NEURAL INFORMATION PROCESSING SYSTEMS
Europe Satellite Event
Paris 9.12 - 13.12

Interdisciplinary Centre for Security, Reliability and Trust

Text-to-CAD Evaluation with CADTESTS

Dimitrios Mallis[1], Marco Wang[1], Ahmet Serdar Karadeniz[1], Elisa Ricci[2 3], Anis Kacem[1], Djamila Aouada[1]

[1]SnT, University of Luxembourg [2]University of Trento [3]Fondazione Bruno Kessler

Luxembourg National Research Fund

GitHub

Hugging Face

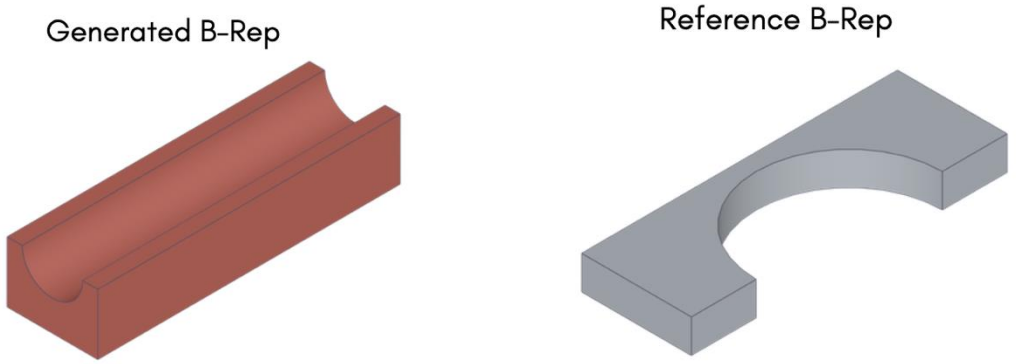
Project Page

UNIVERSITÀ DI TRENTO

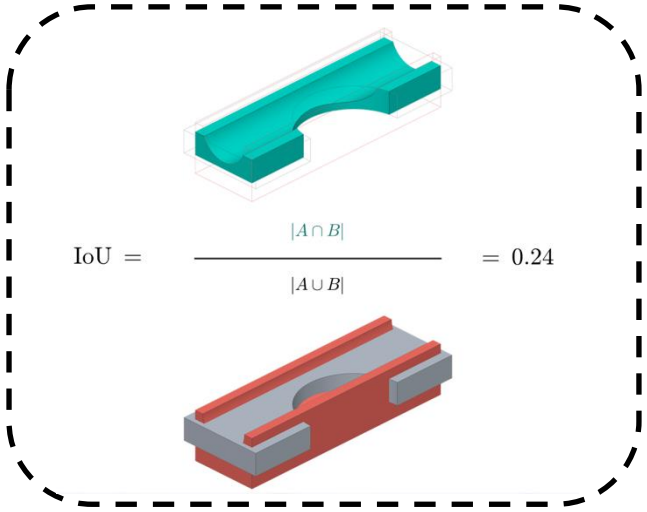
Geometry similarity metrics for Text-to-CAD

Textual Prompt

Prompt: Generate a long rectangular prism. Then cut a semi-circle.



Intersection over Union (IoU)



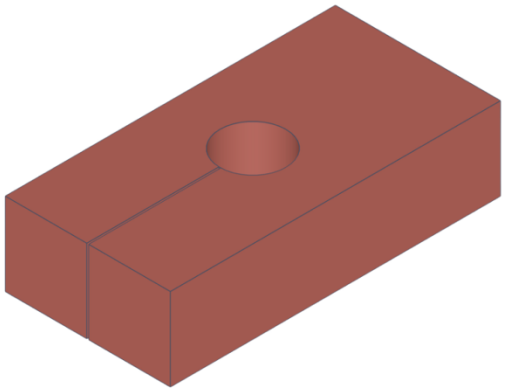
Geometric similarity metrics fail to account for the space of valid design variations within the degrees of freedom unconstrained by the prompt.

