REALCADBENCH: EVALUATING PARAMETRIC CAD MODELIMNG VIA REAL INDUSTRIAL DESIGN INTENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

As coding and visual capabilities of frontier models approach towards singularity, a growing interests of using them for CAD modeling are widely witnessed. However, existing CAD benchmarks lack *comprehensive* diversity of design intents, most of which solely focus on *renders* as inputs. We introduce **RealCADBench**, a benchmark with four slices covering an abundant *real* industrial design intents for intent-to-program CAD modeling with a delicate eval protocol. It contains 12,632 tasks from 19 Factory Automation related product categories and spans *text descriptions*, *2D engineering drawings*, *real product pictures*, and *rendered images* for both *Part* and *Assembly* modeling. We evaluate the exported model with executability, Solid IoU, Surface IoU, and a rubric-based visual-semantic identity Judge. Among the nine standalone frontier foundation models we evaluate, no model leads all four metrics. Across the six frontier-scale foundation models, executability ranges from 0.565 to 0.812, Solid IoU from 0.2841 to 0.5379, and Surface IoU from 0.112 to 0.217 across the four Part regimes. The highest regime-balanced composite comes from a different model than the leaders on the four component metrics. On RCB-Assm25, Codex with GPT-5.5 improves executability and both IoU metrics over standalone GPT-5.5, but lowers the Judge by 6.98 pp, leaving GPT-5.5 as the Judge leader. We also observe recurring failure modes, most notably missing fine structures, loss of part identity, and incorrect assembly placement. These results show that execution alone is insufficient to characterize realistic CAD modeling and that frontier models and agents differ substantially across executability, IoUs, and visual-semantic identity.

1 INTRODUCTION

For parametric CAD modeling, a runnable program is only a partial indicator of success. A syntactically valid program may still omit mounting holes from a bracket. A plausible render may still place the wrong components in an assembly. Executability, IoUs with a stated target, and visual-semantic identity therefore do not necessarily move together. Evaluating systems by executability alone, or by a single IoU score, misses these failure modes.

The problem becomes harder on real industrial design intents. Inputs may come from text specifications, 2D engineering drawings, real product pictures, or rendered images. Each source reveals a different and often incomplete view of the intended geometry. Interface details and repeated structures can be decisive even when they are only partially visible. Evaluation in this setting must therefore distinguish whether a system can produce a valid artifact, recover the underlying shape, and preserve the visible identity of the target part or assembly.

ABC (Koch et al., 2019), Fusion 360 Gallery (Willis et al., 2021), DeepCAD (Wu et al., 2021), and SketchGraphs (Seff et al., 2020) provide large geometric corpora and construction histories. Text2CAD (Khan et al., 2024) and CAD-Recode (Rukhovich et al., 2025) evaluate generation conditioned on language or observations. More recent benchmarks expand the scope further by introducing mixed inputs (Wang et al., 2026; Chen et al., 2026; Doris et al., 2026), editable or executable outputs (cadbench.ai, 2026), and semantic, assembly, or engineering criteria (Zhang et al., 2026; Yang et al., 2026; Dong et al., 2026; Singh et al., 2026). Most existing benchmarks still emphasize synthetic or CAD-native settings, limited input modalities, or executability and geometric IoUs

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.

alone. As a result, they often fail to separate executability, IoUs, and product identity in realistic industrial settings. They also usually evaluate standalone models rather than agents.

We introduce **RealCADBench** (Figure 1), a benchmark for intent-to-program CAD modeling from real industrial design intents. It contains 12,632 tasks from 19 factory-automation categories. These tasks span text descriptions, 2D engineering drawings, real product pictures, and rendered images, and cover both Part and Assembly modeling. We report results on a 1,770-task evaluation slice consisting of 1,745 Part tasks across four input regimes and RCB-Assm25, a stratified 25-task assembly study used in every reported assembly comparison. Each method generates FreeCAD API Python, which a shared runtime executes to export `result.stl` as the final 3D model. We evaluate the exported model using executability, Solid IoU, Surface IoU, and a rubric-based visual-semantic identity Judge that compares rendered views of the exported model with the original inputs. In this paper, we use "agent" to refer to a harness configuration rather than a standalone model.

Our experiments show three main patterns. First, executability, IoUs, and visual-semantic identity separate in practice. Among the nine standalone frontier large models we evaluate, no model leads all four metrics. The model with the highest regime-balanced composite is also different from the leaders on the four component metrics. Second, difficulty does not follow a single order across input regimes. Across the six frontier-scale large models, executability ranges from 0.565 to 0.812, Solid IoU from 0.2841 to 0.5379, and Surface IoU from 0.112 to 0.217 across the four Part regimes. Third, frontier models and agents differ substantially once evaluation goes beyond execution alone. On RCB-Assm25, Codex with GPT-5.5 improves executability and both IoU metrics over standalone GPT-5.5, but lowers the Judge by 6.98 pp, leaving standalone GPT-5.5 as the Judge leader. We also observe recurring failure modes, most notably missing fine structures, loss of part identity, and incorrect assembly placement.

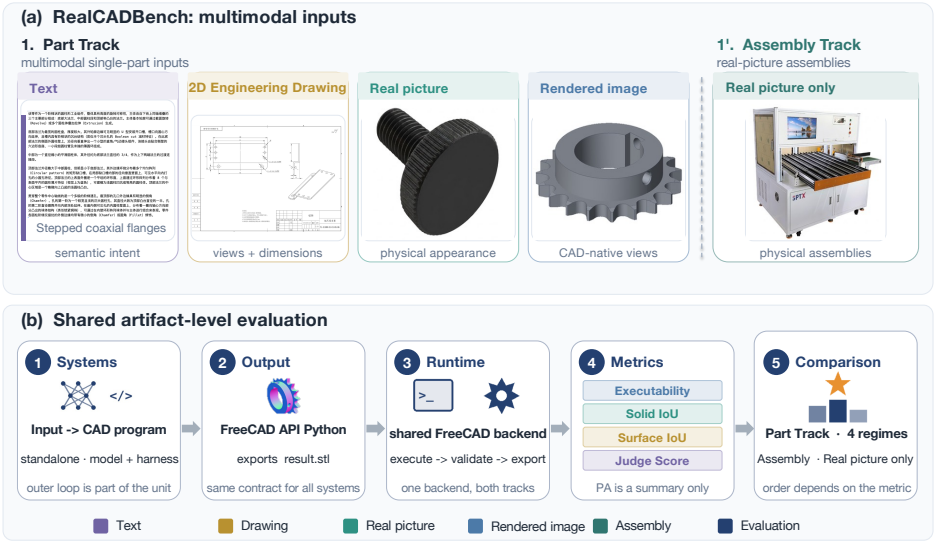


Figure 1: Overview of RealCADBench. (a) Four Part input regimes and the real picture Assembly track. (b) Shared evaluation: methods generate FreeCAD API Python, the common runtime exports the final 3D model, and scoring reports four metrics.

The paper makes three contributions:

- **A comprehensive benchmark for real industrial design intents**: RealCADBench curates 12,632 tasks across text, 2D engineering drawing, real picture, and rendered image inputs, spanning both Part and Assembly modeling under one shared FreeCAD artifact contract. It covers 19 factory-automation categories and is designed to evaluate frontier multimodal large models and state-of-the-art agents in the intent-to-program setting.
- **A systematic evaluation protocol beyond executability and IoUs**: RealCADBench reports executability, Solid IoU, Surface IoU, and a rubric-based visual-semantic identity Judge for both

parts and assemblies. The Judge complements the IoUs by assessing product identity, salient features, and assembly relations. The component scores and their denominators remain the basis for interpretation when Profile Average is reported (Equation 1).

- **A comparative study of frontier models and agents on realistic CAD modeling**: RealCAD-Bench evaluates nine standalone frontier large models on the Part Track and two state-of-the-art agents, together with their paired standalone models, on RCB-Assm25. The resulting quantitative and qualitative analyses show that different metrics favor different systems and reveal recurring failure modes in intent-to-program CAD modeling.

2 Related Work

Representative geometry and construction-history corpora. ABC (Koch et al., 2019), Fusion 360 Gallery (Willis et al., 2021), DeepCAD (Wu et al., 2021), and SketchGraphs (Seff et al., 2020) provide large corpora of CAD geometry, design sequences, command histories, and constrained sketches. Text2CAD (Khan et al., 2024) extends this line of work to language-conditioned generation, while CAD-Recode (Rukhovich et al., 2025) recovers executable code from point clouds. These resources broaden the range of CAD modeling tasks that can be studied, but they do not evaluate executability, IoUs, and visual-semantic identity as distinct outcomes for real design intents.

CAD benchmarks beyond a single geometry score. Recent CAD benchmarks expand evaluation beyond a single geometry score in several ways. Text2CAD-Bench (Wang et al., 2026), UniCAD (Chen et al., 2026), and CADBench Doris et al. (2026) study mixed-input settings. Parametric CAD Bench (cadbench.ai, 2026) and CADGenBench (Hugging Face, 2026) focus on executable or editable outputs. BenchCAD (Zhang et al., 2026), P3D-Bench (Yang et al., 2026), MUSE (Dong et al., 2026), and CADEngBench (Singh et al., 2026) incorporate semantic, assembly, or engineering criteria. Despite these advances, most CAD benchmarks still focus on tasks of render images, whereas RealCADBench evaluates real industrial design intents and reports four distinct measures.

Execution and inference-time repair. Zero-to-CAD (Ataei et al., 2026) and the reference agent in CADGenBench (Hugging Face, 2026) execute and revise programs at inference time. In such settings, runtime feedback, tool use, and compute budget become part of the evaluated method. RealCADBench therefore compares each agent with its paired standalone model on the same assembly tasks. Table 1 places this evaluation choice in the context of prior CAD benchmarks.

Table 1: Selected CAD benchmarks along input, output, and system-level evaluation axes.

Work	Input	Track	Evaluated output	Beyond geometry	Agent
DeepCAD (Wu et al., 2021)	CAD sequence	Part	Parametric sequence	×	×
Text2CAD (Khan et al., 2024)	Text	Part	Parametric sequence	×	×
CAD-Recode (Rukhovich et al., 2025)	Point cloud	Part	Executable code	×	×
CADBench (Doris et al., 2026)	Mesh / image / render	Part	Executable program	×	×
BenchCAD (Zhang et al., 2026)	Text / image / code	Part	CadQuery program	✓	×
P3D-Bench (Yang et al., 2026)	Text / image	Part + assembly	Parametric program	✓	×
CADGenBench (Hugging Face, 2026)	2D Engineering Drawing / CAD edit	Part	STEP/BREP	✓	✓
MUSE (Dong et al., 2026)	Text specification	Assembly	Executable B-Rep	✓	×
CADEngBench (Singh et al., 2026)	Specification / CAD	Part + assembly	CAD / behavior	✓	×
RealCADBench	Text / 2D Engineering Drawing real picture / rendered image	Part + assembly	FreeCAD Python API	✓	✓

3 RealCADBench: Tasks and Protocol

3.1 Tasks and Tracks

In RealCADBench, a method takes real industrial design intents as input and produces a parametric CAD program whose export matches a reference 3D model. Each task is defined by an input x_i, a reference 3D model $C_i^\star$, and metadata for the input regime, category, and views. The method outputs a Python program p_i that calls the FreeCAD API (Riegel et al., 2026). The shared runtime executes p_i and exports `result.stl` as the final 3D model. Scoring depends only on that exported model, not on the source code or construction path.

The benchmark contains two tracks: the *Multimodal Part CAD Modeling Track* and the *Image-based Assembly CAD Modeling Track*. The Part Track contains 12,465 tasks, including 568 text descriptions, 236 2D engineering drawings, 11,288 real product pictures, and 373 rendered images. The Assembly Track contains 167 real picture tasks. In this paper, we report a 1,770-task evaluation slice consisting of 1,745 Part tasks across four input regimes, namely 568 text descriptions, 236 2D engineering drawings, 568 real product pictures, and 373 rendered images, together with *RCB-Assm25*, a 25-task assembly study used in all reported assembly comparisons. Table 2 summarizes statistics of the full benchmark and the reported subset. Figure 2 demonstrates representative inputs and their reference ground truths.

