholly manual tuning.

Prism's upstream module begins only after a compatible object-level assembly and valid seed configurations are available. From a target brief, it proposes dimensions, poses, counts, and articulation values across that assembly; it then checks whether the resulting program executes and satisfies explicit geometric and functional predicates. This differs from fitting individual selected parts to a known target mesh. Supplying the assembly and seeds may require human work, which must be counted when evaluating any claimed saving. If no compatible assembly exists, the module abstains until the library is extended.

A.6 More Details on Shape Encoding.

We encode real objects using a pretrained VAE to obtain latent codes. To better represent the appearance, we perform uniform points sampling strategy on objects to obtain a point cloud P_u that represents the coarse shape of the objects, and we further detect and sample the sharp edges of each object, so as to obtain a point cloud with sufficiently rich geometric details. We believe that the geometric details of objects can be represented by sampled sharp points from sharp edges. We achieve the sampling of sharp points through two steps:

Sharp Edge Detection. Given the mesh of an object, we determine whether an edge is a sharp edge based on the size of the dihedral angle between two adjacent faces, because the size of the dihedral angle can directly reflect the curvature of the mesh edge. A large dihedral angle indicates that the edge is a sharp edge.

For the common edge e of any two adjacent faces f_1 and f_2. The dihedral angle θ_e is computed as follows:

$$\theta_e = \arccos\left(\frac{n_{f_1} \cdot n_{f_2}}{\|n_{f_1}\|\|n_{f_2}\|}\right), \tag{8}$$

where n_{f1} and n_{f2} are the normal vectors of f_1 and f_2 respectively. The set of sharp edges $\mathcal{E}$ includes all edges with a dihedral angle exceeding the preset threshold τ:

$$\mathcal{E} = \{e \mid \theta_e > \tau\}. \tag{9}$$

The threshold is a hyperparameter set as 10 degrees, and sharp edges are all edges with a dihedral angle which exceeds this threshold.

Sharp Point Sampling. After obtaining the set of sharp edges, we aim to sample sharp points from the sharp edges to obtain the point cloud which represents the geometric details. Let P_s be the set of sharp points, which is initially an empty set. For each sharp edge, we retain its two vertices v_1 and v_2 and add them to the set of sharp vertices: $P_s = P_s \cup \{v_1, v_2\}$. After obtaining P_s, according to the preset sharp point threshold.

When the number of sharp vertices is sufficient ($N_d \leq N_V$), we use FPS (Farthest Point Sampling) to down-sample $P_{\mathcal{E}}$ to obtain P_s. When the number of sharp vertices is insufficient ($N_V < N_d$), we include all vertices in $P_{\mathcal{E}}$ and supplement them with interpolated points P_I. These additional points are generated by uniformly sampling $(N_d - N_V)/N_{\mathcal{E}}$ points on each sharp edge in $\mathcal{E}$ to ensure comprehensive coverage of the sharp features.

Therefore, we obtain uniformly sampled point clouds P_u to represent the coarse shape and the sharp sampled point clouds P_s to represent the geometric details. Then we employ Fourier Feature Embedding(FFE) to get the corresponding embeddings:

$$E_u = \text{FFE}(P_u), E_s = \text{FFE}(P_s). \tag{10}$$

We encode E_u and E_s through a dual cross-attention mechanism, enabling the encoder to focus on both coarse shape information and fine-grained geometric information. After obtaining dense point cloud representations, we adopt FPS to down-sample P_u and P_s:

$$P_f = \text{FPS}(P_u, N_u) \cup \text{FPS}(P_s, N_s), \tag{11}$$

where N_u and N_s are the numbers of point clouds down-sampled from P_u and P_s respectively, resulting in P_f. Then, we calculate cross-attention features for uniform points and sharp points respectively:

$$\begin{aligned} G_u &= \text{CrossAttn}(P_f, E_u, E_u), \\ G_s &= \text{CrossAttn}(P_f, E_s, E_s). \end{aligned} \tag{12}$$

Finally, the two attention features are concatenated to obtain the final point cloud features, which serve as the latent space representations of the real geometric information $G = G_u \oplus G_s$. This design can focus on both coarse shape and geometric details respectively, during the feature extraction process. And form a better representation of latent space.

A.7 More Experiment Results

A.7.1 More Implementation details

Data Processing. Due to significant noise in geometry and appearance, we exclude low-quality meshes from our training data including those with thin structures, holes, and texture-free surfaces, to guarantee high data quality. This process yielded a refined training dataset containing roughly 3k objects and a test dataset with around 1k objects including 16 categories: scissor, mug, chair, bucket, bottle, table, faucet, knife, refrigerator, display, microwave, trash can, door, vase, dishwasher, and storage furniture.

For mesh processing, we follow CLAY (Zhang et al., 2024) mesh to ensure watertight models. For sharp edge sampling, we set the number of sampled points $N_u = N_s = 32768$, and sampling angle threshold $\tau = 10$ degrees.

