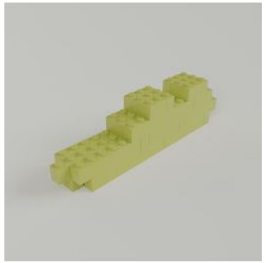
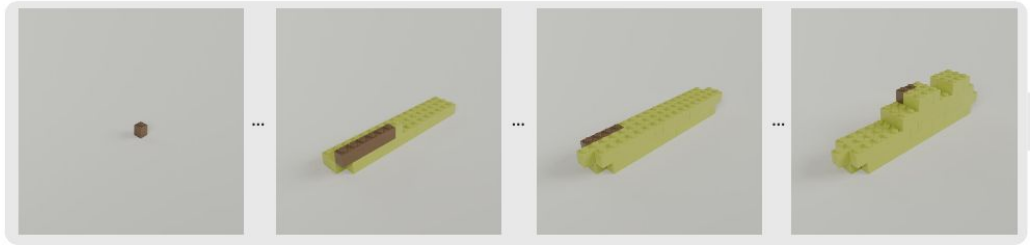
Generating Physically Stable and Buildable Brick Structures from Text

Ava Pun, Kangle Deng, Ruixuan Liu, Deva Ramanan, Changliu Liu, Jun-Yan Zhu
Carnegie Mellon University
ICCV'25 Best Paper Award (Marr Prize)

Input Text Prompt: “A streamlined vessel with a long, narrow hull.”

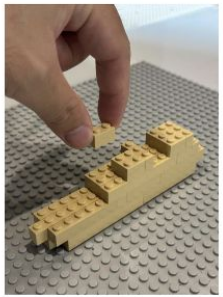


Generated Structure using LEGO Bricks



Intermediate Steps

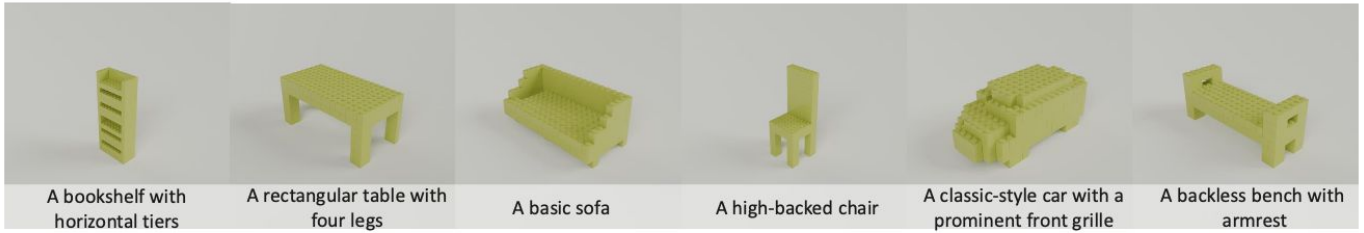
(a) Physically Stable Text-to-Brick Generation



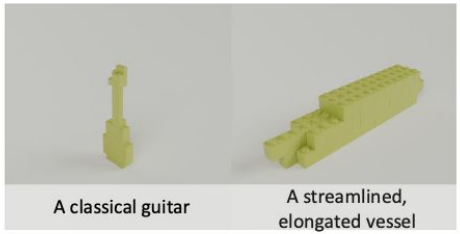
Manual Assembly following the steps

Automated Assembly by Robot Arms

(b) Real-world Assembly using LEGO Bricks

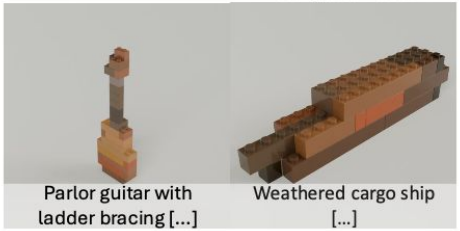


A bookshelf with horizontal tiers | A rectangular table with four legs | A basic sofa | A high-backed chair | A classic-style car with a prominent front grille | A backless bench with armrest



A classical guitar | A streamlined, elongated vessel

Gothic cathedral bookshelf [...] | Walnut wooden table [...] | Vintage floral tapestry with deep reds and golds [...] | Rustic farmhouse chair [...] | Hot rod with flame paintwork [...] | Rustic stone bench with moss growth [...]



