Thinking in Pictures: A Diagnostic Study of Visual vs. Textual Chain-of-Thought Reasoning in Vision-Language Models

Ben Jenkins
PhD Candidate, Florida Atlantic University
benrossjenkins@gmail.com

Abstract

Chain-of-thought (CoT) reasoning has become a standard technique for eliciting complex reasoning in large language models, and recent work has extended it to vision-language models (VLMs). However, virtually all multimodal CoT methods generate intermediate reasoning steps in natural language, even for inherently visual problems such as spatial reasoning, geometric manipulation, and object tracking. We ask a focused question: *when does textual reasoning help or hurt a VLM, and does generating visual artifacts help because of the code or because of the rendered image?* We present VISCOT-DIAG, a diagnostic benchmark of 1,200 instances across five visual reasoning categories, and compare five CoT paradigms across four VLMs. Our results characterize a consistent modality gap: textual CoT degrades spatial transformation by up to 16.5% and multi-object tracking by 12.7%, while visual CoT yields gains of up to 25.4%. A code-only ablation (V-CoT-NF) recovers only 36% of the V-CoT gain over T-CoT on spatial transformation, indicating that the rendered image, not code generation alone, drives most of the improvement. We identify three failure modes (spatial state collapse, transformation hallucination, tracking loss) and show that adaptive modality routing achieves 78.2% accuracy versus 74.0% for V-CoT-everywhere on this benchmark. We discuss the scope of these conclusions and recommend practitioners use visual CoT for spatial-transformation-heavy tasks and textual CoT for compositional counting.

1 Introduction

Chain-of-thought (CoT) prompting has dramatically improved the reasoning capabilities of large language models by encouraging models to decompose problems into intermediate steps before arriving at an answer (Wei et al., 2022; Kojima et al., 2022). This paradigm has been extended to vision-language models (VLMs), where multimodal CoT methods generate textual rationales that incorporate information from both visual and linguistic inputs (Zhang et al., 2024; Lu et al., 2022).

Recent benchmarks have already documented that direct answering and naive textual CoT underperform on multimodal tasks requiring expert knowledge or fine-grained visual grounding (Zhao et al., 2025; Yue et al., 2025). A growing body of work has advanced multimodal CoT through structured reasoning stages (Xu et al., 2025; Thawakar et al., 2025), visual tool use (Hu et al., 2024; Wang et al., 2025), and visualization of reasoning traces (Wu et al., 2024). What is less studied, and what we focus on here, is a more controlled question: *within visual reasoning, where exactly does textual CoT help or hurt, and when visual CoT helps, is the gain coming from code-based symbolic reasoning or from the rendered image itself?*

Consider the motivating example in Figure 1. A model is shown a scene with colored shapes and asked: *"After rotating the red triangle 90° clockwise and reflecting the blue square horizontally, is the triangle above or below the square?"* Under textual CoT, the model must verbally describe each transformation and its spatial consequence, a process that is error-prone because language lacks the precision to faithfully encode continuous spatial configurations. Under visual CoT, the model can draw the transformed objects directly, producing a visual representation that grounds subsequent reasoning. A natural objection is that V-CoT could be "cheating" by delegating geometric computation to a Python interpreter; we address this directly in §5 via a code-only ablation.

This phenomenon is consistent with cognitive-science findings on mental rotation (Shepard and Metzler, 1971), dual coding theory (Paivio, 1991), and mental imagery (Kosslyn, 1995), though we caution that VLM behavior need not mirror human cognition.

We make three contributions. First, we introduce

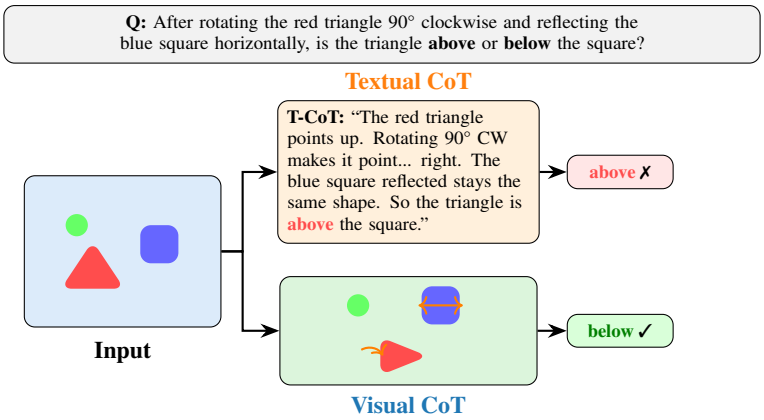

Figure 1: Motivating example. Given a stated spatial-transformation question (top), textual CoT produces a plausible but incorrect verbal trace (the model hallucinates the post-rotation spatial relationship), while visual CoT draws the actual transformations and grounds the answer in the rendered image.

VISCOT-DIAG, a diagnostic benchmark of 1,200 instances spanning five visual reasoning categories with fine-grained difficulty calibration (§3). Second, we conduct a controlled comparison of *five* CoT paradigms (direct, textual, structured textual, visual, and visual-no-feedback) across four VLMs, isolating not only the effect of reasoning modality but also whether visual gains stem from code generation or from observing the rendered image (§4–§5). Third, we document three failure modes of textual CoT in visual domains and provide empirical guidance for when each modality helps (§6).

We emphasize the scope of our claims: VISCOT-DIAG comprises relatively simple, controlled visual tasks. Our findings should be read as a diagnostic characterization on this controlled distribution rather than a universal claim about visual reasoning at large; we discuss this scope explicitly in §8.

2 Related Work

Textual Chain-of-Thought. Wei et al. (2022) demonstrated that prompting LLMs to produce intermediate reasoning steps improves performance on arithmetic, commonsense, and symbolic reasoning. Zero-shot variants achieve similar effects without exemplars (Kojima et al., 2022). These methods operate exclusively in natural language.