Shape similarity metrics for Text-to-CAD

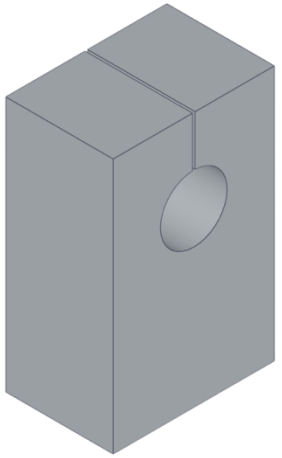
Textual Prompt

Prompt: Create a rectangular block with a circular hole at the top center and a thin wire-cut slit

Generated B-Rep



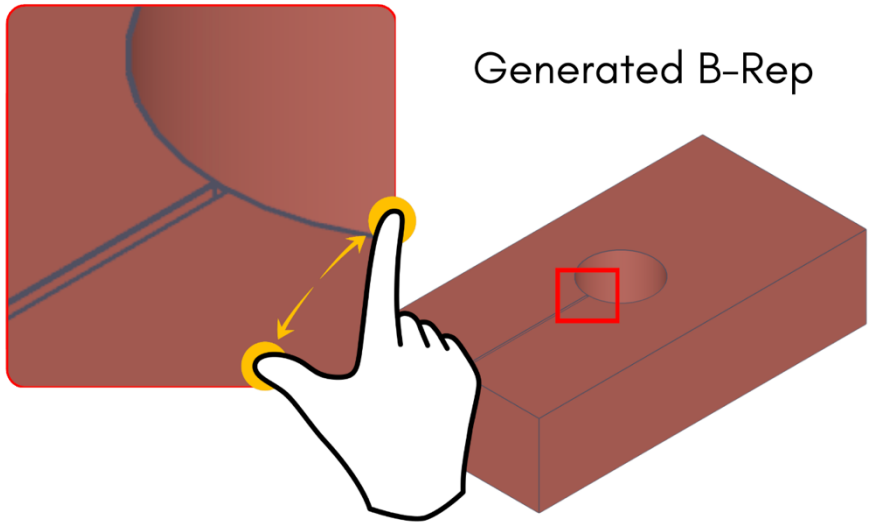
Reference B-Rep

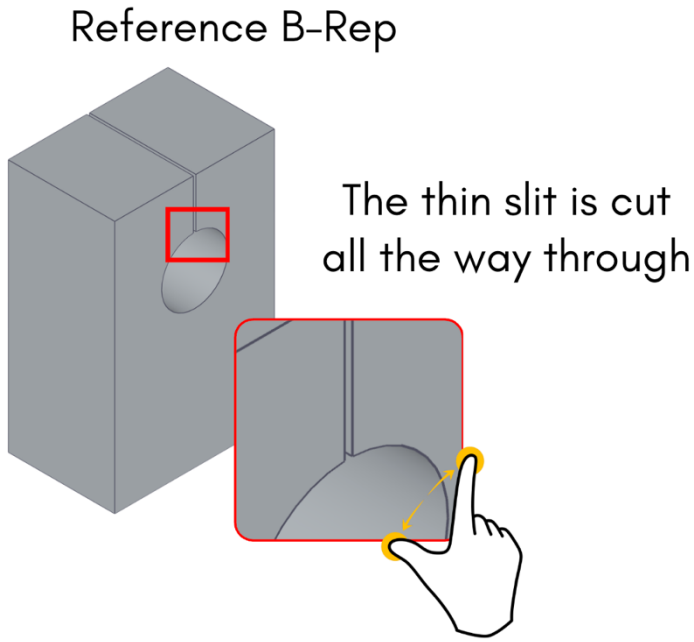


Shape similarity metrics for Text-to-CAD

Textual Prompt

Prompt: Create a rectangular block with a circular hole at the top center and a thin wire-cut slit





Geometric similarity metrics fail to identify subtle structural errors

A test-driven evaluation paradigm for Text-to-CAD

Drawing inspiration from the field of program synthesis.