Parlor guitar with ladder bracing [...] | Weathered cargo ship [...]

(c) Result Gallery

Motivation

- Creating real-world objects with existing methods remains challenging: many fail to be physically realized.
- Difficult to **assemble** or fabricate; physically **unstable**
- Address the challenge of generating **physically realizable objects**
- Structures made of interlocking toy bricks, e.g. LEGO blocks
- Widely used in entertainment, education, artistic creation, and manufacturing prototyping

Related work

- Text-to-3D generation
  Dreamfusion
  Diffusion models, large reconstruction models, U-Nets, and AR models
- Autoregressive 3D modeling
  LLaMA-mesh
- Brick assembly and design generation
  - Early works: optimization problem guided by heuristic rules
  - Deep graph generative model; diffusion model. Limitation: use regular building blocks instead of bricks with **interlocking** connections
- Physics-aware generation
  - Collision avoidance, contact requirements, structural stability and dynamic behavior

Dataset

- Introduce StableText2Brick
  - +47k toy brick assembly structures (from 21 object categories of the ShapeNet-Core dataset)
- Brick representation

$$B = [b_1, b_2, \ldots, b_N] \qquad b_i = [h_i, w_i, x_i, y_i, z_i]$$

position of the stud

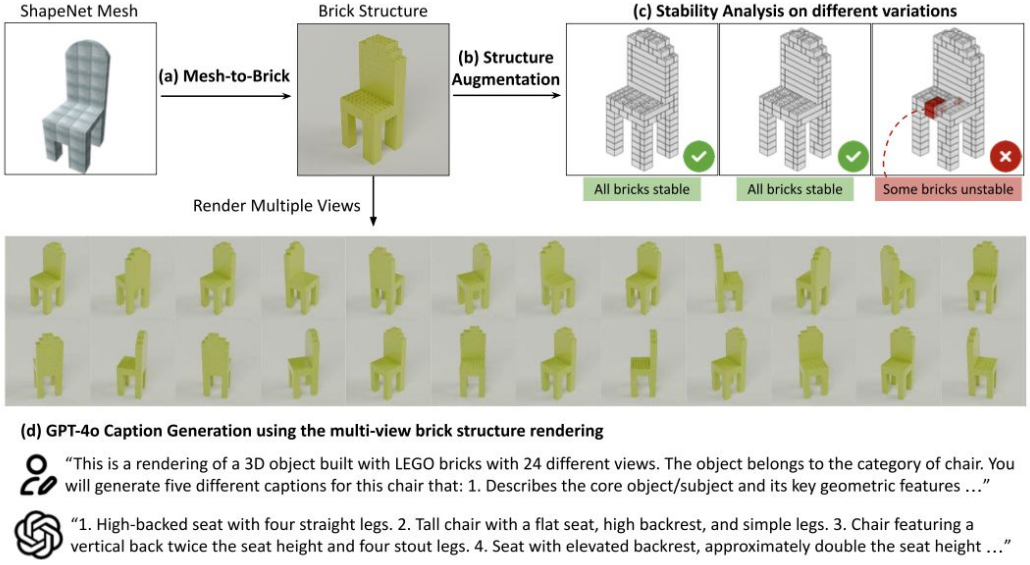
- Mesh-to-brick
  - Voxelize and downsample to 20x20x20, delete-and-rebuild
  - 8 types: 1x1, 1x2, 1x4, 1x6, 1x8, 2x2, 2x4, and 2x6

Input model | Voxelized model | LEGO model

- Stability score
  - integrating physical stability verification into autoregressive inference
  - A structure is stable if all bricks have stability scores greater than 0.
- Caption generation
  - Render from 24 viewpoints, and prompt GPT-4o (omit color)

