SCOPE: Spatially-Constrained Parametric Editing for Text-Guided CAD Models

Md Zahid Hasan[†] **Soumalya Sarkar**[♮]

[†]Department of Electrical and Computer Engineering, Iowa State University, Ames, IA

[♮]RTX Technology Research Center (RTRC), East Hartford, CT

Summary: Spatially-Constrained Parametric Editing

- We propose **SCOPE**, a framework that integrates **language-guided spatial reasoning** and **relative position** into Text-to-CAD editing.
- We introduce a **Spatial Relation (<SR>) token** to explicitly model relative positions (e.g., "above," "left of") between CAD features.
- We develop an **automated data synthesis pipeline** to generate large-scale, spatially grounded CAD editing data (150K samples).
- SCOPE achieves a **+63.6%** in D-CLIP score, with **-6.2%** JSD and **-5.1%** Chamfer Distance than CAD-Editor [Yuan et al., 2025].

Background: Text-Guided CAD Editing

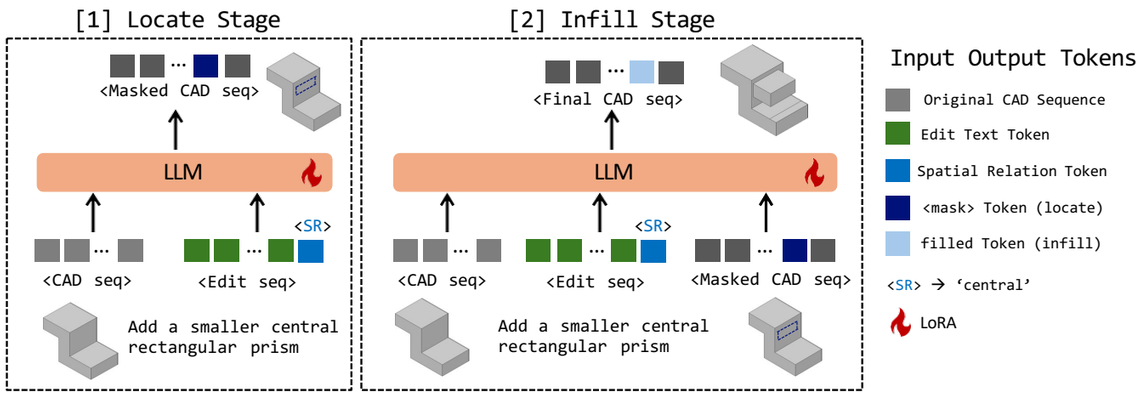
CAD Editing Challenges. CAD designers iterate through *sketch-and-extrude* and reference spatial relationships between CAD components. However, existing text-based CAD editing methods lack spatial understanding and can't interpret relative positions/geometric constraints.

Existing Approaches.

- **Design Variation** (SkexGen, Hnc-CAD, FlexCAD) → limited control
- **Text-to-CAD** (Text2CAD, CAD-LLaMA) → uneditable models
- **Text-based Editing** (CAD-Editor) → no spatial grounding

SCOPE Framework

SCOPE extends the *locate-then-infill* paradigm [Yuan et al., 2025] with hierarchy-aware spatial tokens for precise, context-aware CAD editing.



Locate Stage: Given a CAD sequence and a spatially-augmented instruction $I' = (I, \text{<SR>})$, predict a <mask> sequence:

$$C_{\text{mask}} \sim P_\theta(\cdot \mid I', C_{\text{main}})$$

Infill Stage: Generate the final edited CAD sequence by filling in the <mask> regions, conditioned on the full spatial context:

$$C_{\text{edit}} \sim P_\theta(\cdot \mid I', C_{\text{main}}, C_{\text{mask}})$$

Joint Optimization:

$$\mathcal{L}_{\text{SCOPE}}(\theta) = \sum_{(I', C_{\text{orig}}, C_{\text{edit}}) \in \mathcal{D}} [\log P_\theta(C_{\text{mask}} \mid I', C_{\text{main}}) + \log P_\theta(C_{\text{edit}} \mid I', C_{\text{main}}, C_{\text{mask}})]$$

The <SR> token transforms implicit spatial reasoning into a **structured sequence modeling task**, enabling precise and targeted editing.