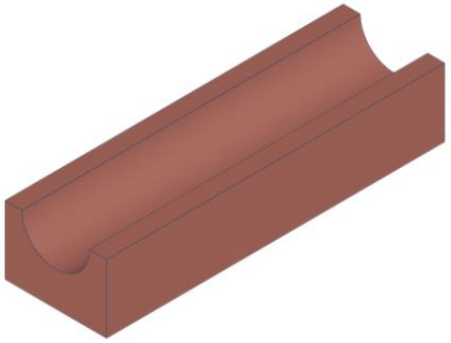
We introduce CADTESTS, **executable software tests** verifying whether a generated CAD model meets the design specifications of the input prompt

Text-driven Evaluation for Text-to-CAD

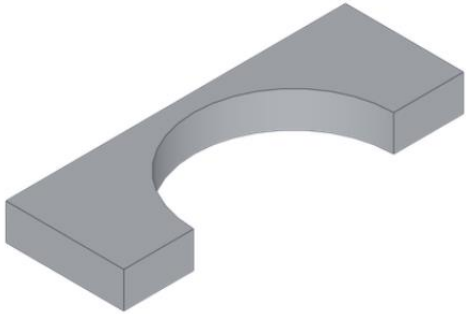
Textual Prompt

Prompt: Generate a long rectangular prism. Then cut a semi-circle.

Generated B-Rep



Reference B-Rep



CADTests

Test 1: *Verifies the model has at least one cylindrical face from the semi-circular cutout*

Test 2: *Verifies the model is elongated — the longest dimension is significantly larger than the shortest*

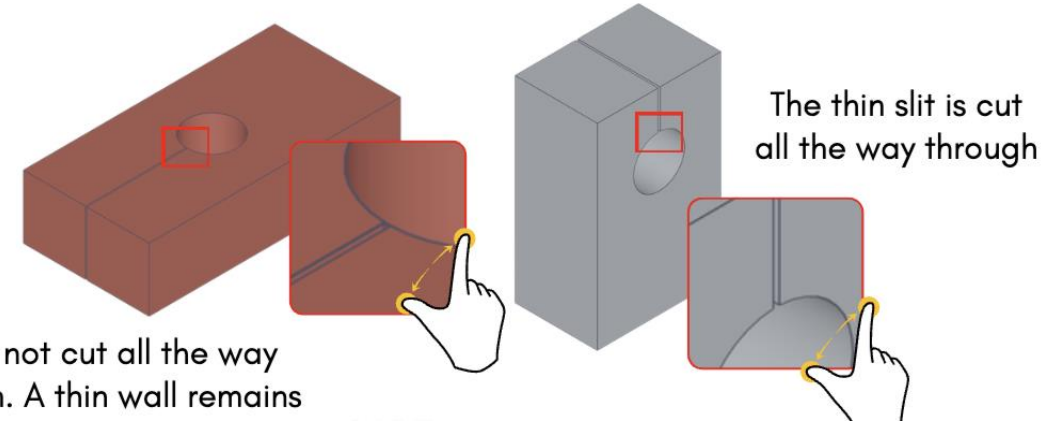
Chamfer Distance: (high) **Passed All Tests:** True

Textual Prompt

Prompt: Create a rectangular block with a circular hole at the top center and a thin wire-cut slit extending from the hole to the nearest edge.