Sheng-Jie Luo et al. Legolization: Optimizing LEGO Designs, SIGGRAPH 2015

Dataset



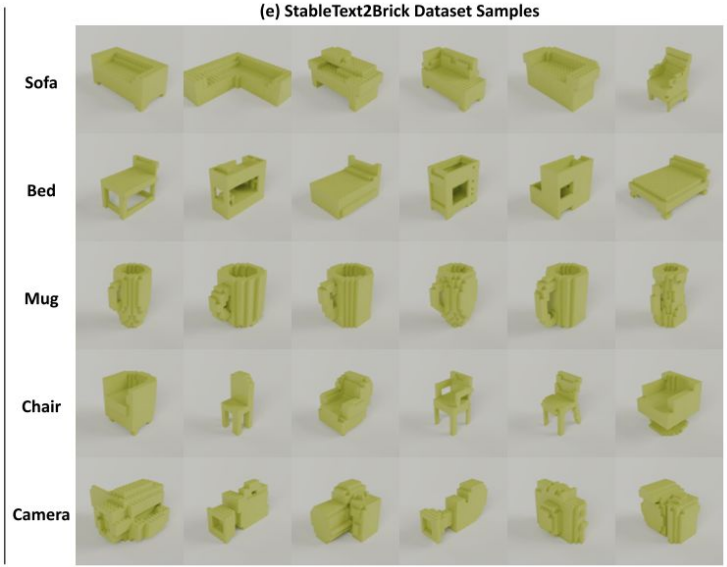


Figure 2. **StableText2Brick Dataset.** (a) From a ShapeNetCore [2] mesh, we generate a brick structure by voxelizing it onto a $20 \times 20 \times 20$ grid, then constructing its brick layout with a delete-and-rebuild algorithm. (b) We augment each shape with multiple structural variations by randomizing the brick layout while preserving the overall shape. (c) Stability analysis [38] is performed on each variation to filter out physically unstable designs. (d) To obtain captions for each shape, we render the brick structure from 24 different viewpoints and use GPT-4o [1] to generate detailed geometric descriptions. (e) Data samples from 5 categories in our StableText2Brick dataset.

Method - Model fine-tuning

- Pre-trained base model
  - LLaMA-3.2-1B-Instruct
- Instruction fine-tuning dataset
  - LDraw: LDraw™ is an open standard for LEGO CAD programs that allow the user to create virtual LEGO models and scenes.
  - LDraw limitations: no brick dimensions; contain unnecessary info (brick orientation and scale).
  - Customized format: {h}×{w} ({x},{y},{z})
- Autoregressive prediction

$$p(b_1, b_2, ..., b_N|\theta) = \prod_{i=1}^{N} p(b_i|b_1, ..., b_{i-1}, \theta)$$

Abhimanyu Dubey et al. The Llama 3 herd of models, arXiv 2024

Method - Integrating physical stability

- Motivation
  - Although trained on physically stable data, our model sometimes generates designs that violate physics and assembly constraints.
- Incorporate physical stability verification into autoregressive inference
- A structure is stable if all bricks can reach static equilibrium:

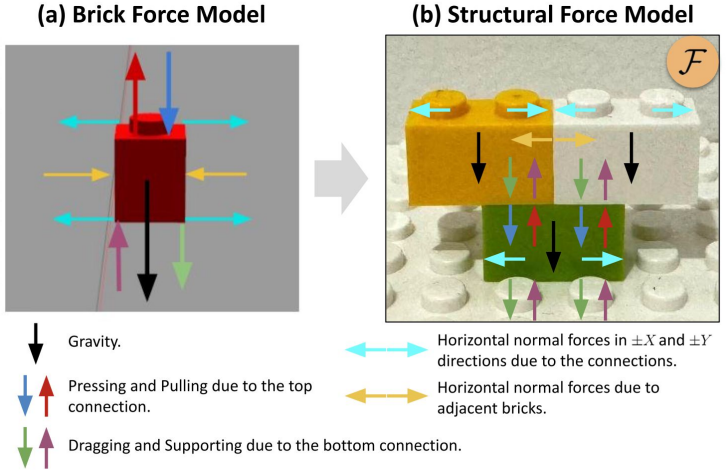
$$\sum_{j}^{M_i} F_i^j = 0, \qquad \sum_{j}^{M_i} \tau_i^j \doteq \sum_{j}^{M_i} L_i^j \times F_i^j = 0$$