Table 2: Benchmark totals and reported evaluation sets. All regimes use verified benchmark references. More details appear in next paragraphs and Appendix A.1.

Track	Input regime	Benchmark tasks	Reported tasks
Part	Text	568	568
Part	2D Engineering Drawing	236	236
Part	Real picture	11,288	568
Part	Rendered image	373	373
Assembly	Real picture	167	25
Total tasks		12,632	1,770

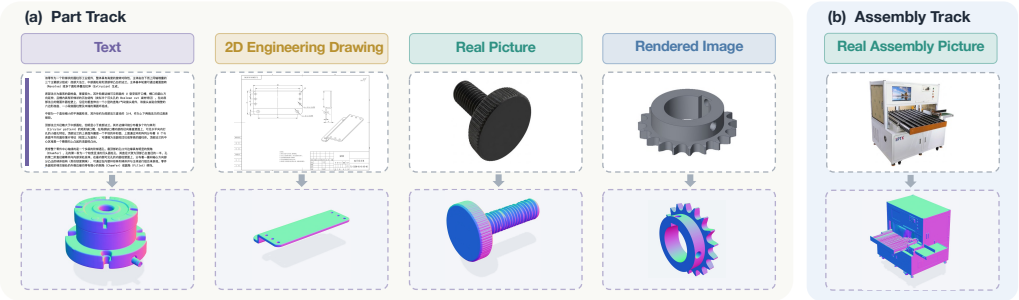


Figure 2: Representative inputs and reference 3D models.

3.2 Data and References

Reference construction. All benchmark references are verified before evaluation. For rendered image tasks, the reference 3D model is exported from the same verified STEP asset used to generate the input render image(s). For real product pictures, 2D engineering drawings, and text, paired vendor CAD asset is usually unavailable, so the benchmark reference is firstly generated with state-of-the-art image-to-3d model and then scored by CAD experts. Text tasks are derived from corresponding real picture cases through Gemini 3.0 Pro generating text descriptions with a style of CAD modeling, and the Judge evaluation for the text regime therefore uses the corresponding real picture input. Geometric scores are computed against the verified benchmark reference for each task. Appendix A.1 provides the construction details.

Data curation. Candidates come from an industrial marketplace and a purchased repository of CAD assets, including STEP files, 2D drawings, and metadata. We remove corrupted, duplicate, ambiguous, or otherwise unusable records. For visual inputs, this screening removes corrupted files, exact or near duplicates, records with ambiguous product identity, and samples that cannot be associated with a consistent product version. For CAD assets, we require successful parsing, non-empty geometry, and successful rendering or 3D model generation when the asset is used to construct benchmark tasks. Before partitioning, we group records that share a SKU, product family, CAD file, or near-duplicate image cluster to reduce overlap across benchmark splits.

Coverage. The final tasks cover 19 factory-automation categories (Appendix Figure 6), including mounting interfaces and repeated structures. Table 3 summarizes the scale of the candidate pools, and Appendix A.1 provides additional construction details.

Table 3: Source pools before final curation.

Source	Raw scale	Filtered	Primary assets
Public industrial marketplace	$\sim$ 680k	366,487	Product images
Purchased industrial data	$\sim$ 200k	125,489	STEP files, 2D drawings, metadata

Task construction. Task construction separates proposal, screening, reference verification, and evaluation. For instances besides rendered image, we use state-of-the-art image-to-3D models to first propose reference 3D models, which are then screened for reconstruction quality and input–geometry consistency. Candidate 3D models are scored by CAD experts, and only about 17% with scores of at least 8 are retained as benchmark references. Accepted references are frozen only after practitioner cross-checks within the curation workflow. Proposal methods, screening models, evaluated frontier large models and agents, and the Judge remain distinct throughout construction and evaluation (Appendix A.1). The same grouping rules also keep Judge-development cases separate from reported evaluation. RCB-Assm25 preserves the category, component-count, visibility, and geometry-complexity strata of all 167 assemblies.

3.3 Evaluation Protocol

Required output. Every submission must create a valid FreeCAD document and export `result.stl` in millimetres. A part must contain at least one non-empty solid, while an assembly may contain multiple solids. Geometric metrics are computed on the exported union. The shared runtime keeps frontier large models and agents comparable. Feature histories, constraints, tolerances, assembly trees, and manufacturing readiness matter only when they affect the exported 3D model. The visual-semantic identity Judge can assess visible structure and assembly relations, but it cannot recover parametric metadata that is not present in the export.

Scoring. *Executability* requires a valid exported artifact, so syntax errors, runtime failures, missing outputs, empty geometry, and invalid models all count as failures. After deterministic signed-PCA alignment removes pose and uniform-scale differences, *Solid IoU* measures occupied volume, while *Surface IoU* emphasizes boundaries and thin details. The rubric-based *visual-semantic identity Judge* compares rendered views of the exported model with the original inputs and never sees the reference model. For text tasks, the Judge uses the corresponding real picture evidence from which the text description was generated and validated, so its score remains independent of the geometric reference 3D model. The Part rubric assesses identity and salient features, while the Assembly rubric also scores components, placement, and system structure. To develop the Judge, we sampled benchmark input evidence and reference pairs and evaluated candidate prompts. We then iteratively refined the protocol through CAD-expert review, preference screening, and post-hoc validation before freezing the Kimi K2.6 protocol (Moonshot AI, 2026a). Appendix A.5 reports a six-system consistency check on BenchCAD, and Appendix A.6 provides the full prompt details.

For a reported setting c, let E_c, G_c^{solid}, G_c^{surface}, and J_c denote the four aggregate scores. Their displayed summary is

$$P_c = \tfrac{1}{4}\left(E_c + G_c^{\text{solid}} + G_c^{\text{surface}} + J_c/100\right). \tag{1}$$

We refer to P_c as the *Profile Average* (PA). Here E_c uses every assigned task, whereas each quality term uses only artifacts available to its evaluator. Section 4 explains the component columns together with their denominators. Appendix A.2 gives the alignment, voxelization, and aggregation details.

4 Experiments

We study three questions on the reported evaluation slice: **RQ1.** *Do executability, geometric IoUs, and the visual-semantic identity Judge favor the same frontier large models?* **RQ2.** *Does difficulty follow a single order across the four Part regimes, or does that ordering depend on the metric and*

the input type? **RQ3.** *What recurring failure modes remain when frontier large models and agents are applied to realistic CAD modeling tasks?*

4.1 SETUP

Models. We evaluate nine standalone frontier large models on the reported Part slice: Qwen3-VL-8B Bai et al. (2025), Qwen3-VL-32B Bai et al. (2025), Qwen3.8-27B Qwen Team (2026), Claude Opus 4.8 Anthropic (2026b), Gemini 3.1 Pro Preview Google DeepMind (2026a), GPT-5.4 OpenAI (2026b), GPT-5.5 OpenAI (2026c), Kimi K3 Moonshot AI (2026b), and Doubao Seed 2.0 Pro ByteDance Seed (2026). Qwen3-VL-8B, Qwen3-VL-32B, and Kimi K3 are open-weight, while Claude Opus 4.8, Gemini 3.1 Pro Preview, GPT-5.4, GPT-5.5, and Doubao Seed 2.0 Pro are proprietary. All request settings use the provider default. The three Qwen baselines are limited to the Part evaluation, whereas the six frontier-scale large models also enter the assembly comparison. Each run record preserves the evaluated model version, provider settings, and platform metadata.

Agent comparisons. On RCB-Assm25, we additionally evaluate Codex with GPT-5.5 OpenAI (2026a) and Claude Code with Claude Opus 4.8 Anthropic (2026a). These are general-purpose coding agents with access to files, execution, error observation, and revision, but no CAD-specific tools. We compare each agent with its standalone model on the same tasks under the recorded budget, so the evaluated object is the complete method rather than the base model alone.

Aggregation. Standalone large models share the task instruction, the FreeCAD backend, and the maximum context, while request settings and agent limits follow the recorded provider or tool defaults. PA follows Equation 1. For the Part Track, regime-balanced PA macro-averages the four regime-specific PAs, and the comparisons below are interpreted through the component columns. Appendix A.2 gives the alignment, voxelization, and aggregation details.

4.2 RQ1: DIFFERENT METRICS FAVOR DIFFERENT FRONTIER LARGE MODELS

Executability and surface recovery diverge. Table 4 reports the regime-balanced Part Track results, while Appendix A.4 provides the full regime-specific breakdown. GPT-5.5 executes successfully on 93.11% of tasks in the regime-balanced average, but its conditional Surface IoU is only 0.1393. Kimi K3 attains the highest conditional Solid IoU and Surface IoU, yet it executes on only 34.22% of tasks and reaches a Surface IoU of just 0.1657. The main finding is that surface recovery remains limited even among successful executions.

Different frontier large models lead different metrics. Gemini 3.1 Pro Preview has the highest regime-balanced PA (0.5507). GPT-5.5 leads executability and the visual-semantic identity Judge, while Kimi K3 leads both IoU metrics. No frontier large model leads all four metrics.

Among open models, Qwen3.8-27B executes on 79.61% of tasks, close to GPT-5.4. However, its conditional Surface IoU is only 0.1126 and Judge score is 56.30. From Qwen3-VL-8B to Qwen3-VL-32B, executability rises from 0.2100 to 0.3322, and Surface IoU from 0.0885 to 0.0995. For the 8B and 32B Qwen3-VL models, limited execution remains the dominant constraint.

4.3 RQ2: DIFFERENT INPUT REGIMES CHANGE THE DIFFICULTY ORDER

Rendered-image inputs lead every component. Across six frontier-scale large models, the rendered image regime leads every component: 0.8123 executability, 0.5379 Solid IoU, 0.2165 Surface IoU, and 75.80 Judge (Figure 3). The other three Part regimes also change order across metrics, so any average over the four regimes combines results from different input evidence types.

The ordering does not stay fixed beyond the rendered image regime. Among the remaining three Part regimes, no single condition leads every metric. As Figure 3 shows, the regime ordering changes once evaluation moves from executability to IoUs and then to the Judge. This confirms that input type affects different aspects of CAD modeling in different ways.

Table 4: Regime-balanced Part Track results. Each component is macro-averaged across the four input regimes, and regime-balanced PA macro-averages the four regime-specific PAs. The three rightmost columns are conditional on evaluator-available artifacts and should be read together with Exec. Appendix Tables 7–10 report the full regime-specific results. Best and second-best values are **bold** and underlined only for Exec. and regime-balanced PA.