For each mesh, we construct an image for conceptual representations using the procedural language in Analytic Concept. For details, we select an appropriate view to render the concept meshes into images which can best represent its topological structure and fuse with real-world images to serve as visual priors. We adopt DALL-E3 (Betker et al.) to produce real-world images based on ground truth objects rendered by the same camera parameters which were used to render conceptual images, and transfer procedure language into nature language descriptions to form as language conditions of prior representations by GPT-4o (Hurst et al., 2024).

Evaluating Metrics. We conduct a quantitative analysis of the experimental results from two aspects. One is to measure the visual effect of the generative model, which is used to determine the distribution difference between generated results and ground truth models. The other is to measure

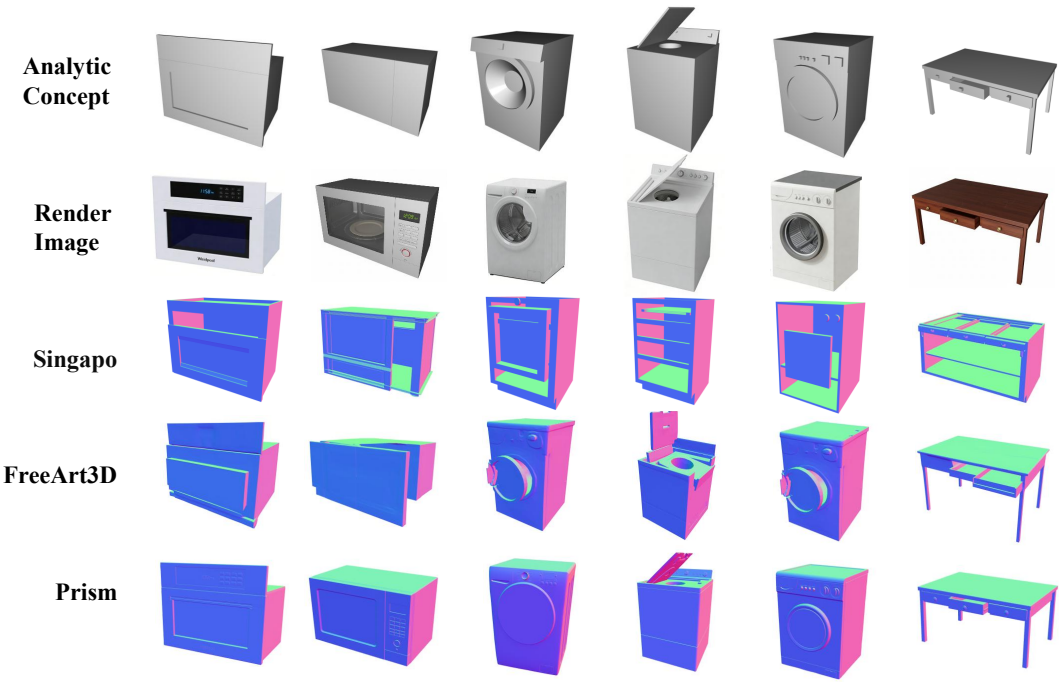


Figure 10: Qualitative comparisons with baseline methods on articulated objects.

the consistency of topological structure between the generated results and Analytic Concept, which is used to determine whether the generated results is consistent with the structural concept.

We use FID and CD to measure the appearance between ground truth shapes and generated shapes. We introduce Uni3D to calculate the feature similarity between real-world images and the generated results. This measurement can be used to measure the consistency of topological structures between the two modalities. A higher similarity indicates a higher consistency in topological structure.

To evaluate the motion joints, following previous methods Liu et al. (2023); Jiang et al. (2022); Le et al. (2024), we evaluate the accuracy of the estimated articulation models using the following metrics: (1) Axis Ang Err (°): The angular deviation between the predicted joint axis and the ground truth for both revolute and prismatic joints. (2) Axis Pos Err (m): The minimum distance between the predicted and ground-truth joint axes for revolute joints. (3) Type Err: The joint type error for predicted motions.

Simulation Experiment Setting. The simulator placement and grasp protocol are summarized in Section 4.3.3; the figures below show the corresponding concept-guided interactions.

A.7.2 More Qualitative comparison results

We present additional examples of articulated assets generated by Prism and compare with other baselines in Figure 15 to Figure 19.

As we can see, Prism exhibits the superior topological structures and geometric details on these complex categories such as storage furniture, refrigerator, and dishwasher, especially for the handle. By integrating structural concept into the priors, Prism provides better controllability than previous methods relying solely on semantic priors.

Then we compare our methods with Singapo and Freeart3D on articulated objects, which is shown in Figure 10 and Figure 11. Singapo tends to generate incorrect topological structures. In contrast, our method can well preserve the structural consistency of the reconstructed models.