Generated B-Rep

Reference B-Rep



CADTests

Test 4: *Verifies the total vertex count is exactly 16, matching the expected topology of block + through-hole + through-slit*

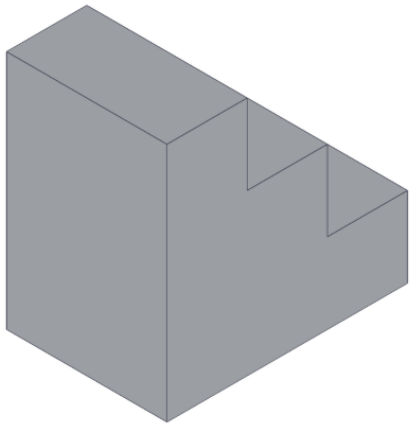
Failure: *Expected exactly 16 vertices for block with through-hole and slit, found 18*

Chamfer Distance: (low) **Passed All Tests:** False

Shape similarity metrics for Text-to-CAD

Prompt: Create a 3D object resembling a 3 step pedestal. First, you can create aside profile of the pedestal. You can do it by creating three rectangles joined to each other. First draw a big rectangle that will represent the base level. On top of this draw a smaller rectangle that represents the second step. On top of this second rectangle draw a third, even smaller rectangle, that represents the third step. Make sure the three rectangles are connected and form a closed shape. Extrude this profile to get a 3D model of the three step pedestal.

Reference BRep



CADTests

```
# Verifies the model is a single solid
solid_count = final_result.solids().size()
check(
    condition=solid_count == 1,
    pass_msg=f"Solid count is {solid_count}, matches expected 1 for a single pedestal",
    fail_msg=f"Three-step pedestal must be a single solid, found {solid_count} solids",
)
```

```
# Verifies the model has exactly 24 edges (consistent with a 3-step rectangular staircase
extrusion)
edge_count = final_result.edges().size()
check(
    condition=edge_count == 24,
    pass_msg=(
        f"Edge count is {edge_count}, matches expected 24 for a 3-step rectangular "
        f"extruded pedestal"
    ),
    fail_msg=f"Three-step rectangular pedestal should have exactly 24 edges, found
{edge_count}",
)
```

CADTests via Mutation Analysis

We develop an automated pipeline for CADTESTS synthesis and their iterative refinement, drawing on the established software engineering technique of Mutation Analysis

Generating CADTests via mutation analysis

Prompt

Create a cylinder, it should be short but have a large diameter. Then create a short and narrow rectangular prism, with a length slightly longer than the cylinder. It should sit horizontally on top of the cylinder.

Mutation Description

This mutation violates the requirement that the rectangular prism should be narrow. Here, the board width is made very wide, resulting in a rectangular prism that is far from narrow and dominates the top of the cylinder.

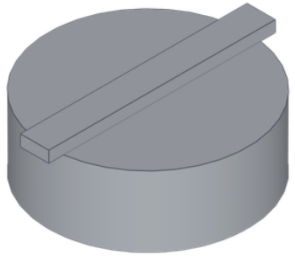
Reference Implementation

```python
import cadquery as cq

def create_cad() -> cq.Workplane:

    cylinder_diameter:float = 1.31858
    cylinder_height:float = 0.468766
    cylinder:cq.Workplane = (
        cq.Workplane("XY")
        .cylinder(cylinder_height, cylinder_diameter/2)
        )

    board_length:float = 1.5
    board_width:float = 0.1875
    board_height:float = 0.075
    board:cq.Workplane = (
        cq.Workplane("XY")
        .box(board_length, board_width, board_height)
        )

    part:cq.Workplane = (
        cq.Workplane("XY")
        .union(cylinder.translate((0, 0, -cylinder_height/2)))
        .union(board.translate((0, 0, board_height/2)))
    )

    return part
```