- Stability analysis:

$$\arg\min_{\mathcal{F}} \sum_{i}^{N} \left\{ \left| \sum_{j}^{M_i} F_i^j \right| + \left| \sum_{j}^{M_i} \tau_i^j \right| + \alpha \mathcal{D}_i^{\max} + \beta \sum \mathcal{D}_i \right\}$$

Solver: Gurobi optimizer

D: dragging forces



Method - Integrating physical stability

- Per-brick stability score:

$$s_i = \begin{cases} 0 & \sum_j^{M_i} F_i^j \neq 0 \\ & \vee \quad \sum_j^{M_i} \tau_i^j \neq 0 \\ & \vee \quad \mathcal{D}_i^{\max} > F_T, \\ \frac{F_T - \mathcal{D}_i^{\max}}{F_T} & \text{otherwise,} \end{cases}$$

- When to apply stability analysis?
  - step-by-step validation is time-consuming
- Brick-by-brick rejection sampling
  - if a brick violates the heuristics, we resample a new brick from the model
- Physics-aware rollback
  - If the resulting design is unstable, we roll back the design to the state before the first unstable brick was generated

Method - Brick texturing and coloring

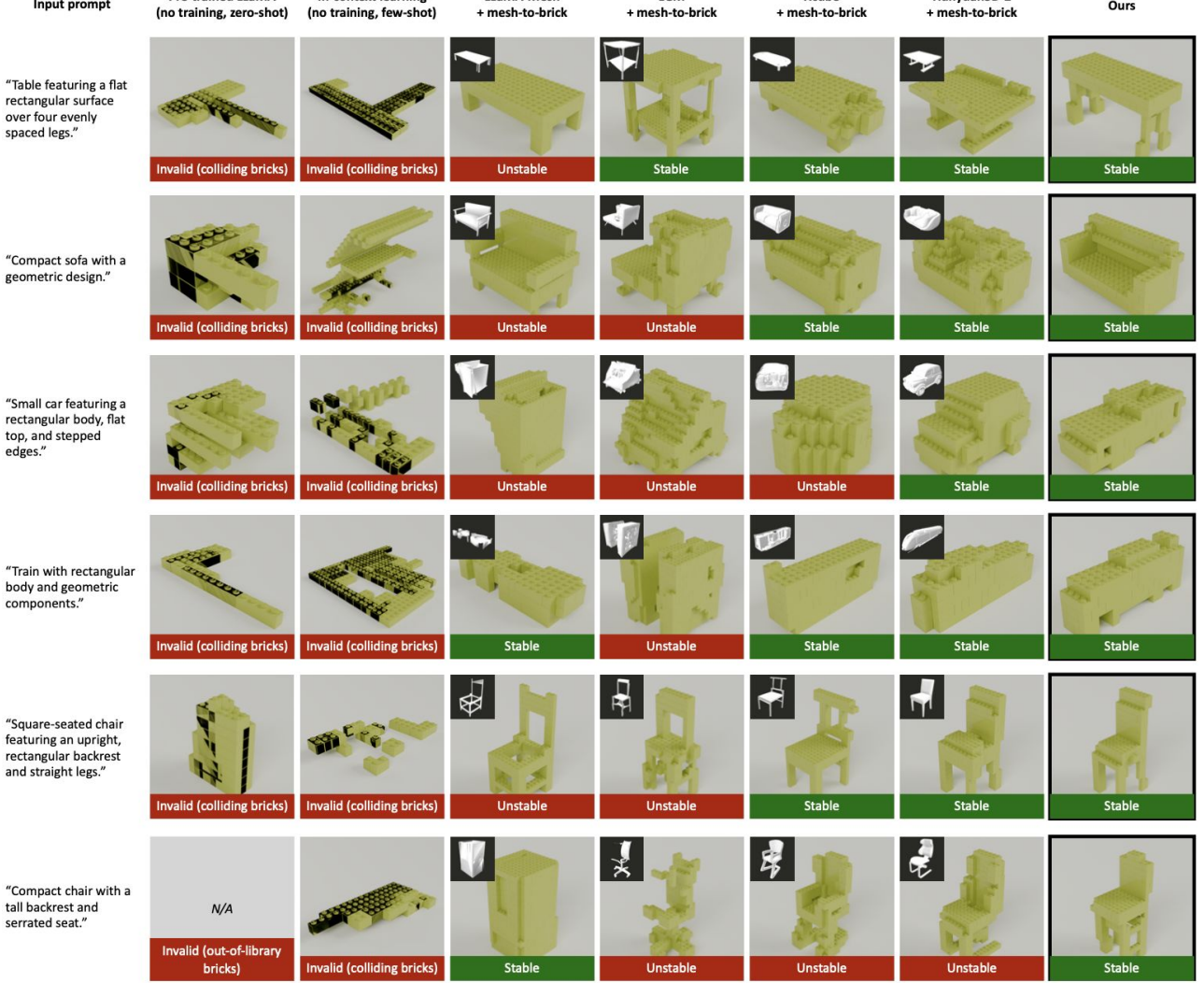
- UV texture generation
- Text-based mesh texturing approach: FlashTex