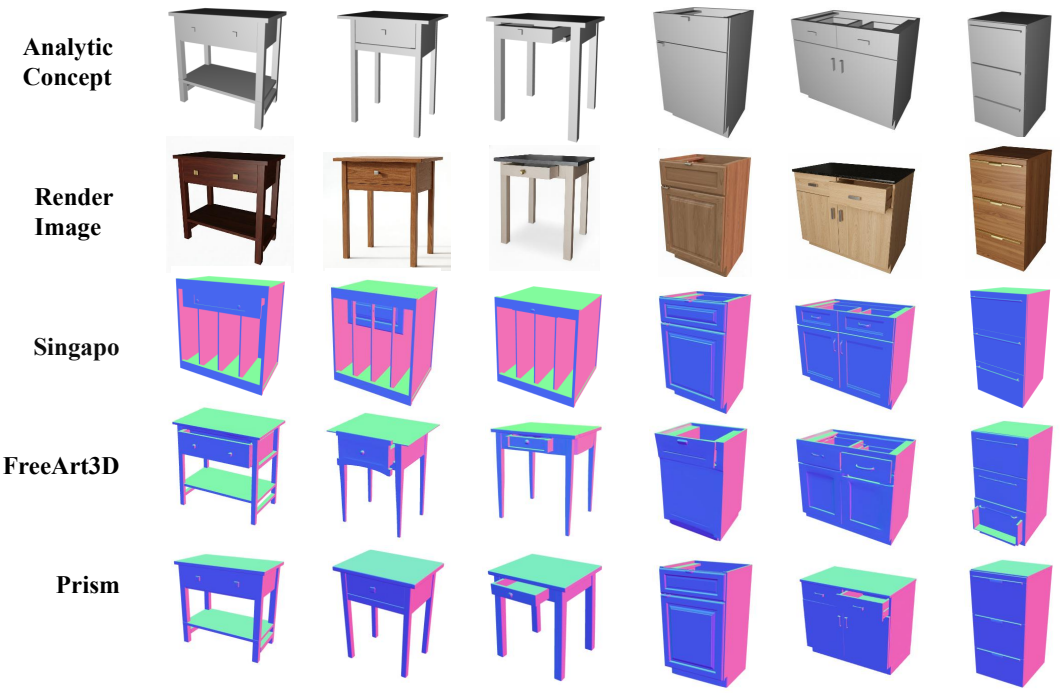


Figure 11: Qualitative comparisons with baseline methods on articulated objects.

Threshold	FID↓	CD×100↓	Uni3D-R↑	Uni3D-C↑
1 degree	12.70	1.09	0.34	0.29
30 degree	11.64	1.04	0.32	0.28
10 degree	**9.15**	**0.92**	**0.34**	**0.30**

(a) Ablation on different angle thresholds.

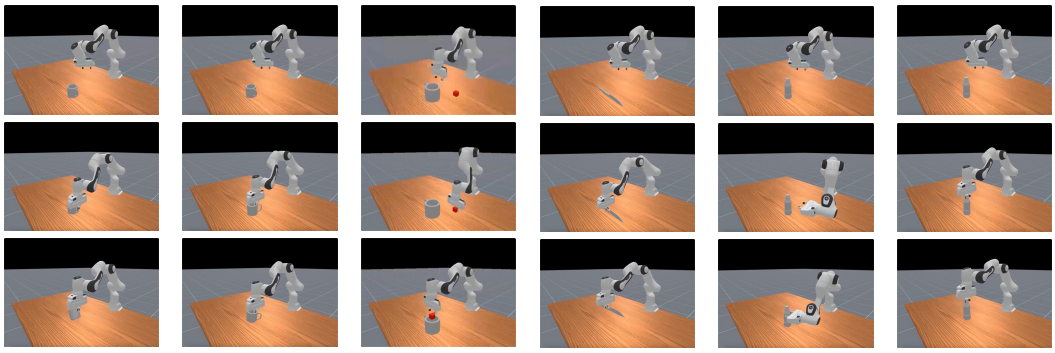
Prior Comparison			FID↓	CD×100↓	Uni3D-R↑	Uni3D-C↑
C_i	R_i	C_t				
✓	✗	✗	18.72	1.74	-	0.22
✓	✓	✗	10.96	0.97	0.31	0.27
✓	✗	✓	23.38	3.29	-	-
✗	✗	✓	14.86	1.54	-	0.24
✓	✓	✓	**9.15**	**0.92**	**0.34**	**0.30**

(b) Ablation on different components of priors.

Table 5: Quantitative results on ablation study. C_i represents conceptual images, R_i represents real-world images, and C_t represents language description of Analytic Concept.

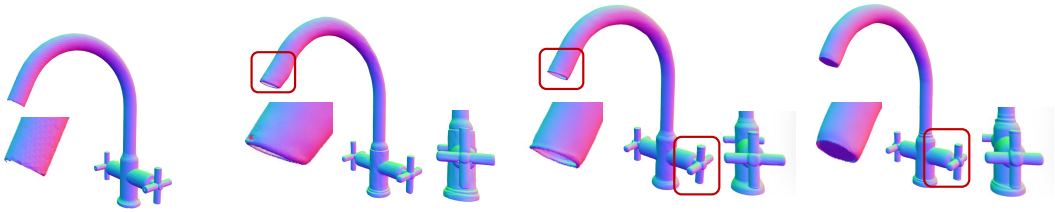
A.7.3 More Simulation Results on Manipulation.