Model	Exec.	Solid IoU (cond.)	Surface IoU (cond.)	Judge (cond.)	Regime-balanced PA
Claude Opus 4.8 (Anthropic, 2026b)	0.8922	0.4080	0.1433	68.31	0.5316
Gemini 3.1 Pro Preview (Google DeepMind, 2026a)	0.8825	0.4291	0.1578	73.33	**0.5507**
GPT-5.4 (OpenAI, 2026b)	0.8005	0.3795	0.1337	61.19	0.4814
GPT-5.5 (OpenAI, 2026c)	**0.9311**	0.3829	0.1393	73.79	0.5478
Kimi K3 (Moonshot AI, 2026b)	0.3422	0.4481	0.1657	72.23	0.4195
Doubao Seed 2.0 Pro (ByteDance Seed, 2026)	0.4315	0.3767	0.1328	59.57	0.3842
Qwen3-VL-8B (Bai et al., 2025)	0.2100	0.2833	0.0885	34.19	0.2309
Qwen3-VL-32B (Bai et al., 2025)	0.3322	0.2885	0.0995	44.07	0.2902
Qwen3.8-27B (Qwen Team, 2026)	0.7961	0.3585	0.1126	56.30	0.4575

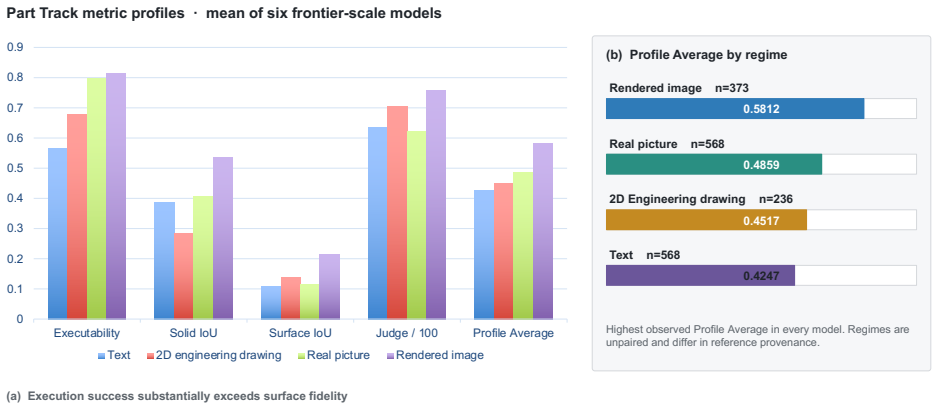


Figure 3: Part Track profiles averaged over six frontier-scale large models. Judge is divided by 100 for display. The three non-rendered regimes change order across metrics.

Assembly Solid IoU is lower than on real picture parts. Across the six shared standalone frontier large models, assembly Solid IoU is 0.2290, compared with 0.4071 on real picture parts. By contrast, executability (0.8133 versus 0.7987) and Judge (64.95 versus 62.39) are similar. Because the two studies are unmatched, the 0.1781 difference reflects two different populations rather than a paired comparison. This pattern suggests that, once multiple components must fit together, an exported model can look plausible in a render while still getting the overall solid composition wrong.

4.4 RQ3: Frontier models and agents reveal recurring failure modes

Agent comparisons expose recurring failure modes. The same separation appears in the assembly setting. GPT-5.4 and Codex + GPT-5.5 both execute successfully on all 25 tasks in RCB-Assm25 (Table 5), yet their Surface IoU values are only 0.1046 and 0.1161, respectively. Codex leads the two IoU metrics, while standalone GPT-5.5 leads the visual-semantic identity Judge. Relative to standalone GPT-5.5, Codex + GPT-5.5 improves executability by 0.1600, Solid IoU by 0.0714, and Surface IoU by 0.0190, but lowers the Judge by 6.98 pp. Relative to standalone Claude Opus 4.8, Claude Code changes PA by only +0.0006, while Surface IoU decreases from 0.1058 to 0.0864. These comparisons treat agents as complete evaluated methods under the recorded budgets. Appendix A.7 traces recurring failure modes in which execution is repaired while product identity, fine structures, or assembly placement remain weak.

Table 5: Results on RCB-Assm25, the stratified study used for all reported assembly comparisons. Executability uses all 25 tasks. The three rightmost columns are conditional on evaluator-available artifacts and should be read together with Exec. Judge is reported on a 0–100 scale. Best and second-best values are **bold** and underlined only for Exec. and PA.

Model	PA	Exec.	Solid IoU (cond.)	Surface IoU (cond.)	Judge (cond.)
Codex + GPT-5.5 (OpenAI, 2026a)	**0.5228**	**1.0000**	0.2817	0.1161	69.35
Claude Code + Claude Opus 4.8 (Anthropic, 2026a)	0.4644	0.8800	0.2732	0.0864	61.80
Claude Opus 4.8 (Anthropic, 2026b)	0.4638	0.8800	0.2670	0.1058	60.24
Gemini 3.1 Pro Preview (Google DeepMind, 2026a)	0.4724	0.9200	0.2306	0.0979	64.10
GPT-5.4 (OpenAI, 2026b)	0.5028	**1.0000**	0.2366	0.1046	66.98
GPT-5.5 (OpenAI, 2026c)	0.4777	0.8400	0.2103	0.0971	76.33
Kimi K3 (Moonshot AI, 2026b)	0.4338	0.7200	0.2665	0.1039	64.48
Doubao Seed 2.0 Pro (ByteDance Seed, 2026)	0.3389	0.5200	0.1632	0.0969	57.56

5 Discussion

The results indicate that RealCADBench measures complementary aspects of system performance, including executability, geometric IoUs, and visual-semantic identity Judge. This pattern is already evident in Tables 4 and 5. Figure 4 further shows that systems with similar geometric IoUs can nonetheless differ substantially in Judge scores. Appendix A.3 provides the aggregate values underlying the modality-profile and track-comparison analyses.

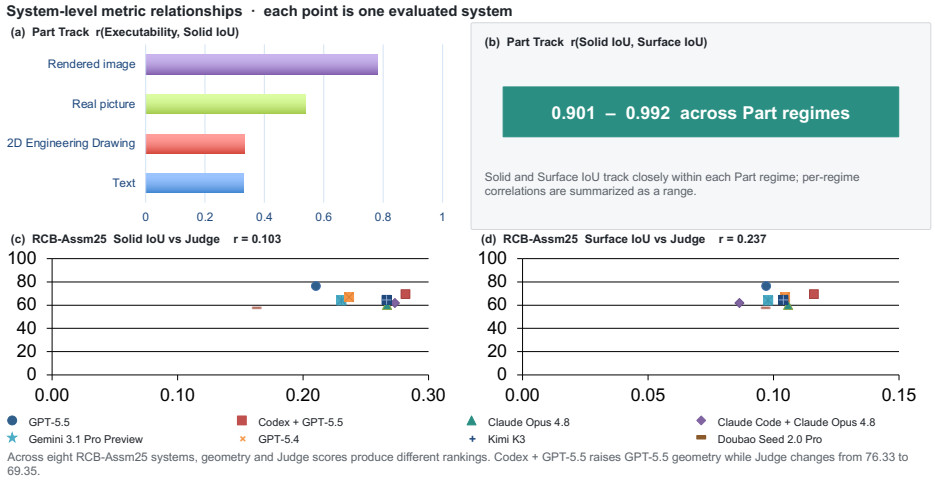


Figure 4: System relationships vary by metric. Solid and Surface IoU are tightly coupled within Part regimes, whereas geometric IoUs and Judge scores are only weakly correlated across RCB-Assm25 configurations ($n = 8$).

Different metrics favor different frontier large models. RQ1 shows that the reported metrics do not rank systems in the same way. The model that leads on executability is not the one that leads on IoUs, and the highest regime-balanced PA comes from a different model again. This difference matters because the metrics answer different questions. Executability asks whether a system can produce a valid artifact at all. The IoUs ask how much of the target geometry is recovered once such an artifact exists. The quality columns are computed only on tasks with valid executions, so they should be read together with the number of tasks that actually enter each average. Otherwise, a model that succeeds on few tasks can still look strong because its IoUs or Judge is averaged over a much smaller subset.

Different input regimes shift the difficulty order. RQ2 shows that the four Part regimes do not follow a single difficulty order. The rendered image regime is consistently the easiest under the reported metrics, but the other three regimes change order as evaluation moves from executability to geometric IoUs and then to the Judge. This pattern suggests that the benchmark is not measuring

one generic notion of CAD difficulty. Instead, different input types make different parts of the CAD modeling problem difficult. Some regimes mainly test visible shape recovery, whereas others put more weight on hidden geometry inference or identity preservation from incomplete evidence.

Agents alter the trade-offs among metrics. RQ3 shows that agents should be treated as distinct evaluated methods rather than as simple wrappers around the same base model. In the assembly setting, tool use and iterative repair can improve execution success and sometimes improve geometry, but they do not necessarily improve visual-semantic identity. An agent loop can fix syntax errors, execution failures, and some construction mistakes without necessarily correcting the core design decision. The comparison is therefore between complete evaluated methods, not base models alone. Each method includes its harness, tools, and budget.

The Judge captures what IoUs miss. The visual-semantic identity Judge matters because IoUs do not fully capture salient features and assembly relations. It compares the exported model with the input evidence rather than with the reference model. It therefore complements the IoUs instead of duplicating them. In practice, it asks whether the model still looks like the intended object, not just how much geometry overlaps after alignment. RealCADBench therefore reports the Judge as a separate metric rather than treating IoUs alone as sufficient. The consistency check in Appendix A.5 supports this choice, and expert post-checks remain part of validation on evaluated outputs.

Recurring failure modes explain the metric separation. The qualitative examples help explain why the reported metrics diverge. Some outputs match the overall shape yet omit thin structures or small interfaces that matter for use. Others obtain non-trivial IoUs while still missing the visual characteristics that determine product identity, particularly when hidden geometry is not specified by the input. For assemblies, rendered views may make local component placements appear reasonable even when the resulting solid composition is incorrect. These failure modes reinforce that executability, IoUs, and visual-semantic identity measure different aspects of quality.

6 Conclusion

We introduced **RealCADBench**, a benchmark for intent-to-program CAD modeling from real industrial design intents, including text, 2D engineering drawings, real product pictures, rendered images, and multi-part assemblies. It places these tasks under a shared FreeCAD artifact contract and evaluates outputs with four measures, including executability, Solid IoU, Surface IoU, and a rubric-based visual-semantic identity Judge.

The results show that no single metric is sufficient for realistic CAD modeling. Frequent successful execution does not imply strong IoUs, and stronger IoUs do not guarantee recovery of product identity or assembly structure. Across the reported Part and Assembly studies, different systems lead different metrics, and the difficulty order shifts across input regimes. Agent configurations should also be treated as complete evaluated methods rather than as wrappers around the same base model. The qualitative analyses further show recurring failure modes in fine structures, part identity, and assembly placement. These findings matter for how realistic CAD modeling is evaluated. Executability, IoUs, and visual-semantic identity should be reported separately rather than collapsed into a single score or treated as if execution alone were sufficient. RealCADBench is designed to make these trade-offs visible on realistic industrial design intents.

In the future, we see three immediate directions. Firstly, we will publicly release the full RealCADBench benchmark. Secondly, we will make agentic systems the first citizen in our evaluations report results. Thirdly, we would like to compare a broader set of specialized AI-for-CAD models or algorithms alongside frontier large models and general-purpose agents.

AI USE STATEMENT

We used generative AI tools, namely Codex and ChatGPT, primarily to check and correct grammatical errors in individual paragraphs of the manuscript. We also used ChatGPT to assist with a literature survey of recent AI-based CAD benchmarks. The authors independently verified the identified studies and their descriptions, selected relevant references, and supplemented the survey for the Related Work section. The authors reviewed all AI-assisted content and take full responsibility for the final manuscript.

REFERENCES

Anthropic. Claude Code overview. https://docs.anthropic.com/en/docs/claude-code/overview, 2026a. Accessed August 13, 2026.

Anthropic. Introducing Claude Opus 4.8. https://www.anthropic.com/news/claude-opus-4-8, 2026b. Published May 28, 2026; accessed August 13, 2026.

Mohammadmehdi Ataei, Farzaneh Askari, Kamal Rahimi Malekshan, and Pradeep Kumar Jayaraman. Zero-to-CAD: Agentic synthesis of interpretable CAD programs at million-scale without real data. *arXiv preprint arXiv:2604.24479*, 2026.

Shuai Bai et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025. URL https://arxiv.org/abs/2511.21631.

ByteDance Seed. Seed2.0 model card: Towards intelligence frontier for real-world complexity. https://research.doubao.com/en/seed2, 2026. Accessed August 13, 2026.

cadbench.ai. Parametric cad bench. https://cadbench.ai/zh, 2026. Accessed August 13, 2026.

