VLMs Can Recover Analytical Solutions from Field Visualizations

Anonymous ACL submission

Abstract

Two-dimensional physical fields are governed by partial differential equations whose solutions take specific symbolic forms. Existing approaches recover these solutions from numerical data, yet visualizing the same fields as heatmaps reveals features that are far less apparent in tabular form. In this work, we find that vision-language models (VLMs) can leverage these visual cues to identify the symbolic form of a physical-field solution directly from its visualization. Through systematic evaluation of multiple frontier VLMs on a benchmark of 30 physically grounded scenarios, we confirm that VLMs consistently identify solution forms more faithfully than text-only LLMs receiving equivalent numerical data. Even zero-shot frontier VLMs substantially exceed classical symbolic regression on structural similarity (e.g., GPT-5.2: 0.676 vs. PySR: 0.380), and further aligning an 8B VLM with ground-truth-guided chain-of-thought supervision pushes this to 0.826. These findings suggest that field visualizations carry richer cues about solution forms than previously leveraged, and that VLMs offer a viable path to symbolic solution recovery.

1 Introduction

Physical fields governed by partial differential equations (PDEs) are among the most common objects in science and engineering. Their solutions, whether describing heat conduction, electrostatic potentials, or quantum wavefunctions, take specific symbolic forms that encode the underlying physics. Recovering these symbolic solutions from observations has been pursued through symbolic regression (SR) (Koza, 1992; Schmidt and Lipson, 2009; Brunton et al., 2016; Udrescu and Tegmark, 2020) and, more recently, through LLM-assisted equation discovery (Shojaee et al., 2025; Zhang et al., 2025a). These methods operate on numerical point sets and have achieved notable success within their respective search spaces.

However, the same physical fields are also routinely *visualized*: heatmaps, contour plots, and gradient fields are standard tools in scientific practice. Such visualizations make certain features of the solution immediately apparent: symmetry axes, singularity locations, oscillation patterns, and decay profiles are all visually salient in a heatmap, yet extracting the same information from a table of numbers requires deliberate analysis. Despite this, the visual modality has not been systematically explored for symbolic solution recovery.

In this work, we investigate whether vision-language models (VLMs) can leverage these visual cues to identify solution forms directly from field visualizations. By "discover" or "recover," we mean identifying a closed-form symbolic expression from observations under a controlled benchmark setting. Through systematic evaluation of multiple frontier VLMs on a benchmark of 30 physically grounded scenarios, we find that VLMs consistently identify solution forms more faithfully when given field visualizations than when the same information is presented as numerical tables to text-only LLMs. Even zero-shot frontier VLMs achieve structural similarity substantially exceeding classical SR baselines (e.g., GPT-5.2: 0.676 vs. PySR: 0.380). By further aligning an 8B open-weight VLM with chain-of-thought (CoT) supervision, we push structural similarity to 0.826, substantially exceeding both zero-shot VLMs and SR baselines under matched evaluation.

We note that our setting and classical SR are not identical: our benchmark focuses on 2D steady-state scalar fields with rich visual structure, while SR benchmarks emphasize broader function classes and higher-dimensional problems. The comparison is therefore informative rather than definitive, and highlights complementary strengths: SR methods achieve strong numerical fit within their operator sets, while VLMs excel at identifying solution forms, particularly for special-function families