For details, we use the Franka Panda gripper to pick up the objects according to the grasp pose described in Analytic Concept, including the handle and body of mugs, the handle of knives, the lid and body of bottles, the handle of the storage furniture and the door of refrigerator. As shown in Figure 12, we apply 6 tasks to illustrate that our generated meshes can align well with Analytic Concept in affordance knowledge. Figure 12(a) and Figure 12(b) show that the robot can successfully pick the generated mug, Figure 12(c) shows that the robot can pick cube (as a replacement for water) into the mug, which means the mug is aligned with Analytic Concept. Figure 12(d) shows that the



(a) Pick Mug Handle (b) Pick Mug Body (c) Pour Cube into Mug (d) Pick Knife Handle (e) Pick Bottle Body (f) Pick Bottle Lid

Figure 12: Results of simulation experiments in SAPIEN simulator.



(a) Ground Truth (b) Angle Threshold:1 (c) Angle Threshold:10 (d) Angle Threshold:30

Figure 13: Visualization results of ablation on different dihedral angle thresholds(/degree).

robot can successfully pick up the knife by its handle. And Figure 12(e) and Figure 12(f) exhibit the interaction to pick up bottle by handle and body respectively.

In addition, as illustrated in Figure 14, we conducted simulation experiments on two common joint types of articulated objects: rotation and translation. By leveraging the descriptions of the Analytic Concept, we can align the generated articulated objects with the corresponding concepts, thereby enabling the robot to perform manipulation in the simulation environment. Specifically, we can make the robot open the door of microwave (rotation joint) and open the drawer storage furniture (translation joint).

The results indicate that Prism can generate 3D assets which are well aligned with Analytic Concept in terms of affordance knowledge. The corresponding videos can be found in supplementary materials.

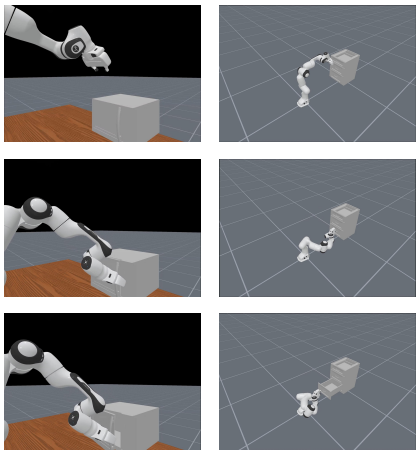
A.7.4 More Ablation Study

Ablation on Shape Encoding. In this section, we conduct an ablation study on sharp point sampling under different dihedral angle thresholds.

The quantitative results are shown in Table 5a, when the angle threshold is set to 10, the generative model achieves the best performance on all metrics. And when set it to 1, the FID is highest because sharp sample strategy cannot capture geometric details well. And when set it to 30, the FID increases to 11.64, CD decreases to 0.0104, Uni3D-R decreases to 0.32 and Uni3D-C decreases to 0.28, which indicates that the ability to control in geometric details and topological structures become lower.

The visualization results are shown in Figure 13. If the angle threshold is too small (set to 1 degree), the sharp points sampling is difficult to capture the model's sharp edges as the geometric detail for the Shape Encoding module. Conversely, if the angle threshold is too large(set to 30 degree), the sharp point sampling strategy cannot capture enough geometric details, resulting in insufficient accuracy of the geometric details generated at the joints and knobs.

Quantitative Results of Ablation on Prior Representation Construction. The quantitative results are shown in Table 5b We evaluate Uni3D with real-world images and generated results, so the



(a)Open the door of microwave (b)Open storage furniture

Figure 14: Results of simulation experiments in SAPIEN simulator.

metric is not available when w/o real-world images. As we can see, when missing real-world images, the FID is higher because the model cannot well control the shape details. When w/o language descriptions, the Uni3D decreases from 0.32 to 0.30, which indicates that the ability to control topological structures of such scenario is lower than our original prior representations.

A.8 Illustration of Data Preparation

Prism is evaluated on PartNet-Mobility Xiang et al. (2020) dataset to verify the effectiveness. PartNet-Mobility is a large-scale benchmark of articulated objects. It provides high-quality CAD models with detailed part segmentation and articulation annotations. The dataset includes various object categories. Each model contains joint types, motion parameters, and hierarchical part structures. This makes it suitable for research on articulated objects.

Besides, ConceptFactory Sun et al. (2024b) provides extensive procedural descriptions on the PartNet-Mobility dataset, which can serve as the Analytic Concepts for articulated objects. In Prism, we use the procedure provided by ConceptFactory as the description of Analytic Concept and then convert it into the corresponding visual and textual representations. Specifically, GPT-4o is used to generate textual descriptions from the code, and images are rendered from an appropriate camera viewpoint aligned with the concept. For real-world images, the Analytic Concept provides object position and orientation aligned with the PartNet-Mobility models, allowing us to render a viewpoint-consistent image of the real objects, which is then transformed by DALL-E 3 to reflect realistic textures and colors. This process yields concept images and textual descriptions aligned with the Analytic Concept, alongside corresponding rendered real-world images.