Jingyuan Chen, Sheng Jin, Haopeng Sun, Wentao Liu, and Chen Qian. UniCAD: A unified benchmark and universal model for multi-modal multi-task CAD. *arXiv preprint arXiv:2606.05058*, 2026.

Xiaoyu Dong, Zhi Li, and Xiao-Ming Wu. MUSE: Benchmarking manufacturable, functional, and assemblable text-to-CAD generation. *arXiv preprint arXiv:2605.28579*, 2026. URL https://arxiv.org/abs/2605.28579.

Anna C. Doris, Jacob Thomas Sony, Ghadi Nehme, Era Syla, Amin Heyrani Nobari, and Faez Ahmed. CADBench: A multimodal benchmark for ai-assisted CAD program generation. *arXiv preprint arXiv:2605.10873*, 2026. URL https://arxiv.org/abs/2605.10873.

Google DeepMind. Gemini 3.1 Pro model card. https://deepmind.google/models/model-cards/gemini-3-1-pro/, 2026a. Published February 19, 2026; accessed August 13, 2026.

Google DeepMind. Gemini 3 Pro model card. https://deepmind.google/models/model-cards/gemini-3-pro/, 2026b. Model release November 2025; last updated May 2026; accessed September 1, 2026.

Hugging Face. CADGenBench: A benchmark for ai-driven CAD generation and editing. https://github.com/huggingface/cadgenbench, 2026. Accessed: 2026-08-13.

Mohammad Sadil Khan, Sankalp Sinha, Talha Uddin Sheikh, Didier Stricker, Sk Aziz Ali, and Muhammad Zeshan Afzal. Text2CAD: Generating sequential CAD designs from beginner-to-expert level text prompts. In *Advances in Neural Information Processing Systems*, volume 37, pp. 7552–7579, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/0e5b96f97c1813bb75f6c28532c2ecc7-Paper-Conference.pdf.

Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. ABC: A big CAD model dataset for geometric deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9601–9611, 2019.

Moonshot AI. Meet Kimi K2.6: Advancing Open-Source Coding. https://forum.moonshot.ai/t/meet-kimi-k2-6-advancing-open-source-coding/369, 2026a. Announcement, published April 21, 2026; accessed September 1, 2026.

Moonshot AI. Kimi K3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026b. URL https://arxiv.org/abs/2607.24653.

OpenAI. Codex manual. https://developers.openai.com/codex/codex-manual.md, 2026a. Accessed August 13, 2026.

OpenAI. Introducing GPT-5.4. https://openai.com/index/introducing-gpt-5-4/, 2026b. Published March 5, 2026; accessed August 13, 2026.

OpenAI. GPT-5.5 system card. https://openai.com/index/gpt-5-5-system-card/, 2026c. Published April 23, 2026; accessed August 13, 2026.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork, August 2026. URL https://qwen.ai/blog?id=qwen3.8.

Jürgen Riegel, Werner Mayer, Yorik van Havre, and FreeCAD Community. FreeCAD: Your own 3d parametric modeler. https://www.freecad.org/, 2026. Software; accessed August 13, 2026.

Danila Rukhovich, Elona Dupont, Dimitrios Mallis, Kseniya Cherenkova, Anis Kacem, and Djamila Aouada. CAD-Recode: Reverse engineering CAD code from point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Ari Seff, Yaniv Ovadia, Wenda Zhou, and Ryan P Adams. SketchGraphs: A large-scale dataset for modeling relational geometry in computer-aided design. In *ICML 2020 Workshop on Object-Oriented Learning*, pp. 8614–8624, 2020.

Harmanjot Singh, Abhra Dubey, and Jorge Alejandro Amador Herrera. CADEngBench: It looks like CAD, but does it work? evaluating parametric design, assembly reasoning, and physics simulation. *arXiv preprint arXiv:2608.09296*, 2026. URL https://arxiv.org/abs/2608.09296.

Liang Wang, Heng Meng, Zekai Xiang, Jin Liu, Pingyi Zhou, Litao Chen, and Yongqiang Tang. Text2CAD-Bench: A benchmark for LLM-based text-to-parametric CAD generation. *arXiv preprint arXiv:2605.18430*, 2026.

Karl DD Willis, Yewen Pu, Jieliang Luo, Hang Chu, Tao Du, Joseph G Lambourne, Armando Solar-Lezama, and Wojciech Matusik. Fusion 360 gallery: A dataset and environment for programmatic cad construction from human design sequences. *ACM Transactions on Graphics (TOG)*, 40(4): 1–24, 2021.

Rundi Wu, Chang Xiao, and Changxi Zheng. Deepcad: A deep generative network for computer-aided design models. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6752–6762. IEEE, 2021.

Yikang Yang, Zhanpeng Hu, Youtian Lin, Mengqi Zhou, Jingxi Xu, Feihu Zhang, Jiaheng Liu, and Yao Yao. P3D-Bench: Benchmarking MLLMs for parametric 3d generation and structural reasoning. *arXiv preprint arXiv:2606.11152*, 2026. URL https://arxiv.org/abs/2606.11152.

Haozhe Zhang, Kaichen Liu, Miaomiao Chen, Lei Li, Shaojie Yang, Cheng Peng, and Hanjie Chen. BenchCAD: A comprehensive, industry-standard benchmark for programmatic CAD. *arXiv preprint arXiv:2605.10865*, 2026. URL https://arxiv.org/abs/2605.10865.

A Appendix

A.1 Additional Benchmark Construction Details

This section adds construction details for the protocol summarized in Section 3. We describe how benchmark references are constructed and verified across regimes, why we report RCB-Assm25 in the main paper, and how release scope is defined. Figure 5 summarizes the path from source validation to final task formation.

Reference construction for non-rendered tasks. For text, 2D engineering drawing, real picture, and assembly tasks, paired vendor CAD is usually unavailable. These tasks therefore follow a staged proposal-and-verification workflow to produce verified benchmark references. Multiple candidate methods first propose candidate 3D models. Gemini-based editing then refines surviving candidates, and Gemini 3 Pro screens reconstruction quality, input quality, and image–geometry consistency (Google DeepMind, 2026b). Five professional CAD practitioners participate across the four Part regimes and the Assembly Track. They review the surviving candidates inside the curation workflow, score the candidate 3D models, and accept a reference only after cross-checks. Consistent with Section 3, only about 17% of candidates with scores of at least 8 are retained as benchmark references. Text tasks are derived from corresponding real picture cases through Gemini 3.0 Pro generation and validation, and the Judge for the text regime therefore uses the corresponding real picture evidence. Gemini 3 Pro used in curation is distinct from the evaluated Gemini 3.1 Pro Preview, and the frozen Judge is Kimi K2.6. Proposal methods, screening models, evaluated frontier large models and agents, and the Judge remain separated throughout construction and evaluation.

Reference construction for rendered image tasks. Rendered-image tasks follow the same verified-reference principle, but their references come directly from the underlying CAD asset. For each rendered image task, the source STEP asset must parse, tessellate, render, and export to a non-empty STL. The standardized input render and the benchmark reference 3D model are then generated from that same verified asset.

Why RCB-Assm25. The Assembly Track contains 167 accepted tasks and forms the benchmark pool. In this paper, we report RCB-Assm25, a stratified study used for all assembly comparisons. Stratification covers category, component count, visibility, and geometry-complexity metadata already present in the track. The study is configuration-complete: six standalone models and two agents are scored on the same 25 tasks. Mean wall-clock time on the diagnostic sheet is 12.1 min for Codex and 31.6 min for Claude Code (Appendix A.7). Future evaluations can extend the same protocol to the full 167-task track.

Selection and quality control. We apply source-specific screening followed by common eligibility checks. For visual inputs, we remove corrupted files, exact or near duplicates, records with ambiguous product identity, and samples that cannot be associated with a consistent product version. For CAD files, we require successful parsing, non-empty geometry, successful tessellation or rendering, and successful STEP-to-STL export when the asset is used in the rendered image regime. Assembly candidates must additionally contain multiple components with recoverable relative placement.

Release boundary. The final public release includes the full benchmark, together with the evaluation code and frozen prompts. It also specifies the scoring environment, including FreeCAD 1.1.1, Python 3.11, and dependency requirements. The release follows the same access conditions and source-license constraints. Per-instance release follows those same conditions. The benchmark tasks and evaluation protocol remain defined at full benchmark scope rather than through a small illustrative subset.

A.2 Metric Implementation and Aggregation Details

Section 3 introduces the four reported scores and Profile Average. This section makes the pairwise definitions, implementation choices, and aggregation sets explicit.

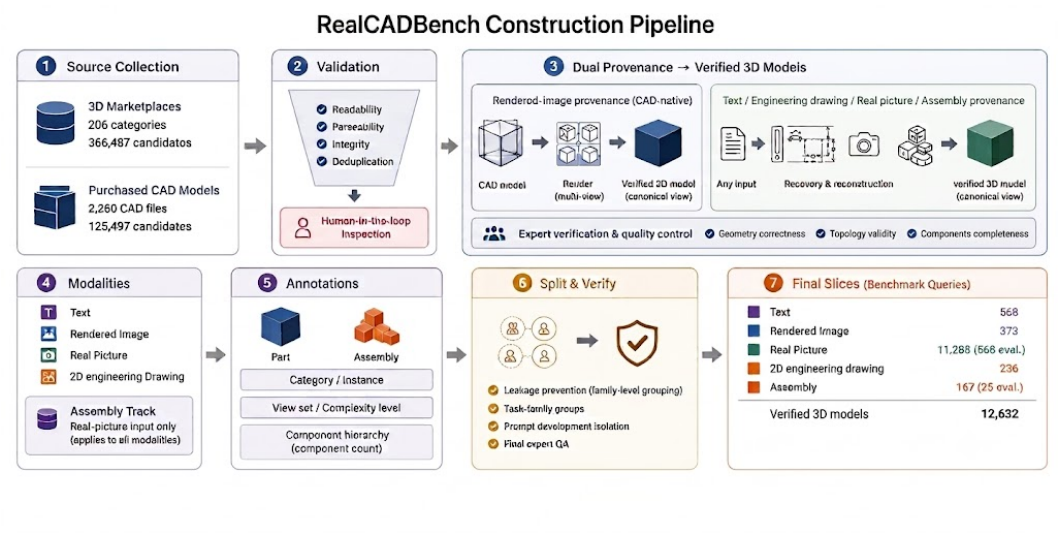


Figure 5: Construction pipeline from industrial source pools to final benchmark tasks. All reported tasks use verified benchmark references. When original vendor CAD is unavailable, the benchmark reference is constructed through multi-system proposals, Gemini editing, Gemini 3 Pro screening (Google DeepMind, 2026b), and practitioner verification. The Assembly Track contains 167 tasks, and RCB-Assm25 is the stratified study used for the reported assembly comparisons.

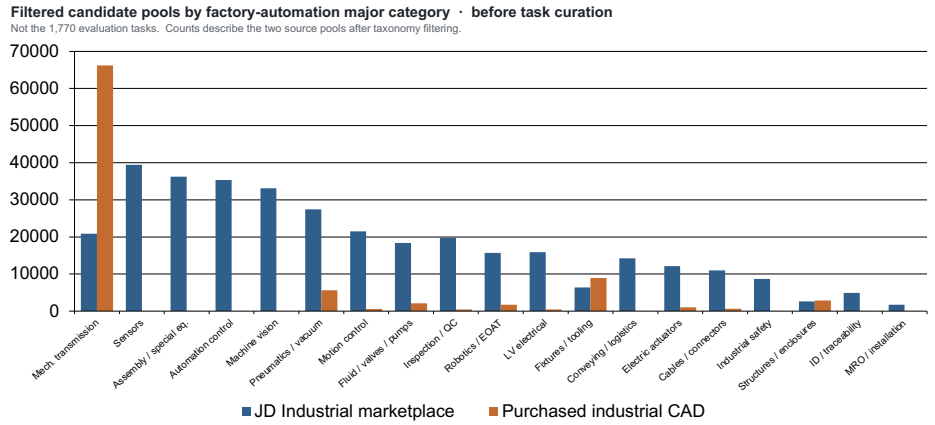


Figure 6: Category composition of the two filtered candidate pools across 19 factory-automation categories before final task curation.