$$I_{\text{texture}} = \text{FlashTex}(\mathcal{M}, \text{UV}_{\mathcal{M}}, c)$$

- Uniform brick color assignment

Kangle Deng et al. FlashTex: Fast relightable mesh texturing with LightControlNet, ECCV 2024

Experiments



CLIP
Text: “A LEGO model of {prompt}”

Experiments

Table 1. **Quantitative Analysis.** We evaluate our method against several baselines on validity (no out-of-library, out-of-bounds, or colliding bricks), stability, CLIP-based text similarity, and DINOv2-based image similarity. Stability, CLIP, and DINO are computed over valid structures only. For LLaMA-Mesh [81], validity requires a well-formed OBJ file. Our method outperforms all baselines as well as the ablated setups on validity and stability using our proposed rejection sampling and rollback, while maintaining high text similarity.

Method	% valid	% stable	mean brick stability	min brick stability	CLIP	DINO
Pre-trained LLaMA (0-shot)	0.0%	0.0%	N/A	N/A	N/A	N/A
In-context learning (5-shot)	2.4%	1.2%	0.675	0.479	0.284	0.814
LLaMA-Mesh [81]	94.8%	50.8%	0.894	0.499	0.317	0.851
LGM [70]	**100%**	25.2%	0.942	0.231	0.300	0.851
XCube [62]	**100%**	75.2%	0.964	0.686	0.322	0.859
Hunyuan3D-2 [89]	**100%**	75.2%	0.973	0.704	0.324	0.868
Ours w/o rejection sampling or rollback	37.2%	12.8%	0.956	0.325	**0.329**	**0.888**
Ours w/o rollback	**100%**	24.0%	0.947	0.228	0.322	0.882
Ours (BRICKGPT)	**100%**	**98.8%**	**0.996**	**0.915**	0.324	0.880

Experiments - ablation + texture/color generation

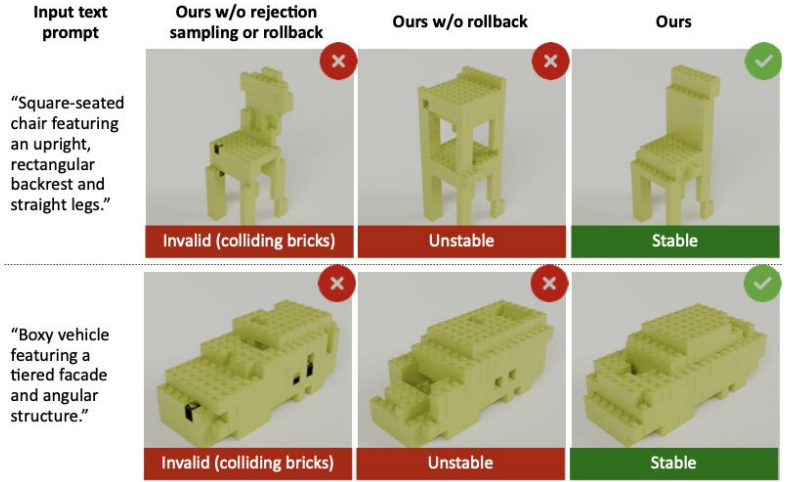


Figure 6. **Ablation study.** Brick-by-brick rejection sampling and physics-informed rollback help to ensure that the generated structure is both valid and stable. Black indicates colliding bricks.