A.9 Discussion

A.9.1 Impact Statement

This paper presents work whose goal is to advance the field of articulated objects generation. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

A.9.2 Limitation

Despite its strong performance in generating high-quality articulated assets with precise structural control, Prism has ample room for future exploration. During the data preparation phase, meshes with thin structures and holes surfaces were explicitly excluded to guarantee high data quality. This exclusion suggests that the current generation pipeline may struggle to accurately reconstruct or generate objects featuring these highly specific, fragile topological characteristics. Automated Analytic

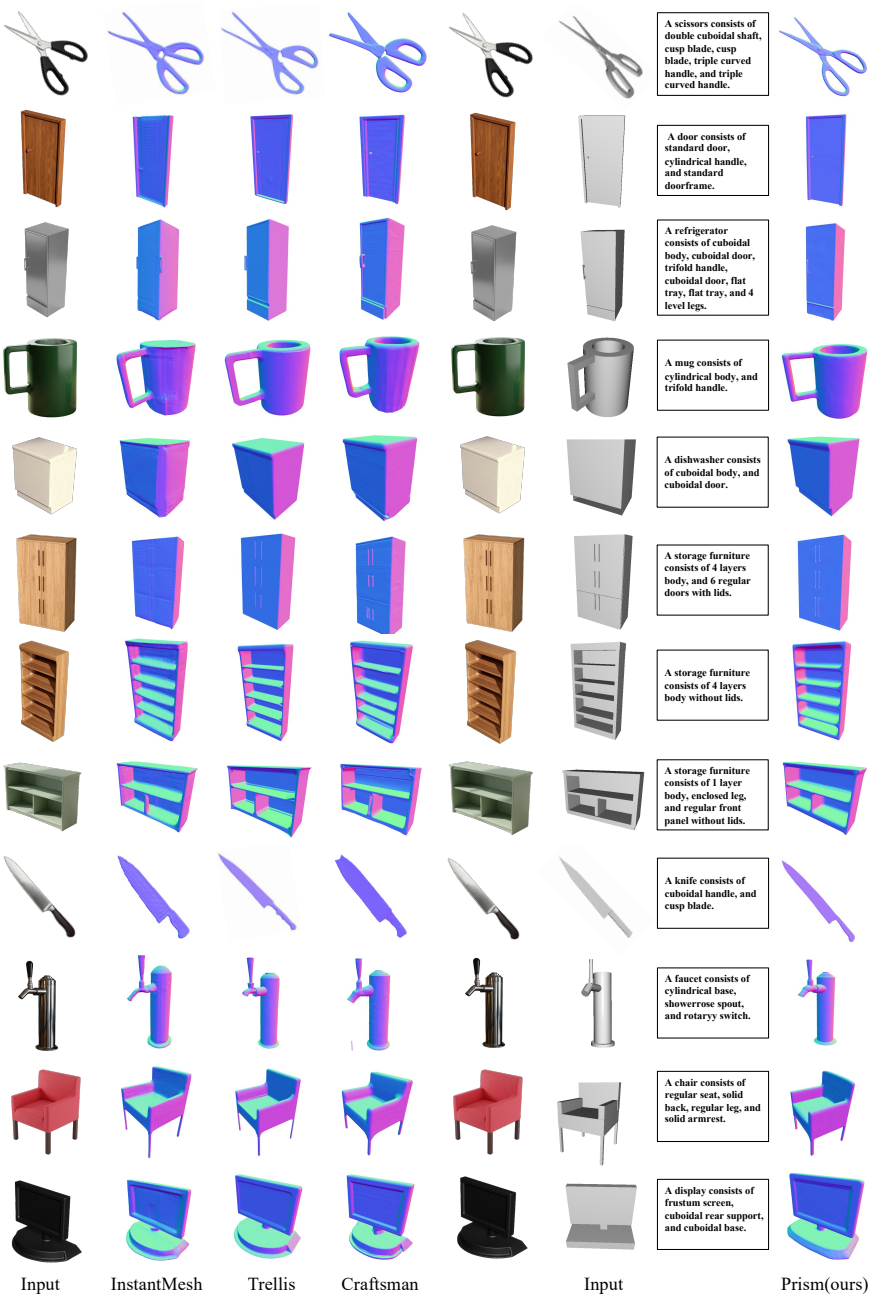


Figure 15: Qualitative comparisons with baseline methods on generation results.

Concept Construction is specified as an upstream extension, but the reported quantitative experiments use existing concepts and do not establish its success rate, construction cost, or generalization to held-out categories. Its execution audit can check explicit predicates but cannot guarantee physical usability or semantic correctness when the target brief is incomplete. New functional primitives and categories without a compatible template still require human implementation.