Multimodal Chain-of-Thought. Zhang et al. (2024) proposed a two-stage framework separating rationale generation from answer inference. Lu et al. (2022) showed that generating explanations as CoT chains improves science QA. LLaVA-CoT (Xu et al., 2025) structures reasoning into four textual stages and achieves strong results through supervised fine-tuning. LlamaV-o1 (Thawakar et al., 2025) introduces step-level evaluation. Compositional CoT (Mitra et al., 2024) uses scene graphs as intermediate textual representations. The common thread is that intermediate reasoning is *entirely textual*.

Multimodal Benchmarks Highlighting CoT Limits. Several recent benchmarks have already shown that direct answering and naive textual CoT struggle on knowledge-intensive or fine-grained multimodal tasks. MMMU (Yue et al., 2024), MMMU-Pro (Yue et al., 2025), MMStar (Chen et al., 2024), and MMVU (Zhao et al., 2025) all report large headroom for multimodal models, and MMMU-Pro in particular documents brittleness of textual reasoning when visual content is presented in non-canonical formats. Our work is complementary: rather than expanding the difficulty of multimodal benchmarks, we hold task complexity controlled and ablate the reasoning *paradigm*, asking when textual reasoning specifically helps or hurts.

Visual Intermediate Representations. Visualization-of-Thought (VoT) (Wu et al., 2024) prompts LLMs to generate ASCII visualizations during spatial reasoning. Visual Sketchpad (Hu et al., 2024) enables VLMs to draw via code execution as part of an agentic reasoning loop, achieving gains on math and vision tasks. VisuoThink (Wang et al., 2025) combines visual-textual interleaving with tree search. The Visual CoT dataset (Shao et al., 2024) contributes bounding-box annotations as intermediate grounding. Each of these demonstrates the

value of visual reasoning in specific settings; we provide a controlled paradigm comparison and, importantly, a code-only ablation that disentangles the contributions of code generation from visual perception of the rendered output.

3 VISCOT-DIAG Benchmark

To systematically evaluate the effect of reasoning modality, we construct VISCOT-DIAG, comprising 1,200 problem instances organized into five categories based on the dominant type of visual reasoning required. Figure 2 provides an overview.

3.1 Reasoning Categories

Spatial Relation (SR). Tasks requiring judgment about relative positions, orientations, and arrangements of objects. We source from BLINK spatial (Fu et al., 2024) and CLEVR (Johnson et al., 2017), filtering for unambiguous ground truth.

Spatial Transformation (ST). Tasks requiring mental manipulation of visual elements: rotation, reflection, translation, and compositions thereof. We construct synthetic instances using procedurally generated 2D polygons (triangles, quadrilaterals, pentagons, L-shapes) with controlled transformation parameters.

Multi-Object Tracking (MOT). Tasks requiring the model to track positions or states of multiple objects through a sequence of changes described in natural language. Generated programmatically using grid-world environments with 3 to 7 colored shapes and 2 to 6 sequential movement instructions.

Compositional Counting (CC). Tasks requiring identification, filtering, and counting of objects based on conjunctions of visual attributes. Adapted from CLEVR (Johnson et al., 2017) and GQA (Hudson and Manning, 2019).

Geometric Reasoning (GR). Tasks involving properties of geometric figures: computing angles, identifying congruent shapes, reasoning about area or perimeter, and applying geometric theorems. Drawn from MathVista (Lu et al., 2024) geometry subsets plus synthetic instances.

3.2 Difficulty Calibration

Within each category, we calibrate difficulty along category-specific axes. **Easy**: 1–2 reasoning steps or 2–3 objects; **Medium**: 3–4 steps or 4–5 objects; **Hard**: 5+ steps or 6–7 objects. We validate difficulty through pilot experiments with three human annotators, confirming monotonically decreasing accuracy across levels (Easy: 94.2%, Medium: 81.7%, Hard: 63.4%).

4 Experimental Setup

4.1 CoT Paradigms

We compare *five* reasoning paradigms applied at inference time via prompting, requiring no fine-tuning. The fifth paradigm, V-CoT-NF, isolates whether V-CoT gains stem from code generation or from visual perception of the rendered output. Figure 3 illustrates the paradigms.

Direct (D). The model receives image and question, produces an answer.

Textual CoT (T-CoT). The model is prompted to "think step by step" (Wei et al., 2022; Kojima et al., 2022).

Structured Textual CoT (ST-CoT). The model produces reasoning in four stages (summary, observation, reasoning, answer), mirroring Xu et al. (2025) as a prompting strategy rather than a fine-tuning objective.

Visual CoT, no feedback (V-CoT-NF). The model is prompted exactly as in V-CoT to generate Python annotation code. The code is executed silently and the resulting image is *not* returned. The model must commit to an answer based on its mental simulation of what the code would have produced. If V-CoT-NF matches V-CoT, then code generation alone explains V-CoT gains; if it matches T-CoT, then visual perception of the rendered image is essential.

Visual CoT (V-CoT). The model alternates between Act (generate annotation code) and Observe (the rendered image is re-encoded and shown back to the model), with up to five turns. Annotation primitives include shapes, lines, arrows, text, and affine transformations (rotation, reflection, translation), following Hu et al. (2024).

4.2 Models

We evaluate four VLMs spanning closed-source frontier and open-weight large-scale: **GPT-5** (OpenAI, 2025), **Claude Opus 4.6** (Anthropic, 2026), **Gemini 2.5 Pro** (Gemini Team, Google DeepMind, 2025), and **Qwen3-VL-72B** (Bai et al., 2025). All use greedy decoding (temperature 0) for reproducibility.