CAD as Structured Text & Spatially-Guided Masking

CAD model → text sequences using sketch-and-extrude modeling (SEM) [Wu et al., 2021] with termination tokens at each hierarchy level:

Curves (line l, arc a, circle c) $\xrightarrow{\text{<curve\_end>}}$ **Loops** (L) $\xrightarrow{\text{<loop\_end>}}$ **Faces** (F) $\xrightarrow{\text{<face\_end>}}$ **Sketches** (S) $\xrightarrow{\text{<sketch\_end>}}$ **Extrusions** (E) $\xrightarrow{\text{<extrude\_end>}}$ **Model** (M)

Spatial terms (e.g., "above," "left of") are interleaved with coordinate tokens. <mask> tokens are added at curve-, loop-, or face-level based on the <SR> token, enabling precise manipulation at SEM levels.

Locate Stage Prompt:

```
Instruction: Replace the parts that need to be modified with <mask>
according to the editing instruction.
Input: <Inst>  <OrigSeq>
Output: <MaskedSeq>
```

Infill Stage Prompt:

```
Instruction: Based on the original CAD sequence. Editing instruction and masked
sequence, generate the edited CAD sequence.
Input:  <Inst> <OrigSeq> <MaskedSeq>
Output: <EditedSeq>
```

Spatially-Grounded Data Synthesis

- We develop an automated pipeline to generate **150K spatially-grounded triplets** $(I, C_{\text{main}}, C_{\text{edit}})$ from the DeepCAD dataset.
- Hnc-CAD generates $(C_{\text{main}}, C_{\text{edit}})$ sequence pairs for each CAD model.
- A Large Vision-Language Model (GPT-oss) performs **stepwise captioning**: (1) describe geometric properties, (2) identify differences, (3) compress into a concise editing instruction with spatial relations.

Example Triplet Structure:

```
{
"instruction": "Add a smaller triangular prism 7 units to the
right of the small circular hole.",
"original_sequence": "line,9,9 <curve_end>  ...",
"edited_sequence": "<edited_seq> ..."
}
```

- Unlike CAD-Editor's LCS-based masking (agnostic to spatial semantics), our <SR> tokens provide direct, unambiguous spatial guidance.

Experimental Results

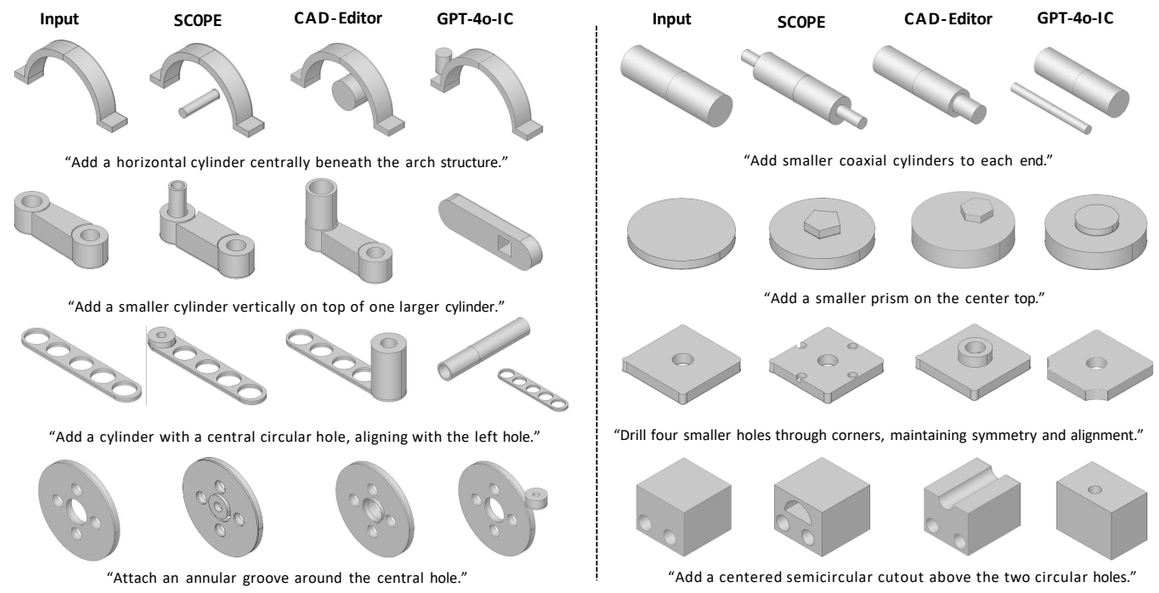
Quantitative Comparison.