Pairwise geometric metrics. Let $\tilde{C}_i$ denote the aligned prediction for task i, and $C_i^\star$ its reference. Solid and Surface IoU reuse one transform and a shared grid of resolution $R = 96$ with 2% padding:

$$g_i^{\text{solid}} = \frac{|V_{\text{filled}}(\tilde{C}_i) \cap V_{\text{filled}}(C_i^\star)|}{|V_{\text{filled}}(\tilde{C}_i) \cup V_{\text{filled}}(C_i^\star)|},$$
$$g_i^{\text{surface}} = \frac{|V_{\text{boundary}}(\tilde{C}_i) \cap V_{\text{boundary}}(C_i^\star)|}{|V_{\text{boundary}}(\tilde{C}_i) \cup V_{\text{boundary}}(C_i^\star)|}. \tag{2}$$

Filled occupancy captures volume. Boundary occupancy is more sensitive to thin structures and local detail. Because alignment includes uniform scaling, these IoUs measure geometric recovery under the stated aligned reference rather than absolute millimetre accuracy.

Evaluation sets and denominators. Let c denote a reported setting, $\mathcal{A}_c$ its assigned tasks, $\mathcal{G}_c^q$ the tasks with a value from geometry evaluator $q \in \{\text{solid}, \text{surface}\}$, and $\mathcal{J}_c$ the tasks with a Judge

result. For task i, executability is defined as

$$e_i = \mathbb{I}\Big[p_i \text{ executes and produces a valid } \hat{C}_i\Big]. \tag{3}$$

The reported aggregates are

$$\begin{aligned} E_c &= \frac{1}{|\mathcal{A}_c|} \sum_{i \in \mathcal{A}_c} e_i, \\ G_c^q &= \frac{1}{|\mathcal{G}_c^q|} \sum_{i \in \mathcal{G}_c^q} g_i^q, \qquad q \in \{\text{solid}, \text{surface}\}, \\ J_c &= \frac{1}{|\mathcal{J}_c|} \sum_{i \in \mathcal{J}_c} j_i, \qquad j_i \in [0, 100]. \end{aligned} \tag{4}$$

Executability uses every assigned task. Each quality column averages the tasks that provide a value for the corresponding evaluator. Missing quality values are neither zero-filled nor inferred from e_i. The released aggregation specification records the corresponding counts, and the component metrics are reported separately alongside PA.

PCA sign resolution. The right-handed PCA construction in Section 3 leaves two principal-axis signs ambiguous. We enumerate $S(s_0, s_1) = \mathrm{diag}(s_0, s_1, s_0 s_1)$ for $s_0, s_1 \in \{-1, +1\}$, with candidate rotation $R = A_\star^\mathsf{T} S A_{\mathrm{pred}}$. Here A_{pred} and $A_\star$ are the prediction and reference PCA bases. The selected candidate minimizes one-way mean nearest-neighbor distance from transformed prediction samples to reference samples. The subsequent uniform scale and center translation are applied once, and the resulting $\tilde{C}_i$ is reused for both metrics in Equation 2.

Voxelization and fallback. The shared grid covers the joint bounding box with padding equal to 2% of its diagonal. Its pitch is the longest padded extent divided by $R = 96$. V_{filled} fills mesh interiors, and V_{boundary} keeps only cells that intersect the mesh surface, with no dilation. When direct surface voxelization is unavailable, boundary occupancy uses a histogram of deterministically sampled surface points on the same grid.

Judge implementation. The frozen Judge model is Kimi K2.6 (Moonshot AI, 2026a). It is distinct from Gemini 3 Pro used during reference screening and from the evaluated Gemini 3.1 Pro Preview. Part Track artifacts use the Part Judge, whereas RCB-Assm25 artifacts use the Assembly Judge. For text tasks, the Judge uses the corresponding real picture evidence from which the text description was generated and validated. Judge development sampled benchmark input evidence and verified reference pairs across the four Part regimes and the Assembly Track, ran candidate prompts on those cases, and asked CAD practitioners to inspect the resulting judgments. Human review focused on rubric clarity, use of visual evidence, JSON validity, separation of part geometry from assembly relations, and reliable parser behavior. Candidate prompts were then revised and re-evaluated through repeated expert review and preference screening until practitioners reached consensus, after which the prompts were frozen for reported scoring. Expert review also includes follow-up checks on evaluated outputs. Each artifact receives one query under the frozen rubric. A separate visibility pass records `full`, `partial`, or `invisible` metadata for assembly components. These labels serve only as evidence metadata and do not change the scoring dimensions or weights. Verbatim prompts appear in Appendix A.6.

Regime-balanced PA. Equation 1 defines P_c for a reported setting c. For the Part Track, let $\mathcal{M}$ be the four input regimes. The reported regime-balanced score is

$$P_{\mathrm{part}} = \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} P_m. \tag{5}$$

This macro-average weights the four regimes equally. Judge columns remain on the original 0–100 scale and are divided by 100 only in Equation 1. Profile Average equally weights executability, Solid IoU, Surface IoU, and the normalized Judge. It provides a compact summary of the reported dimensions, while the component columns provide the corresponding breakdown.

Table 6: Mean Part Track profiles of the six frontier-scale models by input regime. Solid IoU, Surface IoU, and Judge are reported on the tasks that provide the corresponding evaluator outputs.

Input	Exec.	Solid (cond.)	Surface (cond.)	Judge (cond.)	PA
Text	0.5648	0.3871	0.1117	63.53	0.4247
2D Engineering Drawing	0.6775	0.2841	0.1397	70.55	0.4517
Real Picture	0.7987	0.4071	0.1138	62.39	0.4859
Rendered Image	**0.8123**	**0.5379**	**0.2165**	**75.80**	**0.5812**

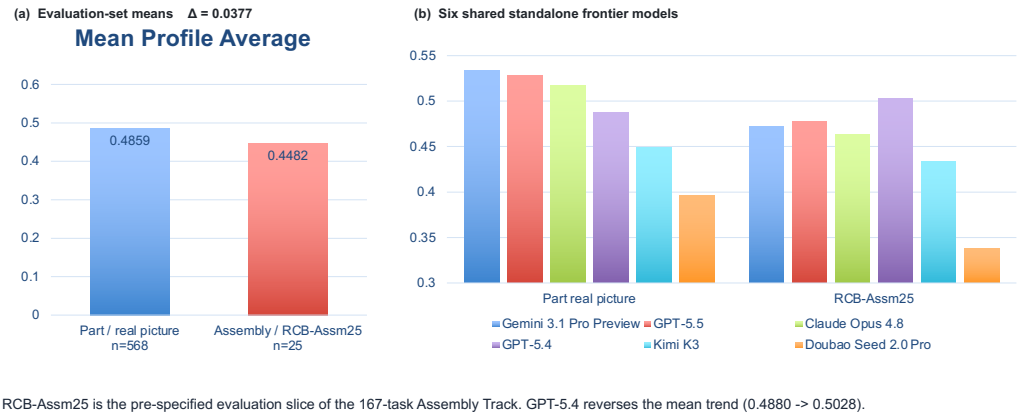


Figure 7: Profiles for the real picture Part regime and RCB-Assm25 across the six shared standalone frontier models.

Why geometry and Judge. Solid IoU and Surface IoU measure overlap with the benchmark reference 3D model after pose and uniform-scale alignment. The Judge instead compares renders of the exported mesh with the original input evidence and never sees that reference. For text tasks, it uses the corresponding real picture evidence from which the text description was generated and validated. The two families therefore capture complementary information, which is why both are reported.

A.3 Additional Aggregate Results

Table 6 reports the exact values visualized in Figure 3.

Figure 4 in the main text shows the system-level relationships among metrics. Figure 8 below shows the paired agent differences on RCB-Assm25. Instance-level diagnostic traces for both outer loops appear in Appendix A.7.

A.4 Complete Part Track Component Results

Tables 7–10 expand the compact summary in Table 4. Executability uses every assigned task. Solid IoU, Surface IoU, and Judge are reported on the tasks that provide the corresponding evaluator outputs, consistent with the reporting protocol in Section 3. Best and second-best values are highlighted only in the Exec. column.

A.5 Cross-Dataset Consistency Check for the Frozen Judge

This section reports a consistency check for the frozen Kimi K2.6 Judge. On the public BenchCAD benchmark (Zhang et al., 2026), which is separate from RealCADBench construction and Judge-prompt development, six fixed system outputs are scored by GPT-5.5 and by two independent runs each of Kimi K2.6 and Doubao. Every variant receives the same evidence and frozen prompt. Only the Judge identity or repeated query changes. This check does not use human ratings.

Table 7: Complete Part Track component results for the text regime ($n = 568$). Solid IoU, Surface IoU, and Judge are reported on the tasks that provide the corresponding evaluator outputs.

Model	Group	Exec.	Solid IoU (cond.)	Surface IoU (cond.)	Judge (cond.)
Claude Opus 4.8 (Anthropic, 2026b)	Frontier	0.6356	0.4041	0.1066	63.49
Gemini 3.1 Pro Preview (Google DeepMind, 2026a)	Frontier	0.8451	0.3995	0.1106	65.45
GPT-5.4 (OpenAI, 2026b)	Frontier	0.6984	0.3643	0.0982	59.25
GPT-5.5 (OpenAI, 2026c)	Frontier	**0.9067**	0.3509	0.0943	63.16
Kimi K3 (Moonshot AI, 2026b)	Frontier	0.0581	0.4584	0.1685	73.26
Doubao Seed 2.0 Pro (ByteDance Seed, 2026)	Frontier	0.2447	0.3451	0.0922	56.60
Qwen3-VL-8B (Bai et al., 2025)	Open 8B	0.0616	0.2575	0.0682	42.33
Qwen3-VL-32B (Bai et al., 2025)	Open 32B	0.1391	0.2507	0.0730	48.33
Qwen3.8-27B (Qwen Team, 2026)	Open 27B	0.7359	0.3600	0.0921	57.48

Table 8: Complete Part Track component results for the 2D engineering drawing regime ($n = 236$). Solid IoU, Surface IoU, and Judge are reported on the tasks that provide the corresponding evaluator outputs.

Model	Group	Exec.	Solid IoU (cond.)	Surface IoU (cond.)	Judge (cond.)
Claude Opus 4.8 (Anthropic, 2026b)	Frontier	**0.9661**	0.2811	0.1390	76.38
Gemini 3.1 Pro Preview (Google DeepMind, 2026a)	Frontier	0.8729	0.3378	0.1729	82.10
GPT-5.4 (OpenAI, 2026b)	Frontier	0.7839	0.2044	0.0959	55.53
GPT-5.5 (OpenAI, 2026c)	Frontier	0.9237	0.2817	0.1400	79.75
Kimi K3 (Moonshot AI, 2026b)	Frontier	0.1271	0.3325	0.1632	70.79
Doubao Seed 2.0 Pro (ByteDance Seed, 2026)	Frontier	0.3915	0.2672	0.1274	58.78
Qwen3-VL-8B (Bai et al., 2025)	Open 8B	0.1314	0.1653	0.0739	23.99
Qwen3-VL-32B (Bai et al., 2025)	Open 32B	0.3220	0.1824	0.0956	37.23
Qwen3.8-27B (Qwen Team, 2026)	Open 27B	0.7627	0.1973	0.1052	56.21

Table 9: Complete Part Track component results for the real picture regime ($n = 568$). Solid IoU, Surface IoU, and Judge are reported on the tasks that provide the corresponding evaluator outputs.