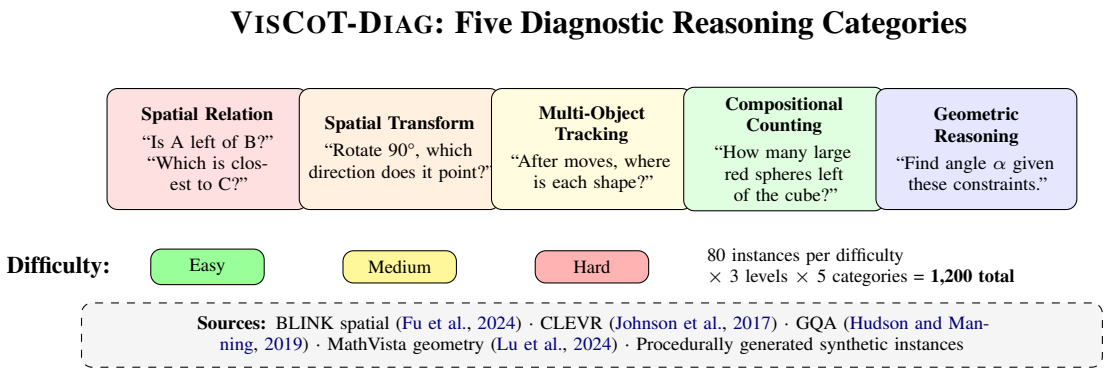

Figure 2: Overview of the VISCOT-DIAG benchmark. Five categories target distinct visual reasoning abilities, with three difficulty levels per category. Instances are sourced from existing benchmarks and procedural generation.

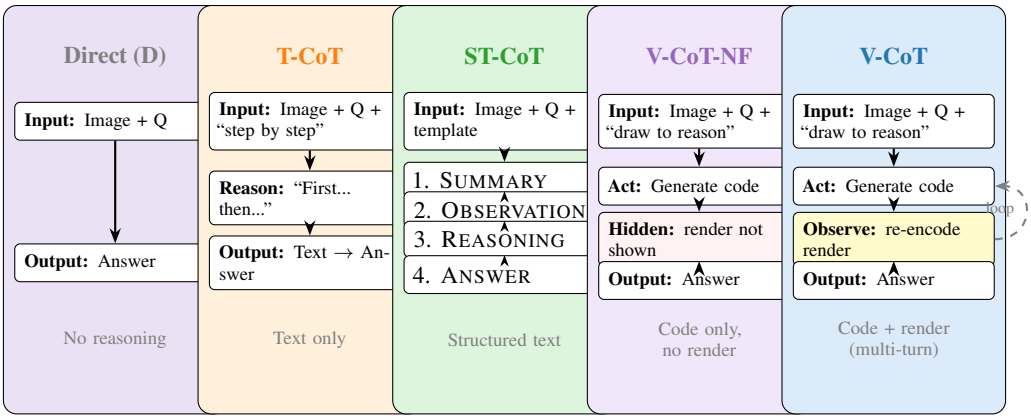

Figure 3: The five CoT paradigms compared in our study. **Direct** produces an answer in one pass. **T-CoT** generates a free-form reasoning chain. **ST-CoT** follows a four-stage template mirroring Xu et al. (2025). **V-CoT-NF** generates the same drawing code as V-CoT but the rendered image is *not* returned to the model, isolating the contribution of code generation. **V-CoT** alternates between code generation and observing the updated image, following Hu et al. (2024).

4.3 Evaluation Metrics

All tasks use **exact-match accuracy** against ground truth. For free-form numeric answers (GR), we allow $\pm 1\%$ tolerance. We additionally report **premise-conclusion consistency (PCC)**, the fraction of instances where the model's final answer is logically entailed by its own reasoning steps, assessed by a Claude Opus 4.6 judge (we verified 92% agreement with a Gemini 2.5 Pro judge on a 100-instance subset). **Annotation quality (AQ)** for V-CoT and V-CoT-NF measures whether generated visualizations are semantically meaningful (assessed on the executed render even when not shown to the model).

5 Results

5.1 Main Results

Table 1 reports accuracy across all five categories and five paradigms, averaged over the four VLMs. The V-CoT-NF column lets us decompose the V-CoT gain into a code-generation contribution and a visual-feedback contribution.

Three observations emerge. First, T-CoT degrades performance on ST (-16.5) and MOT (-12.7). Notably, T-CoT's category-level wins on CC and GR (+16.8 and +13.8) are almost exactly offset by its ST and MOT losses: on average across visual reasoning, T-CoT provides essentially no benefit over direct answering (57.3 vs. 56.1). Second, V-CoT provides consistent gains across all categories except CC (+12 to +25). Third, V-CoT-NF produces partial gains: on ST it recovers from -16.5 (T-CoT) to -1.4 (essentially matching Direct), and on GR it captures most of the V-CoT improvement. The remaining gap between V-CoT-NF and V-CoT is largest on ST (26.5 points) and MOT (19.4), the categories where tracking visual state matters most.

Category	Direct	T-CoT	ST-CoT	V-CoT-NF	V-CoT	Best Δ vs. D
Spatial Relation (SR)	66.1	70.4 +4.3	72.0 +5.9	69.0 +2.9	**78.2** +12.1	+12.1 (V-CoT)
Spatial Transformation (ST)	51.7	35.2 -16.5	38.1 -13.6	50.3 -1.4	**76.8** +25.1	+25.1 (V-CoT)
Multi-Object Tracking (MOT)	59.0	46.3 -12.7	50.3 -8.7	53.6 -5.4	**73.0** +14.0	+14.0 (V-CoT)
Compositional Counting (CC)	55.9	72.7 +16.8	**74.7** +18.8	68.8 +12.9	68.4 +12.5	+18.8 (ST-CoT)
Geometric Reasoning (GR)	48.0	61.8 +13.8	63.8 +15.8	64.0 +16.0	**73.4** +25.4	+25.4 (V-CoT)
Average	56.1	57.3	59.8	61.1	**74.0**	+17.8 (V-CoT)

Table 1: Accuracy (%) averaged across four VLMs on VISCOT-DIAG. Colors indicate change relative to Direct (green: improve; red: decline). Textual CoT degrades performance on ST and MOT; Visual CoT provides consistent gains. V-CoT-NF, which executes the same drawing code as V-CoT but hides the render from the model, sits between T-CoT and V-CoT on spatial tasks, though on CC it underperforms T-CoT. Best result per category is **bolded**.