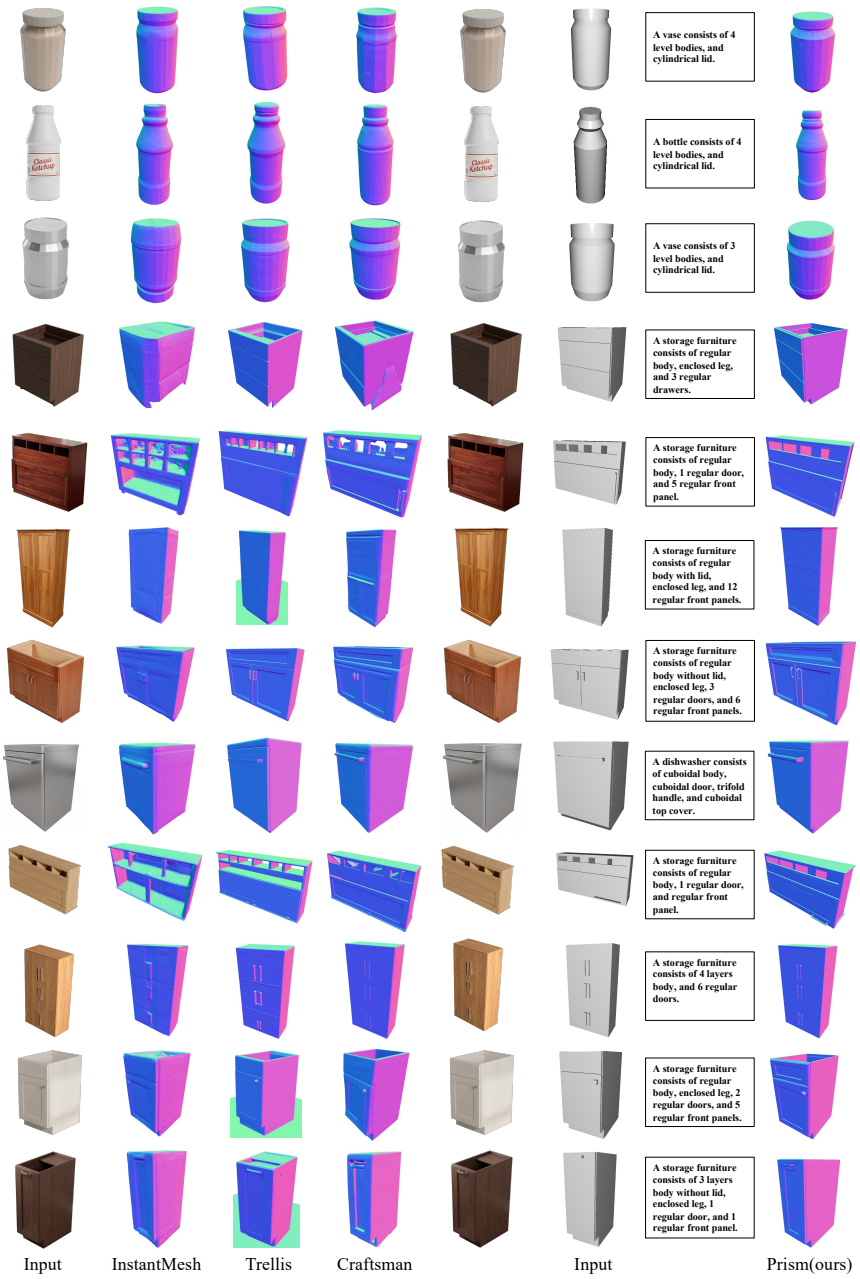


Figure 16: Qualitative comparisons with baseline methods on generation results.