Model	Group	Exec.	Solid IoU (cond.)	Surface IoU (cond.)	Judge (cond.)
Claude Opus 4.8 (Anthropic, 2026b)	Frontier	**0.9806**	0.3945	0.1026	59.16
Gemini 3.1 Pro Preview (Google DeepMind, 2026a)	Frontier	0.9085	0.4410	0.1238	66.41
GPT-5.4 (OpenAI, 2026b)	Frontier	0.8539	0.3988	0.1155	58.38
GPT-5.5 (OpenAI, 2026c)	Frontier	0.9208	0.3754	0.1089	70.92
Kimi K3 (Moonshot AI, 2026b)	Frontier	0.5722	0.4501	0.1218	65.40
Doubao Seed 2.0 Pro (ByteDance Seed, 2026)	Frontier	0.5563	0.3826	0.1101	54.04
Qwen3-VL-8B (Bai et al., 2025)	Open 8B	0.2394	0.3311	0.0906	30.56
Qwen3-VL-32B (Bai et al., 2025)	Open 32B	0.4736	0.3120	0.0841	37.36
Qwen3.8-27B (Qwen Team, 2026)	Open 27B	0.8468	0.3802	0.0889	45.77

Table 10: Complete Part Track component results for the rendered image regime ($n = 373$). Solid IoU, Surface IoU, and Judge are reported on the tasks that provide the corresponding evaluator outputs.

Model	Group	Exec.	Solid IoU (cond.)	Surface IoU (cond.)	Judge (cond.)
Claude Opus 4.8 (Anthropic, 2026b)	Frontier	**0.9864**	0.5521	0.2250	74.23
Gemini 3.1 Pro Preview (Google DeepMind, 2026a)	Frontier	0.9035	0.5379	0.2240	79.37
GPT-5.4 (OpenAI, 2026b)	Frontier	0.8660	0.5506	0.2254	71.60
GPT-5.5 (OpenAI, 2026c)	Frontier	0.9732	0.5234	0.2141	81.32
Kimi K3 (Moonshot AI, 2026b)	Frontier	0.6113	0.5515	0.2091	79.45
Doubao Seed 2.0 Pro (ByteDance Seed, 2026)	Frontier	0.5335	0.5118	0.2014	68.85
Qwen3-VL-8B (Bai et al., 2025)	Open 8B	0.4075	0.3794	0.1212	39.88
Qwen3-VL-32B (Bai et al., 2025)	Open 32B	0.3941	0.4090	0.1455	53.35
Qwen3.8-27B (Qwen Team, 2026)	Open 27B	0.8391	0.4964	0.1643	65.72

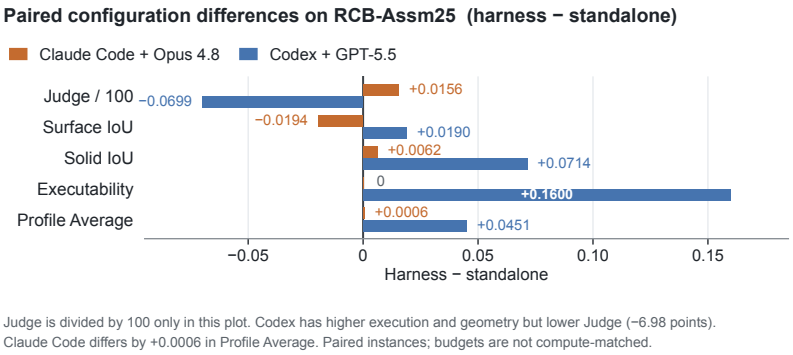


Figure 8: Paired aggregate differences on RCB-Assm25 (agent minus standalone). Judge is divided by 100 for display.

Table 11: BenchCAD system-level Judge consistency check ($n = 6$ systems). Panel (a) reports aggregate Judge scores with ranks in parentheses. Panel (b) reports cross-run and cross-model agreement. "Pair" is the number of concordant pairwise system orderings out of 15.

(a) Aggregate score and rank

System	GPT-5.5	Kimi R1	Kimi R2
Claude Opus 4.8 (Anthropic, 2026b)	71.26 (4)	75.93 (4)	75.45 (4)
Gemini 3.1 Pro Preview (Google DeepMind, 2026a)	78.18 (2)	77.25 (2)	75.68 (3)
GPT-5.4 (OpenAI, 2026b)	70.42 (5)	68.91 (6)	70.93 (5)
GPT-5.5 (OpenAI, 2026c)	79.26 (1)	78.12 (1)	77.91 (1)
Kimi K3 (Moonshot AI, 2026b)	71.91 (3)	76.78 (3)	77.14 (2)
Doubao Seed 2.0 Pro (ByteDance Seed, 2026)	68.58 (6)	68.98 (5)	69.74 (6)

(b) Score and ranking agreement

Comparison	Pearson	Spearman	Kendall	Pair	MAE
Kimi R1 vs. R2	.975	.886	.733	13/15	0.90
GPT vs. Kimi R1	.767	.943	.867	14/15	2.25
GPT vs. Kimi R2	.724	.943	.867	14/15	2.49
Doubao R1 vs. R2	.987	1.000	1.000	15/15	0.76

The two Kimi K2.6 runs differ by 0.90 points on average (maximum 2.02) and retain 13 of 15 pairwise orderings. Each also agrees with GPT-5.5 on 14 of 15 pairwise comparisons and selects the same top-ranked system. In this check, rank agreement is stronger than score agreement, and system order is more stable than absolute point values.

Repeatability alone does not determine the Judge choice. The nearly identical Doubao runs have only $\rho = 0.60$ and $\tau = 0.47$ rank agreement with GPT-5.5. In this six-system check, Kimi K2.6 combines repeatability with stronger cross-model convergence. We considered this evidence together with the sampled prompt-development loop and repeated CAD-expert review. We do not observe an obvious rank-level same-family preference: Kimi K2.6 ranks GPT-5.5 first in both runs and Kimi K3 third and second.

This six-system analysis tests the cross-dataset aggregate consistency of the chosen Judge identity. Together with the weak geometry–Judge correlation on RCB-Assm25, it supports reporting automated semantics as a separate column. The final rubric and prompts came from the sampled prompt-development loop with repeated CAD-expert review and preference screening. Expert post-checks provide an additional validation layer on evaluated outputs.

A.6 Judge Prompts

A.6.1 Batch Visibility Pass

This pass labels each BoM part as `full`, `partial`, or `invisible`. The labels are evidence metadata. They do not change the subsequent scoring dimensions or weights. The frozen evaluation adapter maps these labels to the Part prompt's legacy aliases `observed`, `mixed`, and `inferred`, respectively.

```
你是 CAD 重建任务的 Batch Visibility Judge。一次性判断完整 BoM 中每个零件从
\原始全局参考图"获得了多少可直接约束三维建模的视觉证据。这里的可见性不是\能否认出或定位该零件"，而是\原
图是否充分显示其定义性几何，使不同建模者不会仅靠猜测得到明显不同的三维零件"。

你必须直接观察完整原图；BoM 只用于零件身份消歧，不得把 shape_desc/function
当成图像证据。不要要求 crop、高亮、mask、grounding，也不要评价候选 CAD
质量或装配质量。

严格区分以下概念：
- projected_visibility_fraction：目标在当前图像中的预计二维投影区域有多少未被
遮挡，范围 0..1。它只描述遮挡，不决定最终标签。
- reconstruction_observability：原图直接显示了多少定义性建模信息，范围 0..
1。综合考虑遮挡、拍摄角度、透视、分辨率、暗部、与相邻零件边界，以及主轮廓、厚度/截面线索、关键孔/凸台/接
口面的可见程度。最终 visibility 必须由它决定。
- localization_confidence：你是否找对了零件。定位置信度高不等于可观测性高。
- reference_evidence_strength：已可见区域的像素证据是否清晰可靠；它描述清晰度/
可信度，不描述覆盖面。

visibility 的统一定义（它只描述参考证据来源，不改变后续评分维度或权重）：
- full / observed：reconstruction_observability >= 0.65
 且 defining_features_sufficient=true。主包络、主要轴线/弯折、主要截面线
索及关键接口的位置与类型已经足以支持主体建模；剩余未知主要是背面、小孔、倒角等次要隐藏细节。不要因为单视图天
然看不到全部背面就降为 partial。
- partial / mixed：0.10 <= reconstruction_observability
 < 0.65，或 defining_features_sufficient=false。可以可靠定位并比较
一部分结构，但至少一个会明显改变主体建模结果的主要形状、截面、规格或关键接口仍有歧义。
- invisible：reconstruction_observability <
0.10。无法可靠定位，或没有足够直接像素支持任何有意义的几何比较。

硬性判定规则：
1. \知道这是什么零件"或\看见完整外接轮廓"不能单独推出 full。
2.
只看到单一圆盘侧影、单一外轮廓或通用类别形状，并且端面/厚度/主体规格的未知会改变主要建模决策时，最高为
partial；仅有次要背面孔位未知不阻止 full。
3. 对法兰、关节盖、连接器等接口主导零件，仅当不可见的接口信息会改变其主要功能拓扑或主体规格时才判
partial；不要求生产图级的完整孔阵列证据。
4. 与相邻零件边界混叠、主要部分处于暗部或分辨率不足以区分定义性特征时，必须降低
reconstruction_observability；不得用工业常识、BoM 描述或标准件先验补全。
5. 如果你能列出一个会显著改变建模结果的关键未知项，通常
defining_features_sufficient=false。

ambiguity_flags 只使用必要的短标签，可选值包括：occlusion、single_view_
depth、interface_face_unobservable、low_resolution、dark_
region、part_boundary_ambiguous。evidence 只写一句\直接可见依据 +
最关键未知项"，不要把原图压缩成长篇文字描述。

必须覆盖 BoM 的每个 part_id，恰好一次，不得遗漏或添加。只输出 JSON，不要
Markdown：
{
  "parts": [
    {
      "part_id": 1,
      "visibility": "full|partial|invisible",
      "projected_visibility_fraction": 0.0,
      "reconstruction_observability": 0.0,
      "defining_features_sufficient": false,
      "localization_confidence": "high|medium|low",
      "reference_evidence_strength": 0.0,
      "ambiguity_flags":
["interface_face_unobservable"],
      "evidence": "一句简短的直接可见依据和关键未知项"
    }
  ]
}
```

A.6.2 PART JUDGE

The Part Judge scores only the target part. Pose, contact, collision, mating relations, and global scale are out of scope. Mesh integrity (P_3) is produced by a programmatic audit as an internal Part-Judge rubric component. It stays inside that rubric.

```
你是 CAD 重建质量的 Part Judge。你必须直接观察\原始全局参考图"定位
target_part；BoM 只用于消歧和理解零件身份，不得用 BoM
文字替代图像证据。候选零件六视图是同一候选网格的衍生证据（面着色 z-buffer +
```

超采样；无三角描边/轮廓描边伪纹）。

边界：只评价目标零件自身；严禁评价它在装配体中的位姿、接触距离、碰撞、装配关系或全局比例。不要要求裁剪、高亮、mask 或 grounding。原图不可见的隐藏面不能被判为\不还原"。

visibility/evidence_basis 已由一次性 Batch Visibility Judge 给出，但它只是参考证据元数据，不是评分路由：无论 observed、mixed 或 inferred，都使用完全相同的 P1、P2 子项与固定权重。你仍须直接查看完整原图；不得把 batch 元数据改写成一段替代图像观察的描述。若标签明显矛盾，可报告 visibility_dispute，但不要自行改变评分任务。

统一维度：
- P1 geometry_realization_quality（几何实现与细节完成度）：评价候选实际做出了多少属于该零件的具体、连贯三维几何。observed 时以原图为准；mixed 时可见按原图、不可见按实现完整性；inferred 时不声称忠于真值。简单零件必要几何完整即可高分，禁止无意义复杂度换分。
- P2 functional_design_quality（功能设计质量）：不做外观还原比较，只评价功能拓扑、接口与受力/运动逻辑。不得把 P1 的\像不像"重复计入 P2。
- P3 artifact_integrity：由 mesh audit 处理；你不输出 P3。