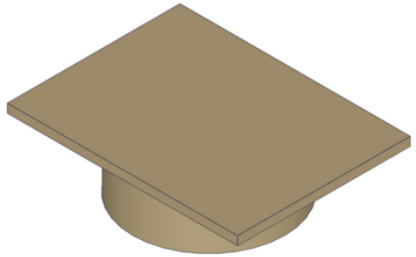
Mutated CAD Program

```python
import cadquery as cq

def create_cad() -> cq.Workplane:

    cylinder_diameter:float = 1.31858
    cylinder_height:float = 0.468766
    cylinder:cq.Workplane = (
        cq.Workplane("XY")
        .cylinder(cylinder_height, cylinder_diameter/2)
        )

    board_length:float = 1.5
    board_width:float = 2.0 # Very wide
    board_height:float = 0.075
    board:cq.Workplane = (
        cq.Workplane("XY")
        .box(board_length, board_width, board_height)
        )

    part:cq.Workplane = (
        cq.Workplane("XY")
        .union(cylinder.translate((0, 0, -cylinder_height/2)))
        .union(board.translate((0, 0, board_height/2)))
    )

    return part
```

Reference B-Rep

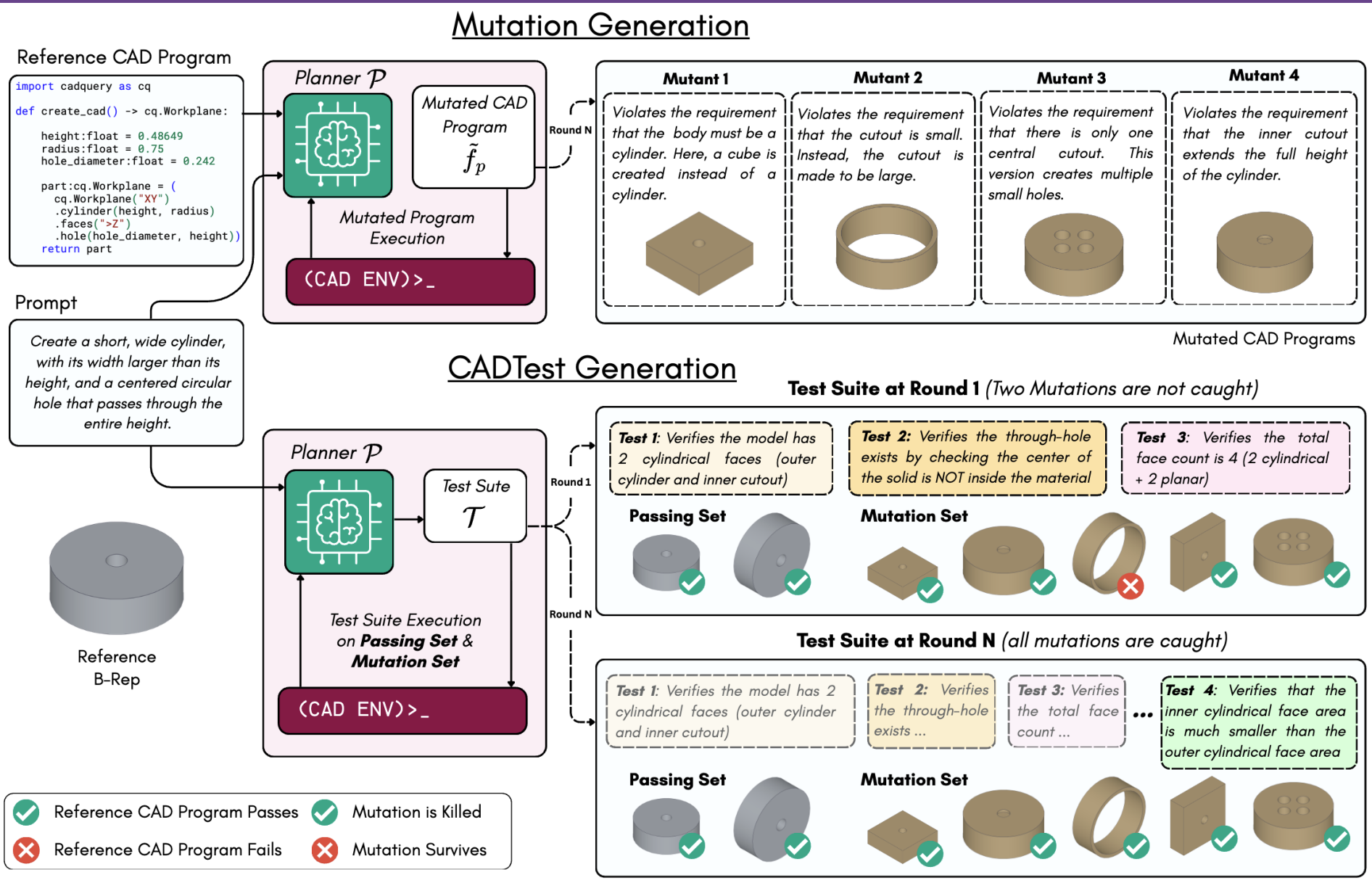


Mutated B-Rep Model



A mutant is a faulty variant of a CAD program created by injecting a controlled artificial modification into its instructions, thereby producing a corrupted geometry.

Synthesizing CADTests



A test-based benchmark for Text-to-CAD

CADTestBench

- Constructed through an automated pipeline that iteratively refines test suites using mutation analysis.
- Consists of 200 CAD programs, each paired with an abstract and a detailed prompt.
- In total 5, 937 tests for both abstract and detailed prompt with an average of $\approx$ 15 tests per sample.

Evaluation of Text-to-CAD methods on CADTESTBENCH

Method	Planner	IR ↓	RS ↑	PR ↑	IR ↓	RS ↑	PR ↑
		Detailed			*Abstract*		
Finetuned Method							
Text2CAD [21]	-	0.005	0.405	0.025	0.015	0.603	0.085