5.2 Decoupling Code Generation from Visual Feedback

The V-CoT-NF ablation directly addresses the concern that V-CoT might be "cheating" by delegating geometric computation to a Python interpreter. If so, V-CoT-NF (which runs the same code) should match V-CoT, since the interpreter performs identical computation either way. The data does not support that interpretation. On ST, V-CoT-NF reaches 50.3% versus 76.8% for V-CoT and 35.2% for T-CoT, recovering only 36% of the V-CoT gain over T-CoT. On MOT, V-CoT-NF recovers 27% of the gain.

We interpret this as evidence that the rendered image carries information the model cannot reconstruct from the code alone. Generating drawing code does provide a measurable benefit on its own, likely because writing code to draw forces the model to commit to precise spatial coordinates, but the act of *seeing* the rendered configuration provides additional accuracy that code-based simulation cannot fully replace. This is consistent with the rendered image functioning as an external visuospatial scratchpad against which the model can verify or correct its mental simulation.

The pattern also clarifies where code generation is most useful in isolation. On GR, V-CoT-NF achieves +16.0 over Direct, recovering 63% of V-CoT's gain. Geometric reasoning involves angle and length computations that benefit from explicit symbolic structure (declaring variables, applying formulas), and much of this benefit transfers without visual feedback. On ST, by contrast, the symbolic structure of the code (an `img.rotate(-90)` call) does not by itself disambiguate the spatial outcome; the model needs to see the result. On CC, V-CoT-NF (68.8%) actually *underperforms* T-CoT (72.7%) by 3.9 points: when the task rewards arithmetic and attribute filtering, tokens spent producing drawing code that the model cannot see appear to displace tokens that would otherwise be used for symbolic decomposition.

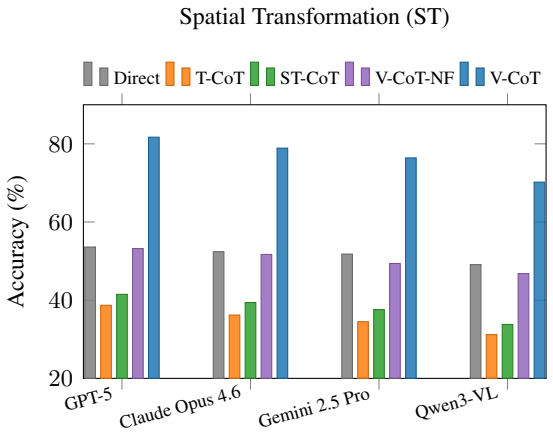

Figure 4: Per-model accuracy on Spatial Transformation. T-CoT (orange) decreases accuracy below Direct (gray) for every model. V-CoT-NF (purple) approaches Direct-level performance; only V-CoT with the rendered image (blue) produces the large gains. The pattern is consistent across all four VLMs.

5.3 Per-Model Results

Figure 4 disaggregates ST results by model across all five paradigms. The pattern is consistent across all four VLMs.

GPT-5 achieves the largest V-CoT gains on ST (+43.0 over T-CoT), likely because its code generation produces higher-quality annotations. The V-CoT vs. V-CoT-NF gap is also model-dependent: 28.5 for GPT-5, 27.2 for Claude Opus 4.6, 27.0 for Gemini 2.5 Pro, and 23.4 for Qwen3-VL. Models that benefit more from rendered feedback may also have stronger visual encoders, though we do not measure encoder quality directly.

Difficulty	T-CoT	V-CoT-NF	V-CoT	Δ (V−T)
Spatial Transformation				
Easy	54.8	65.7	75.2	+20.4
Medium	33.1	52.0	82.2	+49.1
Hard	17.7	33.2	73.0	+55.3
Compositional Counting				
Easy	84.9	80.9	72.8	−12.1
Medium	71.7	69.1	68.0	−3.7
Hard	61.4	56.4	64.5	+3.1

Table 2: Accuracy (%) by difficulty level (averaged across models). On ST, the V-CoT-NF row shows that code generation alone is competitive at Easy difficulty but collapses on Hard chains (33.2% versus 73.0% for V-CoT). Visual feedback is what makes V-CoT robust to chain length.

Adaptive Routing. A DistilBERT classifier that selects T-CoT vs. V-CoT per question achieves 78.2% average accuracy versus 74.0% for V-CoT-everywhere and 57.3% for T-CoT-everywhere on this benchmark (84.7% routing accuracy; see Appendix D). Question-level modality routing is feasible and beneficial.

5.4 Scaling with Difficulty

Table 2 shows how the modality gap interacts with task difficulty. On ST, the V-CoT vs. T-CoT gap widens from 20.4 points at Easy to 55.3 points at Hard. T-CoT errors compound with each additional transformation. Notably, at Hard difficulty T-CoT accuracy on ST falls to 17.7%, well below the 50% chance baseline for binary spatial questions: T-CoT is not merely failing to help but is producing systematically wrong answers. We attribute this to a salience-anchoring bias under uncertainty: when the model cannot resolve a transformation's outcome verbally, it tends to default to the most prominent spatial relation described earlier in the chain rather than tracking the cumulative state. The V-CoT-NF row shows that code-only reasoning also degrades at Hard difficulty (from 65.7 at Easy to 33.2 at Hard), indicating that the code-based component of V-CoT is not robust to long transformation chains; visual feedback is what stabilizes performance.