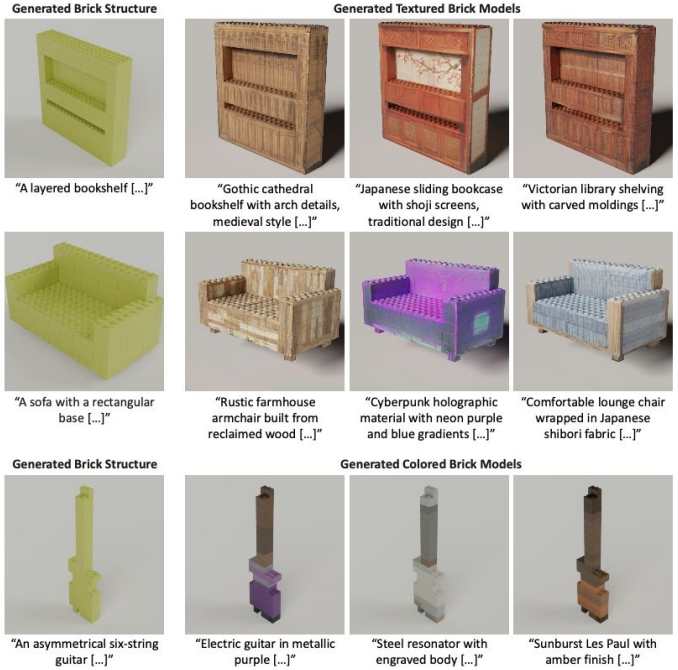
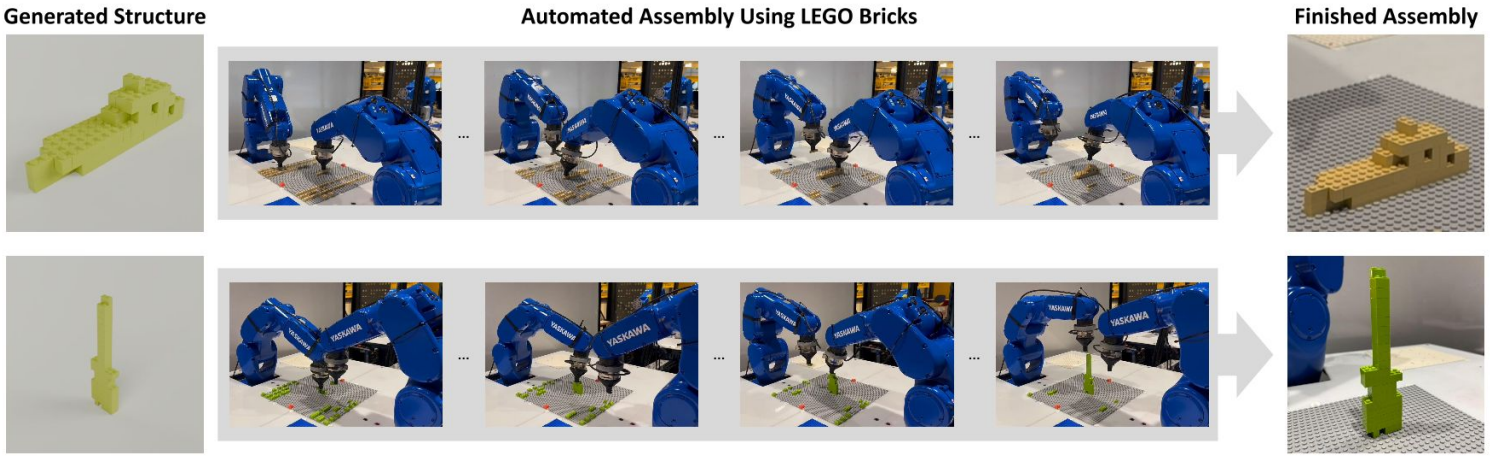


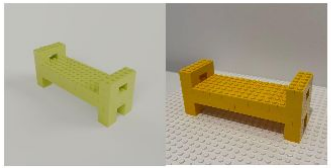
Figure 7. **Brick Texture and Color Generation.** Our method can generate diverse textured (top two rows) and colored (bottom) brick structures based on the same shape while using different appearance text prompts.