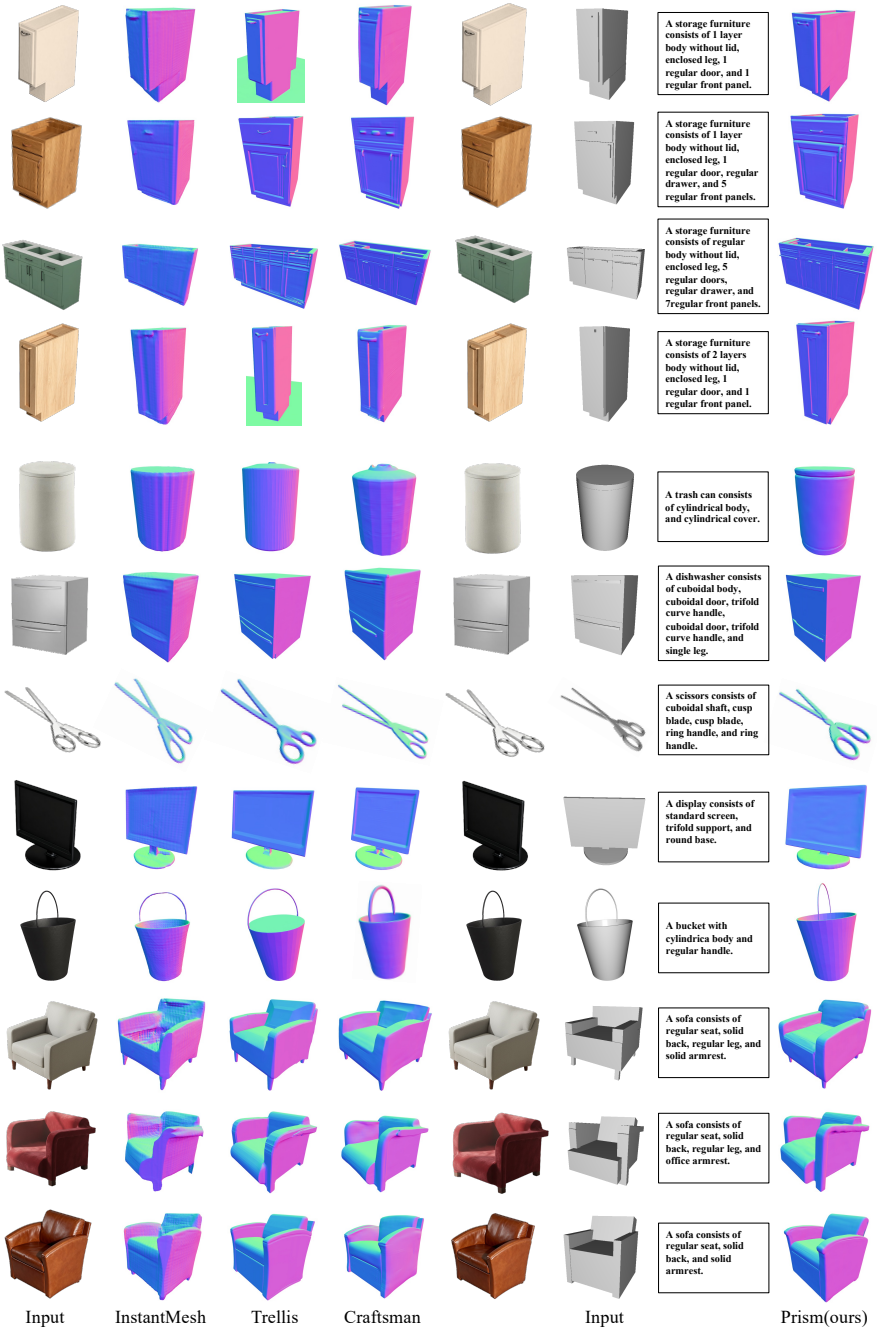


Figure 17: Qualitative comparisons with baseline methods on generation results.