Train-Free Methods							
CadCodeVerify [1]	`GPT-4.1`	0.025	0.794	0.410	0.040	0.880	0.630
10-shots	`GPT-5.2`	0.175	0.718	0.465	0.210	0.742	0.580
Skilled	`GPT-5.2`	0.190	0.688	0.430	0.180	0.766	0.600
ReAct	`DeepSeek-V3`	0.035	0.752	0.370	0.005	0.821	0.390
ReAct	`Qwen3-Coder-480B`	0.025	0.793	0.425	0.01	0.881	0.605
ReAct	`GPT-5.2`	0	0.835	0.480	0.005	0.916	0.695
ReAct + Image	`GPT-5.2`	0.025	0.822	0.485	0.005	0.916	0.695
10-shots	`Claude-4.6-Sonnet`	0.155	0.744	0.510	0.115	0.830	0.64
Skilled	`Claude-4.6-Sonnet`	0.185	0.727	0.505	0.130	0.810	0.640
ReAct	`Claude-4.6-Sonnet`	0	0.874	0.580	0	0.929	0.715

Using CADTESTBENCH, we conduct a thorough evaluation of Text-to-CAD methods.

- **Performance on Detailed vs Abstract Prompts**: consistent observation is that all baselines achieve higher performance on abstract prompts than on detailed prompts.
- **Limitations of Synthetic Finetuning**. Text2CAD underperforms compared to training-free baselines.
- **Effect of VLM Planners on Performance**. We observe that using Claude as the planner yields a significant performance increase over GPT-based variants.
- **Integration with the CAD Kernel.** Baselines that incorporate ReAct style CAD kernel integration achieve higher performance and much lower invalidity rates.

CADTESTS as feedback mechanism during generation

Prompt

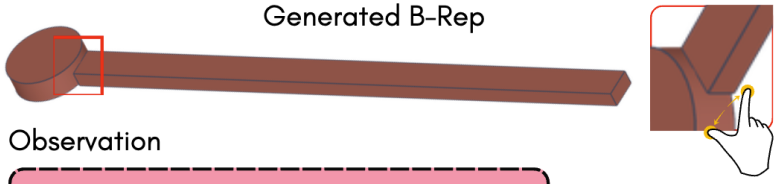
Create a 3D object resembling a thermometer. To create the object, begin by sketching a small circle and slightly extrude it. Then, add a very long rectangle at the top edge of the circle.