5.5 Annotation Quality

For V-CoT, annotation quality (AQ) varies across models: GPT-5 produces semantically correct annotations 89.4% of the time, followed by Claude Opus 4.6 (85.7%), Gemini 2.5 Pro (83.6%), and Qwen3-VL-72B (78.1%). Conditioning on AQ: V-CoT accuracy on instances with correct annotations is 79.6%, compared to 42.8% on instances with incorrect annotations. V-CoT-NF accuracy under correct annotations is 64.3% versus 53.7% under incorrect annotations. The gap between V-CoT and V-CoT-NF is largest precisely when annotations are correct (15.3 points), confirming that the model is reading correct visual content from the rendered image, not just benefiting from the structuring effect of code.

6 Analysis: Failure Modes of Textual CoT

Through qualitative analysis of 200 error cases (stratified by category and model, where T-CoT underperforms), we identify three recurring failure modes. Two annotators achieved 89% agreement on failure mode labels (Fleiss' $\kappa = 0.82$). Figure 5 illustrates each mode.

6.1 Spatial State Collapse

When reasoning about spatial relationships across multiple steps, textual representations progressively lose spatial fidelity. The model may correctly describe pairwise relationships in isolation but fail to maintain a globally consistent spatial configuration. PCC on SR tasks degrades sharply with object count for T-CoT (91.2% with 2 objects to 51.3% with 5+) while V-CoT remains stable (94.8% to 84.7%); see Figure 6. Notably, V-CoT-NF tracks T-CoT closely on this metric (89.7% to 56.4%), indicating that the rendered visual layout, not the act of writing code, is what stabilizes spatial consistency.

6.2 Transformation Hallucination

Asked to predict the result of spatial transformations, T-CoT frequently produces plausible-sounding but incorrect descriptions. On multi-step transformation chains,[1] T-CoT accuracy is 18.3% while V-CoT maintains 61.7%. Even single-step transformations suffer: T-CoT 44.1% versus V-CoT 76.2%. V-CoT-NF on the same chains reaches 38.5%, confirming that the rendered output contributes substantively beyond what code generation alone provides.

[1] Multi-step here means transformation chains of length ≥ 2, partially overlapping but not identical to the Hard difficulty stratum in Table 2.

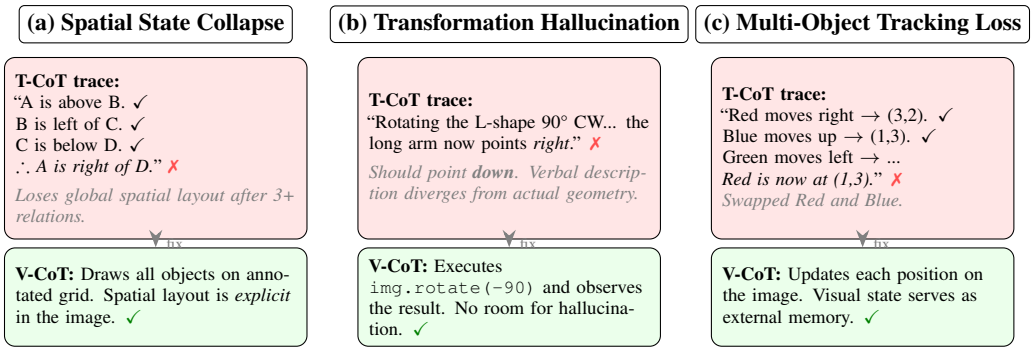

Figure 5: Three failure modes of textual CoT on visual reasoning tasks, and how visual CoT resolves each. **(a)** Spatial State Collapse. **(b)** Transformation Hallucination. **(c)** Tracking Loss.

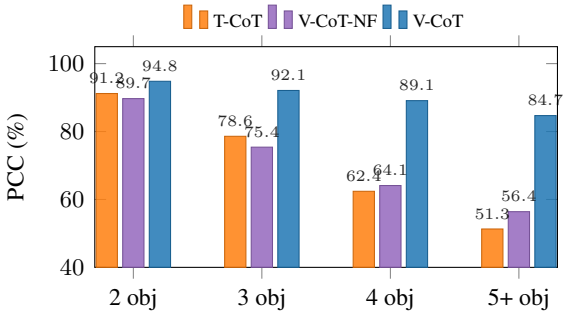

Figure 6: Premise-conclusion consistency on Spatial Relation tasks by object count. T-CoT and V-CoT-NF degrade similarly; only V-CoT (with rendered feedback) maintains consistency.

Category	State Collapse	Trans. Halluc.	Track. Loss
SR	48.3%	12.1%	6.9%
ST	15.7%	61.4%	8.6%
MOT	22.8%	5.3%	57.9%
CC	8.2%	3.1%	2.7%
GR	18.6%	38.4%	4.3%

Table 3: Failure mode frequency in T-CoT errors. Each cell shows the share of errors attributed to each mode; rows need not sum to 100%.

6.3 Multi-Object Tracking Loss

When tracking multiple objects through sequential state changes, T-CoT must maintain a verbal ledger. We observe two sub-patterns on 5-object instances: *identity confusion* (T-CoT 34.7% vs. V-CoT 8.2%) and *update omission* (T-CoT 22.1% vs. V-CoT 5.4%). V-CoT-NF rates fall between (identity confusion 26.3%, update omission 17.8%): code structure helps somewhat but cannot replace external visual memory.

6.4 Failure Mode Distribution

Table 3 reports failure mode frequency across error cases. Spatial state collapse dominates SR errors, transformation hallucination dominates ST, and tracking loss dominates MOT. These modes are rare on CC, where T-CoT errors stem from counting mistakes and attribute misidentification.