Method	Backbone	VR ↑	JSD ↓	CD ↓	D-CLIP ↑
SkexGen	—	74.3	1.94	—	—
Hnc-CAD	—	77.4	1.77	—	—
FlexCAD	—	82.1	1.72	—	—
Text2CAD	Llama3-8B	84.8	2.39	1.91	—
GPT-4o-Basic	>1B	63.2	1.10	2.30	—
GPT-4o-IC	>1B	84.5	0.70	1.55	—
CAD-Editor	Llama3-8B	**95.6**	0.65	1.18	0.11
SCOPE	Gemma3-1B	75.5	1.82	1.51	0.15
SCOPE	Gemma3-12B	91.3	**0.61**	**1.12**	**0.18**

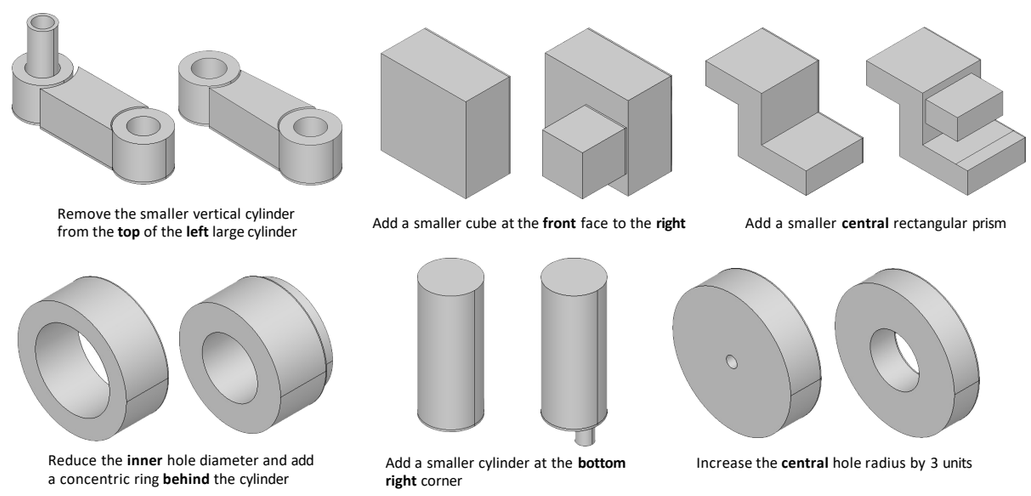
VR:Validity ratio; JSD:Jensen-Shannon divergence; CD:Chamfer dist; D-CLIP:Directional CLIP score

With Gemma3-12B, SCOPE achieves 6.2% reduction in JSD, 5.1% decrease in Chamfer Distance, and 63.6% increase in D-CLIP score over CAD-Editor. With a smaller Gemma3-1B backbone, it achieves a higher D-CLIP score than the Llama3-8B CAD-Editor model.

Qualitative Comparison.



CAD Editing with Spatial Manipulation.



Acknowledgments. This research was supported in part by the computational resources and staff contributions of the RTRC High Performance Computing Cluster.

This page does not contain any technical data.