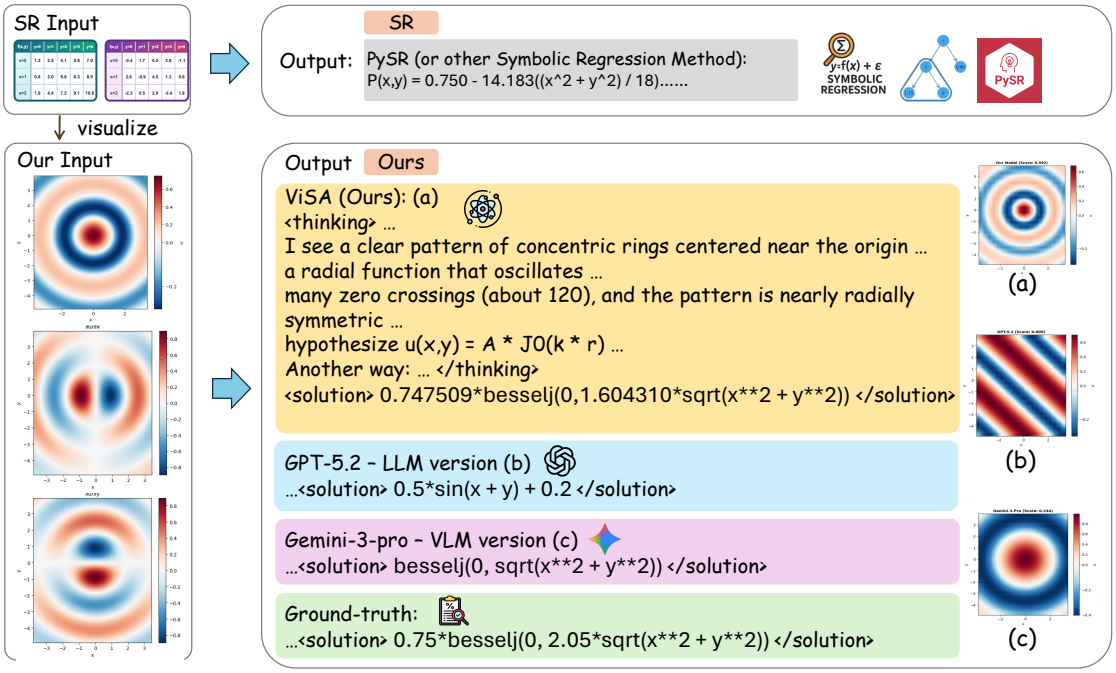

Figure 1: **Visual-to-symbolic solution inference.** Given heatmap visualizations of a physical field and its gradient components, a VLM identifies the symbolic solution form and outputs a closed-form SymPy expression.

(e.g., Bessel, Airy) that may fall outside a given SR search space.

We attribute this capability to two properties of the visual modality. First, field visualizations provide a spatially organized representation in which solution-relevant features (radial symmetry, oscillatory rings, boundary-layer gradients) are perceptually grouped, whereas the same information in a numerical table is scattered across rows without spatial coherence. Second, VLMs generate expressions in natural language (SymPy syntax) and are not constrained to a predefined operator library, allowing them in principle to produce any function expressible in language.

To operationalize this investigation, we construct ViSA-Bench, a benchmark of 30 physically grounded scenarios with exact symbolic ground truth, and develop ViSA-R2, an 8B VLM aligned with chain-of-thought supervision. The CoT traces are synthesized via a programmatic pipeline that uses ground-truth access to construct oracle-guided rationales: it extracts visual and numerical cues from the observation, then aligns them with the solution to produce a coherent derivation whose final form references only observable quantities. Fine-tuning on these traces enables the model to internalize this reasoning pathway.

Our contributions are: **(1)** We provide systematic evidence that VLMs identify the symbolic form of PDE solutions from field visualizations, consistently outperforming text-only LLMs and matching or exceeding classical SR on structural similarity. **(2)** We develop ViSA-R2, a CoT-aligned 8B VLM that substantially exceeds all baselines on structure recovery and, combined with post-hoc coefficient optimization, achieves competitive numerical accuracy. **(3)** We release ViSA-Bench, a benchmark of 30 physically grounded scenarios with exact symbolic ground truth, along with the programmatically synthesized CoT training data.

2 Related Work

2.1 VLMs for Scientific and Mathematical Reasoning

Vision-language models have developed from contrastive image-text pretraining, such as CLIP (Radford et al., 2021), to instruction-following multimodal assistants such as LLaVA (Liu et al., 2023), Qwen-VL (Bai et al., 2024), GPT-4V (OpenAI, 2023), and Gemini (Team, 2023). This progress has enabled broad evaluation on scientific and mathematical visual understanding. Prior benchmarks study chart understanding (Kafle et al., 2018; Methani et al., 2020; Masry et al., 2022), scientific figure interpretation (Roberts et al., 2024), diagram QA (Kembhavi et al., 2016, 2017), and science QA with explanations (Lu et al., 2022; Zhang et al., 2024; Su et al., 2026). Physics-specific

benchmarks, including PhysReason (Zhang et al., 2025b), PhysUniBench (Wang et al., 2025), and UGPhysics (Xu et al., 2025), further test diagram-grounded physical reasoning. Mathematical benchmarks such as MathVista and MMMU (Lu et al., 2024; Yue et al., 2024) show that VLMs can use visual context for mathematical reasoning, while still facing challenges in symbolic abstraction and formalization.