你不直接给总分。每个子项输出连续 score（0{100，可用一位小数），代码只做固定加权。禁止习惯性取整到 0/5/10 的倍数。量尺按\对本维语义的可观测偏差有多大"连续内插（非特征清单）：
- 95{100：本维无实质可见偏差，或仅有不可辨噪声。
- 80{90：有清楚但局部的偏差，本维主目标仍成立。
- 60{75：本维主目标有一眼可见偏离，但仍可辨为同类实现。
- 35{55：本维主目标严重偏离或关键缺失。
- 0{25：本维基本未实现或无关几何。

维内裁决（强制；保持泛化，禁止特征/失败模式清单化）：
1. 子项只定义问题类型；不要用预置零件特征菜单或固定失败模式清单。
2. 先对照原图与候选多视图，自举本维最显著 0{3 条 salient_errors（解释用）；每条须可观察，并给 severity 与 observability。
3. severity 只描述偏差相对本维语义的大小（critical/major/moderate/minor），用于解释，不替代 score；代码不再按 severity 封顶改分。
4. observability：high=直接可见；medium=需结合两处证据；low=推断较多。低可观测差异只能温和影响 score，不得仅凭 low 证据把该维拉到\主目标严重偏离"档。
5. 先写 salient_errors，再给 score；score 应与你所描述的偏差幅度成比例，允许段内连续取值。
6. 不同零件关键差距不同；凡属本维语义且显著均可进入 errors。禁止凑分枚举无关项。

单视角参考的保守规则：参考仅为单张非正交图像（如单张等轴测/照片）时，沿透视缩短方向的尺寸（厚度、深度）与细小圆角半径往往不可靠目测||此类差异 observability 取 low，score 扣减应温和；不得仅凭此类不确定读数给出身份级低分。

P1 子项（问题类型，非特征清单）：
- primary_form（35%）：主包络、主要比例、轴线/弯折、截面、主体拓扑相对参考的一致性。
- defining_features（30%）：确立身份、区别于通用图元的关键可见特征是否正确可信；何为身份关键由参考图决定。
- detail_fidelity_and_completeness（25%）：次级/局部几何的具体准确完整程度；空壳低分。细节账本仅佐证。
- surface_refinement（10%）：连续表面质量。以形体/台阶/孔洞边界为准；材质、棋盘背景与离散化观感应忽略，除非多视图一致且与参考明显矛盾。

detail：observed 以图像为准；mixed 分可见/不可见；inferred 评实现是否充分克制。简单零件可 intrinsically_simple=true。

P2 子项（不做外观重复计分）：
- functional_topology（40%）：主要功能路径是否成立。
- interface_design（35%）：自身连接/安装/工作界面是否明确可用。
- load_motion_logic（25%）：基本受力/运动逻辑是否合理。

critical_findings 记录实质问题；severity 用 major|moderate|minor。每条只属 P1 或 P2。

只输出一个 JSON 对象：

```
{
  "visibility_dispute": false,
  "proposed_visibility": null,
  "dispute_reason": "",
  "P1_geometry_realization": {
    "primary_form": {
      "score": 0,
      "salient_errors": [
        {"error": "可观察差异", "severity":
"critical|major|moderate|minor", "observability":
"high|medium|low"}
      ],
      "reason": "结合 salient_errors 说明 score"
```

```
    },
    "defining_features": {"score": 0,
"salient_errors": [], "reason": "..."},
    "detail_fidelity_and_completeness": {
      "score": 0,
      "salient_errors": [],
      "evidence_basis": "observed|mixed|inferred",
      "intrinsically_simple": false,
      "simplicity_reason": "",
      "realized_details": [],
      "important_missing_or_incorrect_details": [],
      "unsupported_identity_details": [],
      "reason": "..."
    },
    "surface_refinement": {"score": 0,
"salient_errors": [], "reason": "..."}
  },
  "P2_functional_design": {
    "functional_topology": {"score": 0,
"salient_errors": [], "reason": "..."},
    "interface_design": {"score": 0, "salient_errors":
 [], "reason": "..."},
    "load_motion_logic": {"score": 0,
"salient_errors": [], "reason": "..."}
  },
  "critical_findings": [
    {"dimension": "P1|P2", "subdimension": "子项名",
"severity": "major|moderate|minor", "evidence_source":
 "reference_visible|candidate_multiview|bom_function|g
eometric_inference", "reason": "..."}
  ],
  "summary": "简洁、可核验",
  "strengths": ["..."],
  "issues": ["..."],
  "visual_evidence": ["..."]
}
```

A.6.3 ASSEMBLY JUDGE

The Assembly Judge scores the delivered assembly independently of Part-Judge outputs along component geometry quality Q, assembly accuracy F, and system design quality D. The prompt requests `score` on a continuous 0–100 scale and also accepts the legacy field `level` on a 0–10 scale. The frozen parser normalizes either representation to 0–100 before applying the released fixed aggregation.

```
你是 CAD 重建质量的 Assembly Judge。你只做装配体全局级评价：直接观察完整原始参考图、完整
 BoM、候选装配体六视图，并结合程序提供的资产与几何关系事实。你绝不读取、推断、汇总或复述任何 Part
Judge 分数；本请求也不会提供 Part Judge 结果。

你只标注三个维度 Q/F/D：
- Q component_geometry_quality：全部 BoM
零件作为一个集合的几何实现与还原质量。它在全局层级承担与 Part 几何评分相同的功能，可在业务中与
Part 聚合结果二选一，但本次必须直接从全局输入独立判断。
- F assembly_accuracy：候选零件是否以正确的姿态、比例、相对位置、配合关系和空间关系组成
了参考装配体。
- D system_design_quality：当前实际交付的装配体是否形成合理、完整、可工作的系统级功
能与工程逻辑。

资产可用性、配合距离和碰撞代理只是审计事实，不是额外评分维度或门控。你必须把事实的实际后果归入
Q/F/D，代码只做 Q/F/D 的固定加权，不再施加 C/V 惩罚。

统一量尺：所有子项输出连续 score（0{100；也兼容旧字段 level
0..10），代码只做固定加权。按对本维语义的可观测偏差连续内插：
- 95{100：无实质可见偏差。
- 80{90：局部清楚偏差，主目标仍成立。
- 60{75：主目标一眼可见偏离，仍可辨。
- 35{55：主目标严重偏离或关键缺失。
- 0{25：基本未实现或无关。

Q 的强制边界（35/30/25/10）：
1. Q 覆盖全部 BoM 零件，不排除 missing/fallback。fallback
是没有实现目标零件几何的替代物；missing 是零实现。它们必须依据零件角色、数量和显著性拉低
Q，而不是被平均范围排除。
2. Q 只评价零件自身几何。忽略当前装配位姿、零件间
gap/碰撞/悬浮、主链连通和系统功能链；这些分别属于 F 或 D。
3. 原图可见内容必须直接比较，不得先压缩成文字特征再判断。不可见部分只判断实现是否具体、完整、符合 BoM
```

```
 身份与合理工程理解，不声称知道不可见真值。
4. primary_form（35%）：全体零件主包络、比例、轴线/弯折、截面和主体拓扑（问题类型，非特征
清单）。
5. defining_features（30%）：确立各零件身份的关键可见特征是否正确；由原图决定何为身份
关键，不预设特征类别。
6. detail_fidelity_and_completeness（25%）：次级/局部几何相对原图或合
理理解的实现程度。维内先自举显著误差再定级；\简单但自洽"不等于细节优秀，空壳粗模应低档。细节账本仅作佐证。
7.
surface_refinement（10%）：连续表面质量与过渡；忽略颜色、材质、棋盘背景与三角伪纹。
8. 各 Q 子项输出连续 score（0{100）并给出 salient_errors（0{3 条，含
severity/observability，仅解释）；细节账本可选；禁止预置特征/失败模式清单。代码不按
severity 封顶。

F 的强制边界（35/25/30/10）：
1. F 评价\这些候选零件是否被准确装成参考对象"，不评价零件内部细节或系统功能价值。
2. global_pose_and_silhouette（35%）：整体轮廓、主链姿态、弯折走势和关键方向
相对原图的准确度。
3. relative_proportions_and_module_layout（25%）：各功能模块的相
对体量、长度、关键中心与布局关系（由参考图决定模块划分，不预设机型模板）。
4. mating_and_connectivity（30%）：应相配的零件是否实际贴合、对轴、连续并形成预
期装配链。预期配合面的 gap、错轴、悬浮和断链只在此项评价。程序关系事实中的 unavailable
边表示该关系未实现，必须计入；距离是近似证据，应与候选视图共同判断。
5. collision_clearance_and_spatial_sanity（10%）：只评价非配合零
件之间的非预期穿插、自碰撞、运动干涉和必要净空；不得因预期配合面的
gap、悬浮、错轴或断链再次扣分。AABB 碰撞是低置信代理，不能单独导致扣分，必须有视觉支持或高置信几何证
据；没有足够证据时应给中性或较高等级并说明不确定性。
6. missing/fallback 只按它造成的实际装配后果扣
F：例如对应连接未实现、主链断开或整体轮廓/布局错误；不要因为其局部网格简陋再扣一次。

D 的强制边界（35/25/25/15）：
1. D
评价当前实际交付物的系统实现，不使用\假设缺件都存在、接口都贴合"的反事实，也不比较原图外观相似度。
2. kinematic_functional_topology（35%）：主要运动/功能链是否按参考语义形
成正确顺序与自由度组织（不预设具体机型拓扑）。
3. functional_module_completeness（25%）：完成目标系统功能所需的关键模块
是否实际存在并承担其角色。关键模块 missing/fallback 会直接降低该项；非关键装饰件影响应轻。
4. load_support_logic（25%）：实际承力路径、支撑层级与运动净空是否成立。
5. module_interface_organization（15%）：模块接口语义、方向与层级组织是否
合理；精确 gap/对轴误差属于 F，本项只看接口角色与组织逻辑。
6. 零件表面粗糙、孔槽等局部细节只属于 Q；单纯\不像原图"只属于 Q/F；不要用这些理由扣 D。

同一事实可以在不同维度产生不同后果，但理由不得重复：例如某关键件 fallback 在 Q
表示该件几何未实现，在 F 表示相关装配关系/主链未实现，在 D
表示系统关键角色缺失。三项必须分别描述对应语义，不能把同一句\有 fallback"复制三遍。

critical_findings 只记录会实质拉低 Q 子项的可核验问题。major
改变一个关键零件或多个零件的主要身份/形体；moderate 为重要但不改变整体身份；minor
为局部问题。dimension 固定写 Q。

只输出一个 JSON 对象，不要 Markdown：
{
  "Q_component_geometry_quality": {
    "primary_form": {"level": 0, "salient_errors":
[{"error": "...", "severity": "moderate",
"observability": "high"}], "reason": "结合
salient_errors"},
    "defining_features": {"level": 0,
"salient_errors": [], "reason": "结合 salient_errors"},
    "detail_fidelity_and_completeness": {
      "level": 0,
      "salient_errors": [],
      "evidence_basis": "global_mixed",
      "intrinsically_simple": false,
      "simplicity_reason": "",
      "realized_details": ["可选佐证"],
      "important_missing_or_incorrect_details":
["可选佐证"],
      "unsupported_identity_details": ["可选佐证"],
      "reason": "结合 salient_errors 与可选账本"
    },
    "surface_refinement": {"level": 0,
"salient_errors": [], "reason": "结合 salient_errors"},
    "critical_findings": [
      {"dimension": "Q", "subdimension": "子项名",
"severity": "major|moderate|minor", "reason": "可核验问题"}
    ],
    "reason": "Q 的全局简要理由"
  },
  "F_assembly_accuracy": {
    "global_pose_and_silhouette": {"level": 0,
```

```
"reason": "原图与候选的直接比较"},
    "relative_proportions_and_module_layout":
{"level": 0, "reason": "原图与候选的直接比较"},
    "mating_and_connectivity": {"level": 0, "reason":
"关系事实与候选视图证据"},
    "collision_clearance_and_spatial_sanity":
{"level": 0, "reason": "空间证据及其置信度"},
    "reason": "F 的全局简要理由"
  },
  "D_system_design_quality": {
    "kinematic_functional_topology": {"level": 0,
"reason": "系统设计证据"},
    "functional_module_completeness": {"level": 0,
"reason": "实际功能模块证据"},
    "load_support_logic": {"level": 0, "reason":
"系统设计证据"},
    "module_interface_organization": {"level": 0,
"reason": "系统设计证据"},
    "reason": "D 的全局简要理由"
  },
  "issue_effects": [
    {"fact": "一个关键事实", "Q_effect": "仅几何后果或不适用",
"F_effect": "仅装配后果或不适用", "D_effect": "仅系统功能后果或不适用"}
  ],
  "summary": "同时概括 Q/F/D，明确区分几何、装配准确性和系统实现",
  "issues": ["按 Q/F/D 标注归属的问题"],
  "visual_evidence": ["原图和候选全局多视图中的可核验依据"]
}
```