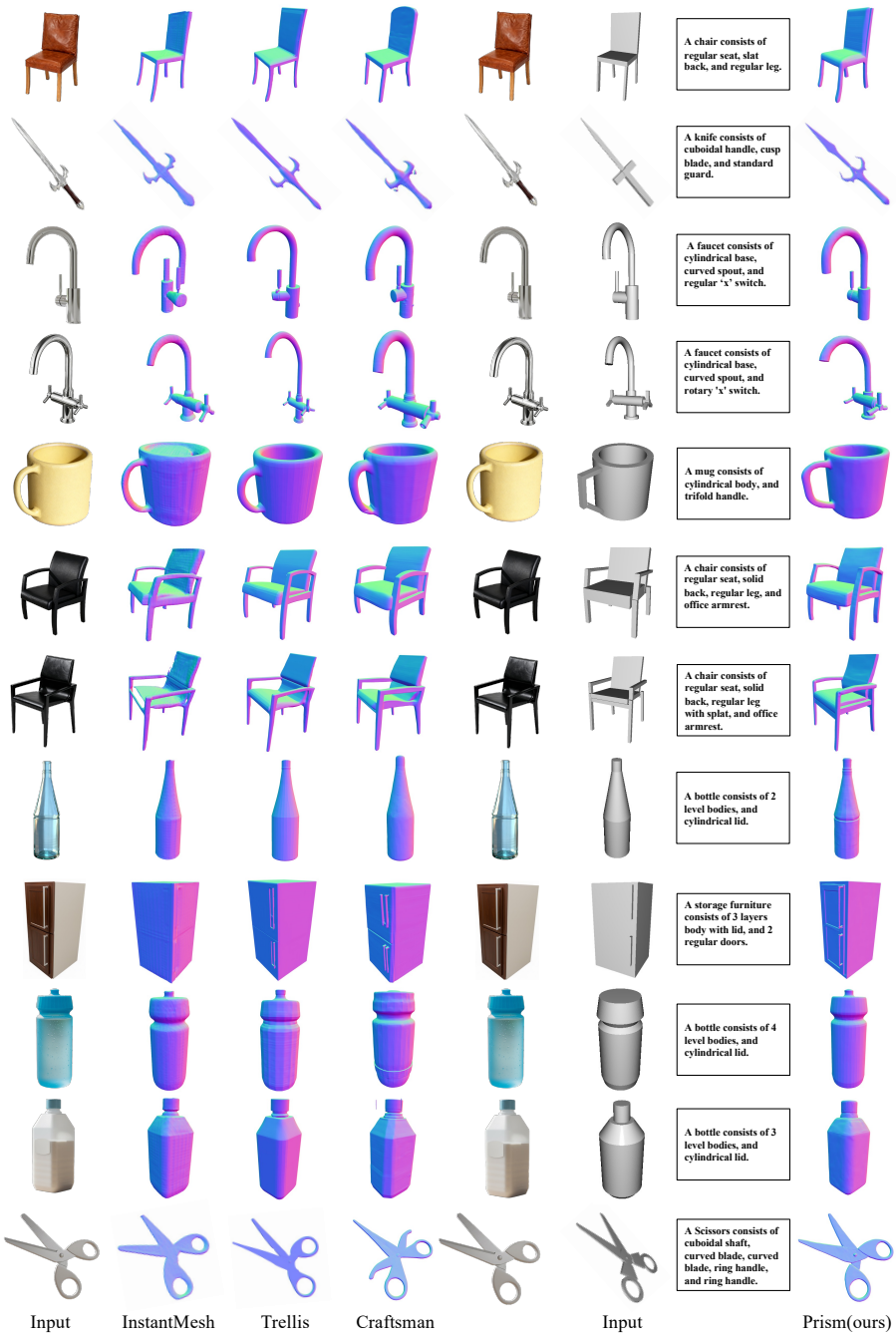


Figure 18: Qualitative comparisons with baseline methods on generation results.

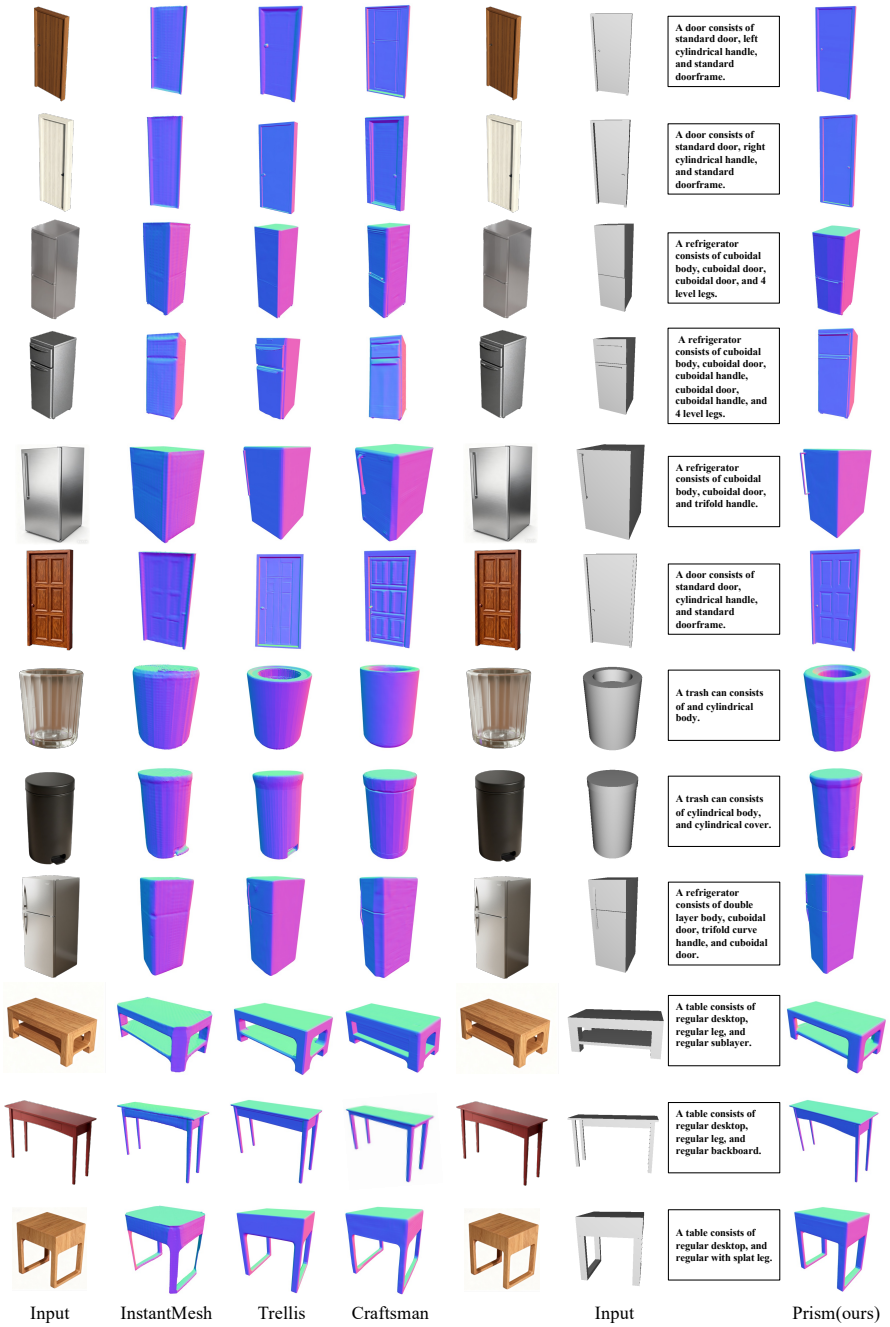


Figure 19: Qualitative comparisons with baseline methods on generation results.