These evaluations are mainly descriptive or classificatory: the model answers questions about visual content, selects among options, or produces a short explanation. In contrast, our task requires a model to generate an executable symbolic expression, with instantiated constants, directly from field visualizations. This places our work at the intersection of VLM evaluation and scientific equation discovery.

2.2 Symbolic Regression and Visual Equation Discovery

Symbolic regression recovers mathematical expressions from numerical observations. Classical approaches include genetic programming (Koza, 1992; Schmidt and Lipson, 2009), sparse identification methods such as SINDy (Brunton et al., 2016), and physics-informed systems such as AI Feynman (Udrescu and Tegmark, 2020; Udrescu et al., 2020). These methods search over expression spaces defined by operators, basis functions, and model-selection criteria, and they can provide accurate and interpretable fits when the target structure is well represented in the search space. Neural and LLM-assisted methods extend this landscape by using pretrained sequence models or language models to propose candidate expressions, programs, or refinements (Biggio et al., 2021; Merler et al., 2024; Shojaee et al., 2025; Zhang et al., 2025a).

Visual equation-discovery work has begun to connect perception with symbolic modeling. Some methods infer physical laws from video or image-like encodings of observations (Udrescu and Tegmark, 2021; Xing et al., 2021), ViSymRe predicts equation skeletons from visual representations (Li et al., 2024), and VIPER-R1 uses VLM perception to recover physical laws from trajectories (Liu et al., 2025). Our setting differs from classical SR in several ways. SR usually starts from numerical point sets and emphasizes broad function classes, higher-dimensional inputs, and numerical accuracy. We instead study 2D steady-state physical fields, use heatmaps and gradient images as input, and evaluate whether a VLM can identify the symbolic solution form, including special-function families such as Bessel, Airy, and piecewise solutions. These directions are complementary: traditional methods provide strong numerical fitting machinery, while VLMs may use visual structure to propose solution forms that can subsequently be refined numerically.

3 Methodology

3.1 Problem Setup

We study visual-to-symbolic solution inference for 2D steady-state scalar fields. Given an observation $\mathcal{X} = (I_1, I_2, m)$, consisting of two RGB heatmaps (I_1: field $u(x, y)$; I_2: gradient components $\partial u/\partial x$, $\partial u/\partial y$) and minimal auxiliary metadata m (domain bounds, basic statistics; typically $<$10 scalars), the model produce a SymPy expression s representing the instance-specific solution. We denote the VLM parameterized by θ as f_θ:

$$s = f_\theta(I_1, I_2, m), \qquad s \in \texttt{SymPyStrings}. \quad (1)$$

With chain-of-thought alignment, the model additionally produces an intermediate rationale r:

$$(r, s) = f_\theta(I_1, I_2, m). \quad (2)$$

The gradient heatmap I_2 is not additional privileged information: it is derived from the same 20×20 observation grid (400 points (x, y, u)) via finite-difference approximations of $\partial u/\partial x$ and $\partial u/\partial y$, then rendered as images. In other words, images (I_1, I_2) encode exactly the same numerical content that classical SR methods receive as a table, but in a spatially organized format that makes structural features perceptually accessible. Classical SR methods cannot exploit this visual cues because they operate on unordered point sets; by contrast, VLMs process spatial layouts natively.

We distinguish three concepts used throughout this paper: *visual cues* refers to perceptually salient features in field visualizations (e.g., symmetry axes, oscillation rings, decay profiles); *solution form* denotes the symbolic function family of the ground-truth expression (e.g., Bessel, Airy, exponential); and *structural similarity* is a tree-edit-distance-based metric that compares predicted and ground-truth expression trees (formally defined in §3.3).