7 Discussion

Are VLMs Actually "Thinking in Pictures"? A reasonable concern is that V-CoT might not constitute visual reasoning at all: the model writes code, and the Python interpreter performs the geometric computation. The V-CoT-NF ablation directly addresses this. If V-CoT were pure delegation, V-CoT-NF (which runs identical code but hides the result) would match V-CoT, since the interpreter does the same work either way. Instead, V-CoT-NF underperforms V-CoT by 26.5 points on ST and 19.4 points on MOT. The model is reading information off the rendered image that it cannot recover from the code alone. We note that "thinking in pictures" here is external (a PNG buffer the model perceives via its visual encoder), not internal latent visual reasoning. Whether internal latent visual reasoning is achievable, and how it would compare to external rendering, is a question for future work.

When to Think in Words vs. Pictures. On this benchmark, tasks toward the spatial end (ST, MOT) are best served by V-CoT, tasks toward the symbolic end (CC) by T-CoT, and tasks blending both (SR, GR) by V-CoT with meaningful T-CoT gains as well (Figure 7).

Recommendations for Practitioners. **(1)** Avoid blindly adding "think step by step" to spatial reasoning prompts; on our controlled tasks it hurts ST (-16.5) and MOT (-12.7). **(2)** For spatial tasks, V-CoT requires both code generation and

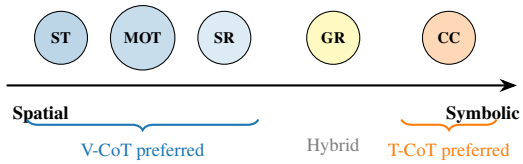

Figure 7: Taxonomy of VISCOT-DIAG categories along the spatial-symbolic axis. Tasks requiring spatial state maintenance favor V-CoT; tasks requiring symbolic decomposition favor T-CoT.

the rendered observation to deliver large gains; either alone is insufficient. **(3)** For compositional counting, T-CoT or ST-CoT is preferable. **(4)** Consider adaptive routing: a simple classifier achieves 78.2% versus 74.0% for V-CoT-everywhere on this benchmark while reducing average cost.

Cost-Accuracy Tradeoff. V-CoT consumes 3.2× more tokens than T-CoT and requires a second image encoder pass per turn. For ST and MOT, the 25- to 40-point improvement over T-CoT easily justifies the cost; for CC, it does not. V-CoT-NF removes the second encoder pass and is thus cheaper than V-CoT, but its substantially lower accuracy on the spatial categories means it is rarely the right default.

Relation to Prior Multimodal Benchmarks. Our findings are complementary to MMVU (Zhao et al., 2025) and MMMU-Pro (Yue et al., 2025), which document that direct answering and naive textual CoT struggle on complex multimodal tasks. We do not claim novelty on *whether* CoT struggles in multimodal settings; that is established. Our contribution is a controlled paradigm comparison (including a code-only ablation) on a deliberately simple, diagnostic distribution where the failure modes can be cleanly attributed and counted.

8 Limitations

Several scope conditions on our claims are worth stating explicitly.

Task simplicity. VISCOT-DIAG comprises relatively simple, controlled visual tasks (2D shapes, grid worlds, CLEVR-style scenes). This is intentional, since simple tasks support cleaner attribution of failure modes, but it means our findings should not be read as a general claim about visual reasoning in the wild. Tasks requiring deeper semantic reasoning, real-world visual complexity, or multi-disciplinary expert knowledge (as in MMVU and MMMU-Pro) may show different patterns; we phrase our claims accordingly throughout.

Code generation as confound. V-CoT relies on the model's code generation ability. Weaker code generators produce incorrect annotations that mislead reasoning. Our V-CoT-NF ablation partially decouples code generation from visual perception, but the absolute level of V-CoT performance still depends on code quality. Future work could provide oracle annotations to fully isolate the effect of visual feedback.

2D, English-only, synthetic. VISCOT-DIAG focuses on 2D reasoning with synthetic and semi-synthetic images and English prompts. Real-world photographs introduce occlusion, ambiguity, and lighting variation that may interact differently with reasoning modality. Languages with different spatial reference frames may also yield different patterns.

External vs. internal visual reasoning. V-CoT uses external code execution with a 5-turn limit. Whether similar benefits can be achieved through internal latent visual representations remains an open question.

Failure mode taxonomy. Our three failure modes (state collapse, transformation hallucination, tracking loss) are empirically grounded but not exhaustive. Other patterns may emerge with different task distributions or model architectures.

9 Conclusion

We presented a controlled diagnostic study of textual versus visual chain-of-thought reasoning in vision-language models on VISCOT-DIAG, a benchmark of 1,200 simple visual tasks. On this distribution, textual CoT degrades performance on spatial transformation (-16.5%) and multi-object tracking (-12.7%) relative to direct answering, while visual CoT yields gains up to +25.4%. A code-only ablation (V-CoT-NF) shows that code generation alone recovers only 36% of V-CoT's gain over T-CoT on spatial transformation: most of the benefit comes from the rendered image, not from delegating computation to a Python interpreter. We documented three failure modes (spatial state collapse, transformation hallucination, tracking loss) and showed that adaptive modality routing achieves 78.2% versus 74.0% for V-CoT-everywhere. We encourage practitioners to think of

reasoning modality as a per-task design choice on visual tasks rather than a universal default, while noting that our claims hold on this controlled distribution and may not transfer to richer real-world visual tasks.

References

Anthropic. 2026. System card: Claude Opus 4.6. https://www.anthropic.com/claude-opus-4-6-system-card.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, and 1 others. 2025. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. 2024. Are we on the right way for evaluating large vision-language models? In *Advances in Neural Information Processing Systems*.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. 2024. BLINK: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision (ECCV)*.