Round 1

Planners Thought

1. Create a small circle and extrude it slightly.
2. Add a long, narrow rectangle at the top edge of the circle, extruded slightly less than the bulb. The rectangle should be centered on the circle and extend upwards.
3. Combine both parts into a single solid.

Generated B-Rep



Observation

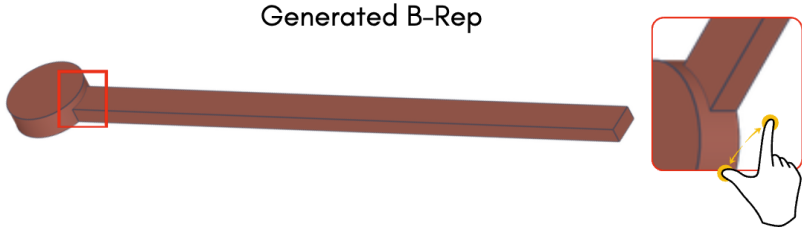
***Assertion Error**: Expected 1 solid, got 2.*

Round 2

Planners Thought

The union produced 2 solids instead of 1. This is because the stem and bulb only touch at a line which is not sufficient for a proper boolean union. I need to make the stem overlap with the bulb slightly so the union creates a single connected solid. I'll move the stem center slightly so the stem starts a bit inside the bulb.

Generated B-Rep



Generation trajectory from the CADTest baseline. The model produces both the CAD program and a set of tests used to verify the generated geometry. Execution logs are returned to the planner as feedback, revealing that the boolean union resulted in two separate solids.

Evaluation of Text-to-CAD methods on CADTESTBENCH

Method	Planner	IR ↓	RS ↑	PR ↑	IR ↓	RS ↑	PR ↑
			Detailed			*Abstract*	
Finetuned Method							
Text2CAD	-	0.005	0.405	0.025	0.015	0.603	0.085
Train-Free Methods							
CadCodeVerify	`GPT-4.1`	0.025	0.794	0.410	0.040	0.880	0.630
10-shots	`GPT-5.2`	0.175	0.718	0.465	0.210	0.742	0.580
Skilled	`GPT-5.2`	0.190	0.688	0.430	0.180	0.766	0.600
ReAct	`DeepSeek-V3`	0.035	0.752	0.370	0.005	0.821	0.390
ReAct	`Qwen3-Coder-480B`	0.025	0.793	0.425	0.01	0.881	0.605
ReAct	`GPT-5.2`	0	0.835	0.480	0.005	0.916	0.695
ReAct + Image	`GPT-5.2`	0.025	0.822	0.485	0.005	0.916	0.695
10-shots	`Claude-4.6-Sonnet`	0.155	0.744	0.510	0.115	0.830	0.64
Skilled	`Claude-4.6-Sonnet`	0.185	0.727	0.505	0.130	0.810	0.640
ReAct	`Claude-4.6-Sonnet`	0	0.874	0.580	0	0.929	0.715
Test-based Baselines							
CADTests	`GPT-5.2`	0.020	0.824	0.500	0.015	0.925	0.72
CADTests + Log	`GPT-5.2`	0.015	0.852	0.515	0.020	0.922	0.700
CADTests	`Claude-4.6-Sonnet`	0.005	0.882	0.590	0	0.953	0.765
CADTests + Log	`Claude-4.6-Sonnet`	0	**0.897**	**0.625**	0	**0.962**	**0.810**

CADTests baselines constitute the most effective feedback mechanism, yielding a consistent performance increase across all metrics.

NEURAL INFORMATION PROCESSING SYSTEMS
Europe Satellite Event
Paris 9.12 - 13.12

Interdisciplinary Centre for Security, Reliability and Trust

Thank you !

GitHub

Hugging Face

Project Page

Luxembourg National Research Fund

UNIVERSITÀ DI TRENTO