3.2 Benchmark

ViSA-Bench is grounded in real physical contexts (electrostatics, heat transfer, quantum mechanics,

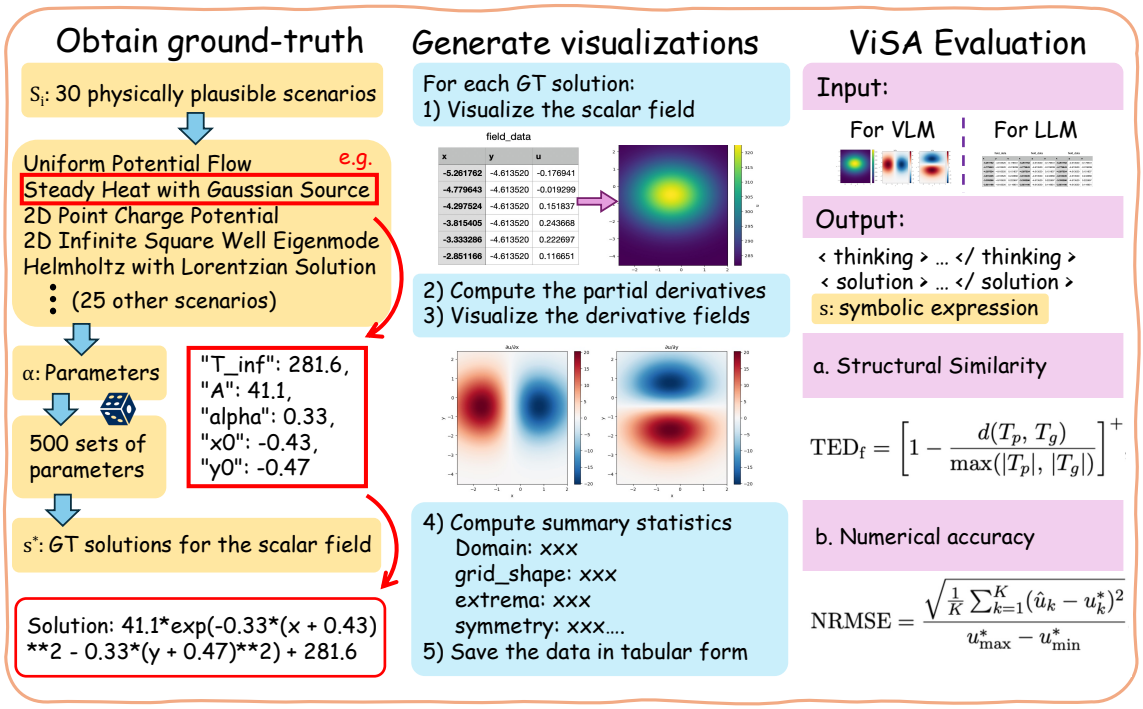

Figure 2: **Overall pipeline.** (Left) Ground-truth generation: a scenario is selected from 30 physical families, parameters are randomly sampled, and the closed-form solution is computed symbolically (red box shows a concrete instance). (Middle) Visualization: the scalar field and its gradient components are rendered as heatmap images. (Right) Evaluation: VLMs receive field images while LLMs receive numerical tables; both output a SymPy expression, which is scored by TED(family) (structural similarity) and NRMSE (numerical accuracy).

fluid dynamics) and includes both elementary functions and special-function families (Bessel, Airy, piecewise) that commonly arise in these domains. This breadth reflects the diversity of solution forms encountered in practice and provides a controlled testbed for evaluating symbolic solution recovery.

We adopt a solution-first construction pipeline. Each scenario $\mathcal{S}_i$ defines a parametric solution template $f_i(x, y; \alpha)$ drawn from the analytic solutions of a specific PDE class, together with a parameter distribution $p_i(\alpha)$ that controls the diversity of instances. For each instance, we sample $\alpha \sim p_i(\alpha)$, instantiate the symbolic solution $u^*(x, y) = f_i(x, y; \alpha)$ in SymPy, compute partial derivatives symbolically, and render the field and gradient images on a 20×20 grid. This ensures exact ground truth without numerical solver artifacts. The observation triplet (I_1, I_2, m) is then assembled: two heatmap images and a compact metadata vector containing domain bounds and basic statistics. In total, 30 scenarios × 500 instances = 15,000 samples; we curate 1,500 for CoT-aligned training and 150 for evaluation. The split ensures no parameter overlap between training and test instances within each scenario.