Gemini Team, Google DeepMind. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Yushi Hu, Weijia Shi, Xingyu Fu, Dan Roth, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Ranjay Krishna. 2024. Visual sketchpad: Sketching as a visual chain of thought for multimodal language models. In *Advances in Neural Information Processing Systems*, volume 37.

Drew A. Hudson and Christopher D. Manning. 2019. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6700–6709.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. 2017. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2901–2910.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213.

Stephen M. Kosslyn. 1995. Mental imagery. In Stephen M. Kosslyn and Daniel N. Osherson, editors, *An Invitation to Cognitive Science, Vol. 2: Visual Cognition*, pages 267–296. MIT Press.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2024. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations*.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems*, volume 35.

Chancharik Mitra, Brandon Huang, Trevor Darrell, and Roei Herzig. 2024. Compositional chain-of-thought prompting for large multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

OpenAI. 2025. GPT-5 system card. *arXiv preprint arXiv:2601.03267*.

Allan Paivio. 1991. Dual coding theory: Retrospect and current status. *Canadian Journal of Psychology*, 45(3):255–287.

Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. 2024. Visual CoT: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. In *Advances in Neural Information Processing Systems*, volume 37.

Roger N. Shepard and Jacqueline Metzler. 1971. Mental rotation of three-dimensional objects. *Science*, 171(3972):701–703.

Omkar Thawakar, Dinura Dissanayake, Ketan More, Ritesh Thawkar, Ahmed Heakl, Noor Ahsan, Yuhao Li, Mohammed Zumri, Jean Lahoud, Rao Muhammad Anwer, Hisham Cholakkal, Ivan Laptev, Mubarak Shah, Fahad Shahbaz Khan, and Salman H. Khan. 2025. LlamaV-o1: Rethinking step-by-step visual reasoning in LLMs. In *Findings of the Association for Computational Linguistics: ACL 2025*.

Yikun Wang, Siyin Wang, Qinyuan Cheng, Zhaoye Fei, Liang Ding, Qipeng Guo, Dacheng Tao, and Xipeng Qiu. 2025. VisuoThink: Empowering LVLM reasoning with multimodal tree search. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pages 21707–21719.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837.

Wenshan Wu, Shaoguang Mao, Yadong Zhang, Yan Xia, Li Dong, Lei Cui, and Furu Wei. 2024. Mind's eye of LLMs: Visualization-of-thought elicits spatial reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 37.

Guowei Xu, Peng Jin, Ziang Wu, Hao Li, Yibing Song, Lichao Sun, and Li Yuan. 2025. LLaVA-CoT: Let vision language models reason step-by-step. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2087–2098.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, and 1 others. 2024. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, Yu Su, Wenhu Chen, and Graham Neubig. 2025. MMMU-Pro: A more robust multi-discipline multimodal understanding benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*.

Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. 2024. Multimodal chain-of-thought reasoning in language models. *Transactions on Machine Learning Research*.

Yilun Zhao, Lujing Xie, Haowei Zhang, Guo Gan, Yitao Long, Zhiyuan Hu, Tongyan Hu, Weiyuan Chen, Chuhan Li, Junyang Song, Zhijian Xu, Chengye Wang, Weifeng Pan, Ziyao Shangguan, Xiangru Tang, Zhenwen Liang, Yixin Liu, Chen Zhao, and Arman Cohan. 2025. MMVU: Measuring expert-level multi-discipline video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

A Prompt Templates

We provide the full prompt templates used for each CoT paradigm below. All prompts are preceded by the task-specific image, passed as the first content block in the multimodal message.

Direct (D).

> "Look at the image and answer the following question. Provide only the final answer with no explanation.\n\nQuestion: {question}\n\nAnswer:"

Textual CoT (T-CoT).

> "Look at the image and answer the following question. Think step by step, explaining your reasoning carefully before giving your final answer.\n\nQuestion: {question}\n\nLet me think through this step by step:"

Structured Textual CoT (ST-CoT).

> "Look at the image and answer the following question. Structure your response using the following format:\n\nSUMMARY: Briefly state what you need to determine.\nOBSERVATION: Describe the relevant visual content you see in the image.\nREASONING: Work through the problem step by step, explaining your logic.\nANSWER: State your final answer clearly.\n\nQuestion: {question}"

Visual CoT, no feedback (V-CoT-NF).

> "You have access to a Python environment with PIL/Pillow for image manipulation. The input image is loaded as `img`. To reason about this question, write Python code to annotate the image with helpful visual aids. **Note: the rendered output of your code will not be shown back to you. You must reason about what your code would produce and answer based on that mental simulation.**\n\nAvailable drawing primitives: `draw.rectangle()`, `draw.line()`, `draw.polygon()`, `draw.ellipse()`, `draw.text()`, and affine transformations via `img.rotate()` and `img.transpose()`.\n\nQuestion: {question}"

Visual CoT (V-CoT).

> "You have access to a Python environment with PIL/Pillow for image manipulation. The input image is loaded as `img`. To reason about this question, write Python code to annotate the image with helpful visual aids such as bounding boxes, arrows, highlighted regions, auxiliary lines, or transformed shapes. After each annotation step, the updated image will be shown to you. Observe the result and continue reasoning. When you are confident in your answer, state it clearly.\n\nAvailable drawing primitives: `draw.rectangle()`, `draw.line()`, `draw.polygon()`, `draw.ellipse()`, `draw.text()`, and affine transformations via `img.rotate()` and `img.transpose()`.\n\nQuestion: {question}\n\nBegin by examining the image and deciding what to draw first:"

For V-CoT, the multi-turn pipeline operates as follows: (1) the model generates a code block; (2) the code is executed in a sandboxed Python environment with the current image; (3) the resulting annotated image is re-encoded and sent back to the model; (4) the model generates either another code block or a final answer. We allow up to 5 annotation turns. V-CoT-NF uses the same pipeline except step (3) is omitted; the executed image is recorded for offline AQ scoring but is not returned to the model.