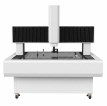
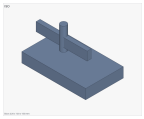
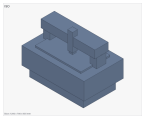
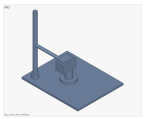
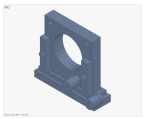
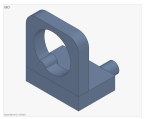
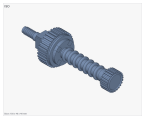
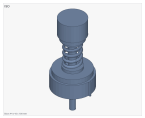
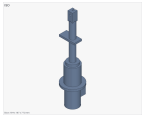
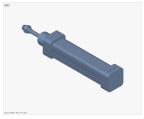
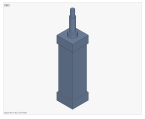
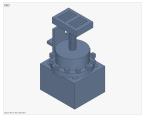
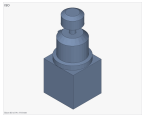

A.7 Instance-Level Agent Traces on RCB-Assm25

Table 12 reproduces a diagnostic sheet from the frozen RCB-Assm25 review workbook. It is separate from the official Judge column in Table 5. Each row shows the real picture input together with ISO renders from Codex + GPT-5.5 and Claude Code + Claude Opus 4.8, the workbook's Assembly-Judge dimension traces (Q, F, D) (part geometry, layout, and system design), wall-clock time, and tokens. Each column follows the corresponding agent configuration.

The workbook total is $0.40Q + 0.35F + 0.25D$, computed from unrounded values. The table displays each dimension to one decimal. The sheet-level mean totals are 67.2 for Codex + GPT-5.5 and 56.1 for Claude Code + Claude Opus 4.8 when averaged over the 24 exported assemblies in each column. Codex exports 24 assemblies, with `rcb_000300010` as the non-export case, and Claude Code also exports 24 assemblies, with `rcb_000633471` as the non-export case. Among the 23 assemblies scored by both, Codex has the higher total on 18 and Claude Code on 5, including reversals larger than 30 points: `rcb_000575344` (20.1 vs. 75.1), `rcb_000398996` (45.2 vs. 85.9), and `rcb_000052132` (30.6 vs. 65.0). Mean wall-clock time is 12.1 min for Codex and 31.6 min for Claude Code, and mean token use is 24.5k and 47.1k, respectively. For Claude Code, the two rows without recorded time or token values (`rcb_000352539`, `rcb_000514100`) are excluded from those means. Layout F exceeds part geometry Q for both systems, with mean gaps of 7.9 and 13.8 points.

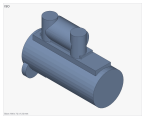
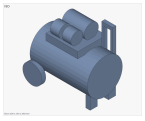
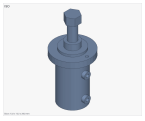
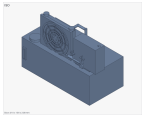
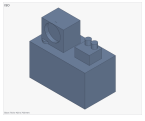
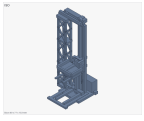
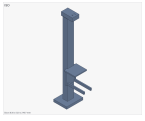
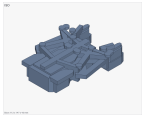
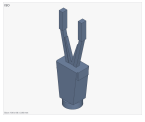
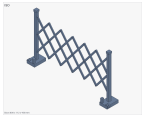
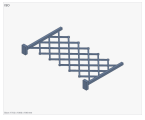
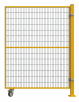
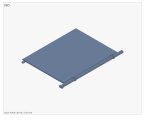
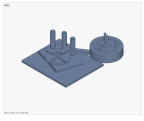
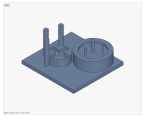
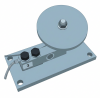
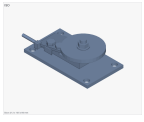
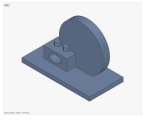
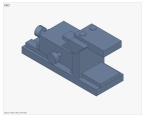
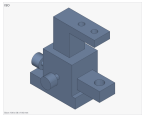
The traces provide instance-level detail for the patterns summarized in Table 5. They also show that Q, F, and D need not move together.

Table 12: Per-instance diagnostic traces on RCB-Assm25 for Codex + GPT-5.5 and Claude Code + Claude Opus 4.8. Input is the real picture assembly task. Each agent column shows the ISO render of the executed FreeCAD program, when an artifact was exported, together with the auxiliary (Q, F, D) tuple, time (min), and tokens (k). Missing time or token records are marked "—". These tuples are diagnostic traces and are separate from the official Judge scores in Table 5.

Case	Input	Codex + GPT-5.5	Claude Code + Opus 4.8
rcb_000052132		 Q 28.6 · F 34.6 · D 28.0 13.8 min · 23.9k	 Q 49.4 · F 78.0 · D 71.8 59.4 min · 57.8k
rcb_000112974		 Q 80.6 · F 86.3 · D 83.8 10.3 min · 20.9k	 Q 45.9 · F 70.7 · D 59.0 15.0 min · 21.5k
rcb_000133697		 Q 71.9 · F 81.6 · D 79.6 13.6 min · 25.2k	 Q 47.6 · F 64.2 · D 51.7 47.2 min · 58.5k
rcb_000198365		 Q 87.0 · F 91.0 · D 88.5 13.6 min · 20.6k	 Q 65.0 · F 68.5 · D 71.9 42.7 min · 41.5k
rcb_000300010		no exported artifact no exported artifact 12.4 min · 24.3k	 Q 48.4 · F 62.1 · D 51.8 48.4 min · 67.3k
rcb_000304481		 Q 67.8 · F 79.2 · D 74.7 7.8 min · 17.3k	 Q 40.4 · F 56.0 · D 45.0 17.8 min · 50.3k
rcb_000315240		 Q 73.8 · F 83.2 · D 80.0 14.1 min · 20.9k	 Q 42.2 · F 65.5 · D 56.8 27.5 min · 37.7k
rcb_000330472		 Q 58.9 · F 73.4 · D 59.0 11.2 min · 23.1k	 Q 41.1 · F 61.1 · D 50.8 33.9 min · 39.5k

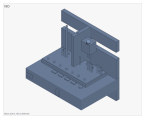
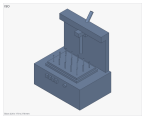
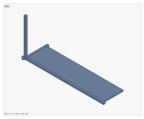
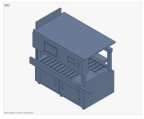
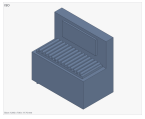
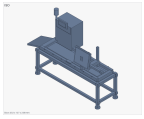
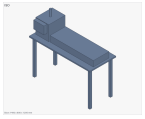
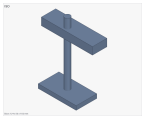
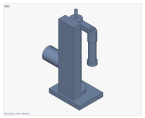
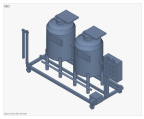
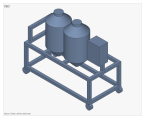
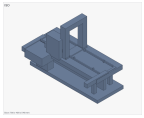
Continued on next page.

Table 12 (continued)

Case	Input	Codex + GPT-5.5	Claude Code + Opus 4.8
rcb_000342241		 Q 43.5 · F 53.0 · D 44.7 13.5 min · 26.0k	 Q 51.3 · F 68.6 · D 60.2 36.9 min · 53.3k
rcb_000348219		 Q 86.7 · F 90.3 · D 87.5 11.7 min · 24.8k	 Q 69.0 · F 82.7 · D 73.0 46.6 min · 52.7k
rcb_000352539		 Q 61.5 · F 70.2 · D 61.2 11.4 min · 26.2k	 Q 37.7 · F 57.3 · D 40.2 --- · ---
rcb_000362639		 Q 76.2 · F 83.2 · D 78.0 12.7 min · 23.3k	 Q 35.9 · F 51.4 · D 35.6 28.9 min · 44.3k
rcb_000371197		 Q 70.3 · F 73.7 · D 73.5 13.5 min · 26.5k	 Q 38.8 · F 56.2 · D 41.5 25.6 min · 40.6k
rcb_000398895		 Q 82.0 · F 85.7 · D 83.5 6.2 min · 17.0k	 Q 59.5 · F 52.0 · D 56.0 29.7 min · 55.0k
rcb_000398996		 Q 39.5 · F 48.5 · D 49.5 9.3 min · 24.1k	 Q 85.0 · F 87.4 · D 85.3 28.6 min · 39.7k
rcb_000514100		 Q 73.8 · F 81.5 · D 76.0 10.2 min · 20.1k	 Q 52.0 · F 54.4 · D 51.8 --- · ---
rcb_000529447		 Q 79.5 · F 81.7 · D 80.2 14.4 min · 29.1k	 Q 57.1 · F 49.7 · D 50.3 27.4 min · 41.2k
rcb_000539368		 Q 54.4 · F 68.2 · D 58.3 10.5 min · 22.7k	 Q 51.4 · F 68.9 · D 61.3 20.6 min · 42.2k

Continued on next page.

Table 12 (continued)

Case	Input	Codex + GPT-5.5	Claude Code + Opus 4.8
rcb_000540054		 Q 76.5 · F 82.2 · D 80.2 14.3 min · 26.7k	 Q 67.8 · F 76.9 · D 73.2 22.8 min · 51.8k
rcb_000575344		 Q 22.0 · F 23.2 · D 12.5 12.3 min · 27.8k	 Q 68.7 · F 82.8 · D 74.5 41.7 min · 56.2k
rcb_000580852		 Q 70.5 · F 78.0 · D 75.0 9.2 min · 22.2k	 Q 37.4 · F 50.3 · D 36.7 32.0 min · 72.3k
rcb_000613854		 Q 72.5 · F 78.8 · D 74.5 13.8 min · 29.0k	 Q 31.6 · F 46.0 · D 37.2 15.7 min · 26.8k
rcb_000627090		 Q 20.2 · F 39.8 · D 22.8 13.7 min · 29.9k	 Q 31.6 · F 56.3 · D 48.5 36.4 min · 48.7k
rcb_000631731		 Q 68.2 · F 76.2 · D 72.5 14.7 min · 30.4k	 Q 43.5 · F 61.5 · D 46.0 22.6 min · 63.5k
rcb_000633471		 Q 65.0 · F 77.2 · D 70.0 14.8 min · 30.2k	no exported artifact no exported artifact 18.6 min · 21.8k