3.3 Baselines and Evaluation

We compare against three categories of baselines. VLM baselines (GPT-5.2, Claude-Opus-4.5, Claude-Sonnet-4.5, Claude-Haiku-4-5, Grok-4-Fast, Qwen3-VL-8B, Gemini-3-Pro) receive the same observation (I_1, I_2, m) under a standardized prompt. LLM-only baselines receive the same numerical table without images. Classical SR includes two PySR (Cranmer, 2023) configurations: (1) standard operators $(+, -, \times, /, \sin, \cos, \exp, \log, \sqrt{\cdot}, \sinh, \cosh, \tanh, \mathrm{abs})$; and (2) an extended set additionally including `besselj`, `airyai`, `airybi`, `besselk`, `atan2`—both with 200 iterations, maximum expression size 30, 30 populations of size 40, and 1000 cycles per iteration. All baselines receive 400 numerical tuples $(x, y, u, \partial u/\partial x, \partial u/\partial y)$.

We evaluate with two metrics that capture orthogonal aspects of symbolic recovery. TED(family) measures structural similarity: both predicted and ground-truth expressions are converted to labeled trees (T_p and T_g respectively) in which all numeric constants are collapsed to a single token C and function nodes are mapped to family labels (e.g., $\sin, \cos, \tan \rightarrow$ trig; $\sinh, \cosh, \tanh \rightarrow$ hyp;