B Benchmark Construction Details

Spatial Relation (SR). 120 instances from BLINK (Fu et al., 2024) spatial split + 120 from CLEVR (Johnson et al., 2017). Difficulty: Easy (2–3 objects), Medium (4–5), Hard (6+).

Spatial Transformation (ST). Procedurally generated 2D shapes with controlled transformation parameters (rotations of 90/180/270 degrees, horizontal/vertical reflections). Easy: one transformation. Medium: two. Hard: three or more. Ground truth computed analytically.

Multi-Object Tracking (MOT). 8×8 grid worlds. Easy: 3 objects, 2 moves. Medium: 4–5 objects, 3–4 moves. Hard: 6–7 objects, 5–6 moves. Ground truth from movement replay.

Compositional Counting (CC). CLEVR and GQA instances requiring multi-attribute filtering with spatial qualifiers. Counts 0–7. Difficulty by attribute filter count: Easy (1–2), Medium (3), Hard (4+).

Geometric Reasoning (GR). 120 from MathVista geometry plus 120 synthetic. Difficulty by reasoning chain length: Easy (1–2), Medium (3–4), Hard (5+).

All synthetic instances use deterministic random seeds. Human validation on 10% stratified sample (120 instances, 3 annotators): 96.7% agreement with ground truth, Fleiss' $\kappa = 0.91$.

C Full Per-Model Results

Table 4 presents the complete per-model, per-category, per-paradigm breakdown.

D Adaptive Routing Preliminary Experiment

Using a fine-tuned DistilBERT model with 5-fold cross-validation on the 1,200 instances:

Model	Paradigm	SR	ST	MOT	CC	GR	Avg.
GPT-5	Direct	70.2	53.6	62.4	59.1	51.8	59.4
	T-CoT	75.4	38.7	50.6	77.8	67.2	61.9
	ST-CoT	77.0	41.5	54.7	79.6	69.1	64.4
	V-CoT-NF	73.1	53.2	56.3	73.4	68.5	64.9
	V-CoT	**83.1**	**81.7**	**77.4**	72.3	**78.6**	**78.6**
Claude Opus 4.6	Direct	67.8	52.4	60.2	56.7	49.3	57.3
	T-CoT	73.6	36.2	47.8	75.4	65.1	59.6
	ST-CoT	75.2	39.4	52.1	77.3	67.0	62.2
	V-CoT-NF	70.8	51.7	54.6	71.2	65.7	62.8
	V-CoT	**80.4**	**78.9**	**75.0**	69.7	**75.8**	**75.9**
Gemini 2.5 Pro	Direct	65.3	51.8	58.7	55.2	47.1	55.6
	T-CoT	68.4	34.5	45.2	70.6	59.3	55.6
	ST-CoT	70.1	37.6	49.5	73.2	61.5	58.4
	V-CoT-NF	68.6	49.4	53.0	67.4	61.8	60.0
	V-CoT	**76.7**	**76.4**	**72.0**	67.4	**71.6**	**72.8**
Qwen3-VL-72B	Direct	60.9	49.1	54.5	52.4	43.8	52.1
	T-CoT	64.2	31.2	41.5	66.9	55.4	51.8
	ST-CoT	65.8	33.8	44.9	68.5	57.6	54.1
	V-CoT-NF	63.5	46.8	50.3	63.2	60.1	56.8
	V-CoT	**72.4**	**70.2**	**67.7**	64.0	**67.5**	**68.4**

Table 4: Full per-model results on VISCOT-DIAG. V-CoT achieves the highest average for every model. T-CoT degrades ST and MOT universally. V-CoT-NF sits between T-CoT and V-CoT on spatial categories, isolating the contribution of code generation from rendered visual feedback. Best per model-category (excluding CC) is **bolded**.

Strategy	Avg. Accuracy (%)
T-CoT everywhere	57.3
ST-CoT everywhere	59.8
V-CoT-NF everywhere	61.1
V-CoT everywhere	74.0
Adaptive router (T vs. V)	78.2
Oracle routing	81.5

Table 5: Adaptive routing results. The router achieves 78.2%, outperforming the best single paradigm (V-CoT, 74.0%) by 4.2 points.

The router achieves 84.7% routing accuracy. Oracle routing (always choosing the better paradigm per instance) achieves 81.5%. Our router recovers 78.2% of this bound. The 3.3-point gap suggests room for improvement, potentially through multimodal routing that considers the image as well as the question text. We did not include V-CoT-NF as a routing target because it is rarely the best paradigm for any instance type; future work could explore three-way routing including V-CoT-NF as a cost-aware fallback.

E V-CoT and V-CoT-NF Annotation Examples

High-quality (ST). The model generates code to draw the original shape in solid fill, then draws the rotated version with a dashed outline and a curved rotation arrow, labeling both states. Under V-CoT the model observes this rendering and answers correctly. Under V-CoT-NF the model wrote the same code but answered based on its expectation of the rendering; on this particular instance both succeeded, but accuracy on similar instances diverges (V-CoT 81.7% vs. V-CoT-NF 53.2% on ST for GPT-5).

Medium-quality (GR). The model draws auxiliary lines and labels angles. The spatial structure is interpretable. V-CoT and V-CoT-NF perform similarly on GR (78.6 vs. 68.5 for GPT-5), the smallest gap among spatial-leaning categories, consistent with GR benefiting more from the symbolic structure of the code than from the rendered visual itself.

Low-quality (MOT). The model draws bounding boxes around objects but fails to update positions after described movements. Under V-CoT this leads to incorrect answers; under V-CoT-NF the model never sees the (incorrect) render but still answers from its mental simulation, which exhibits the same tracking loss failure mode.