Experiments - extensions & applications

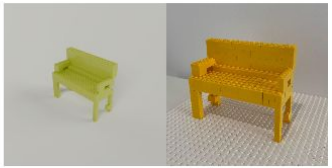
- Robotic assembly of generated structures.



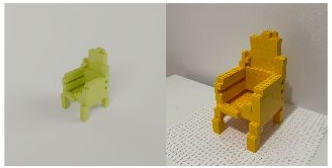
Experiments - manual assembly



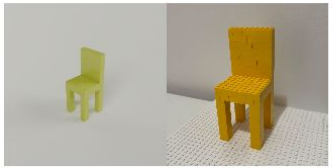
"A backless **bench** with armrest."



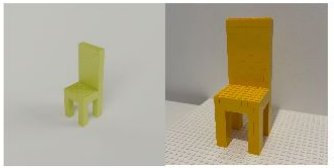
"A two-seater **bench** with a straight backrest and an open rectangular design on the sides, […]"



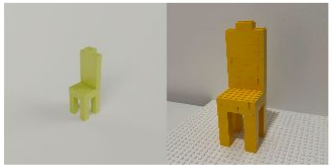
"**Chair** with cushioned backrest and seat, framed by flat, horizontal armrests supported by four straight legs."



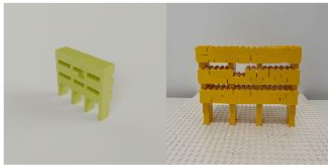
"Straight-backed **chair** with square seat."



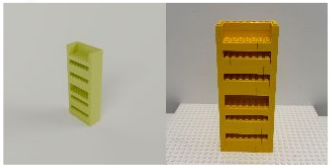
"A high-backed **chair**."



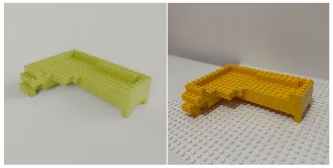
"The **chair** features an arched backrest […]"



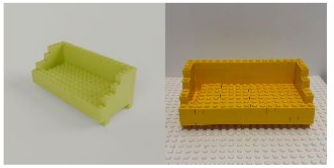
"A rectangular **bookshelf** featuring three horizontal shelves with open sides, supported by four […]"



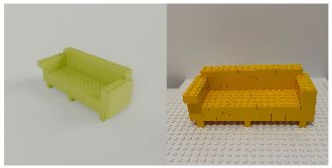
"A **bookshelf** with horizontal tiers."



"An elongated L-shaped **sofa** with a straight backrest, low armrests, and short, sturdy legs."

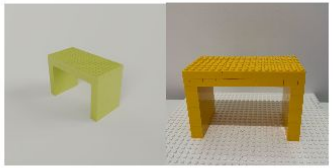


"A basic **sofa.**"

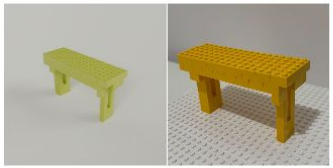


"**Sofa** with a straight backrest and angular armrests."

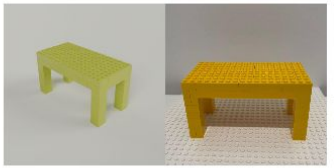
"Rectangular **table** featuring four straight legs and a flat surface."



"Simple **table** with a flat top and two side supports."



"Billiard **table** featuring a broad, rectangular surface and parallel decorative legs."



"A rectangular **table** with four legs."

Limitations

- Small scale dataset. Objaverse-XL is worth exploring.
- Only support 8 type bricks