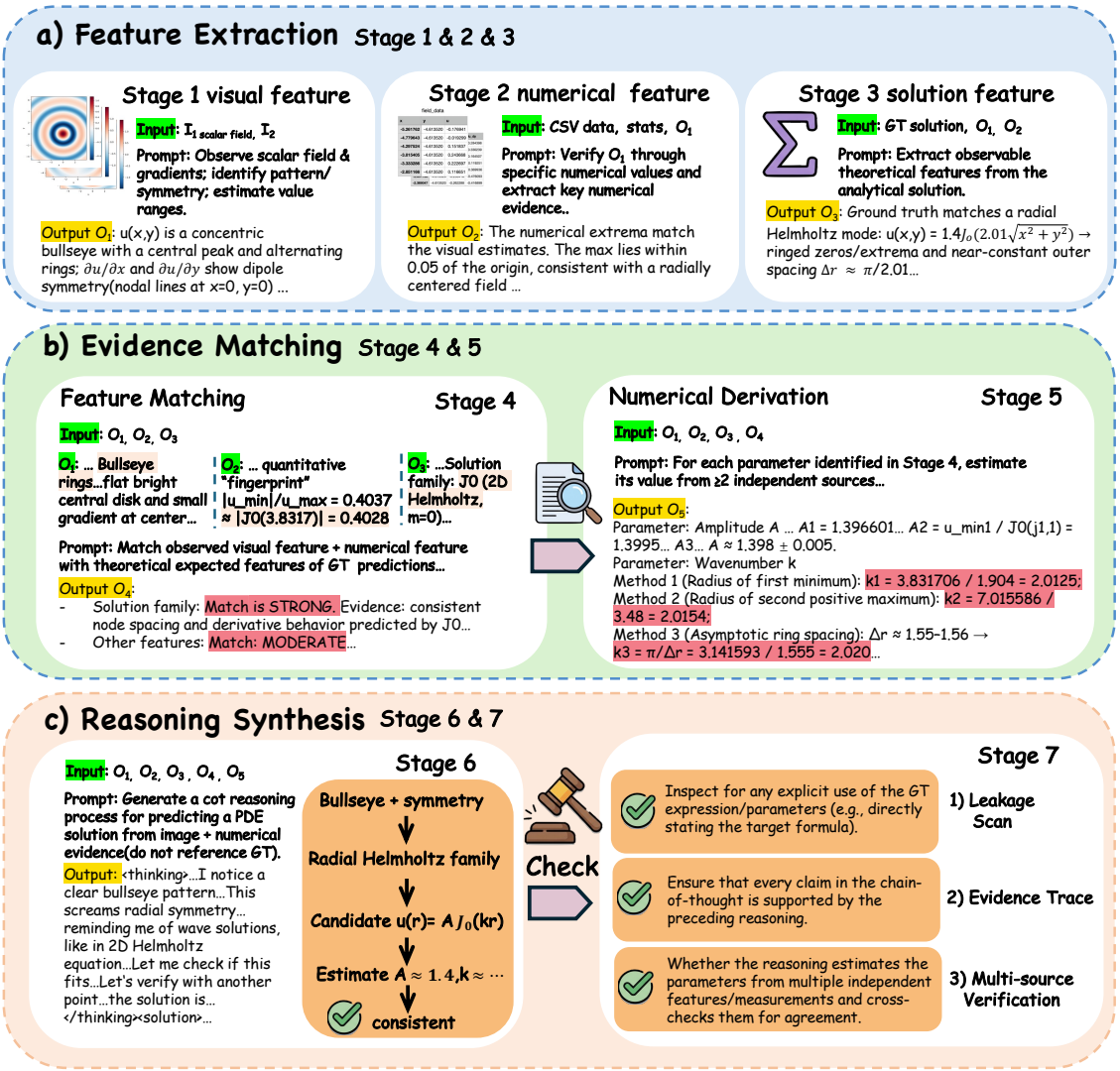

Figure 3: Ground-truth-guided CoT synthesis pipeline. Stages 1–2 extract visual and numerical cues from observations *without* consulting the ground truth; Stages 3–5 use the ground-truth solution to select and align cues into a coherent derivation; Stages 6–7 assemble and validate the final reasoning trace, ensuring no ground-truth expressions appear verbatim. Full stage-by-stage prompts are in the Appendix.

besselj, bessely, ... → bessel). Structural similarity is then

$$\mathrm{TED_f} = \left[1 - \frac{d(T_p, T_g)}{\max(|T_p|, |T_g|)}\right]^+, \quad (3)$$

where d is the tree edit distance (Zhang and Shasha, 1989), $|T|$ is the number of nodes, and $[\cdot]^+ = \max(0, \cdot)$ clamps the score to $[0, 1]$. A score of 1.0 indicates identical tree structure up to numeric constants. NRMSE measures numerical fit: predicted and ground-truth expressions are evaluated on the 400 observation grid points $\{(x_k, y_k)\}_{k=1}^{400}$ and the normalized root mean squared error is

$$\mathrm{NRMSE} = \frac{\sqrt{\frac{1}{K}\sum_{k=1}^{K}(\hat{u}_k - u_k^*)^2}}{u_{\max}^* - u_{\min}^*}. \quad (4)$$

These two metrics are complementary: a method can achieve low NRMSE with a structurally different expression (e.g., a polynomial approximating a Bessel function), or high TED(family) with imprecise coefficients. We discuss the rationale for this metric choice in Appendix D.

3.4 Ground-Truth-Guided CoT Synthesis

We synthesize reasoning traces via a 7-stage programmatic pipeline that uses ground-truth access to construct high-quality rationales (Figure 3; full prompts in Appendix). This is a form of oracle-guided rationale distillation: the ground-truth solution s^* guides which reasoning steps to include, but the final trace references only observation-derivable quantities. Stages 1–2 extract visual

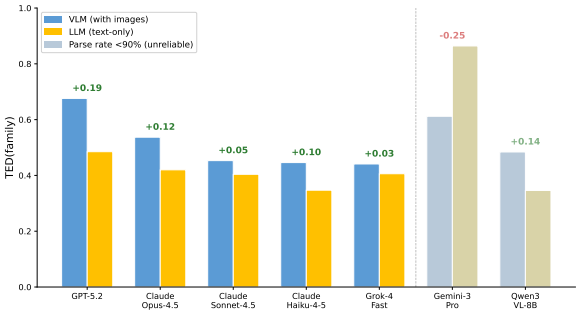

Figure 4: Visual input consistently improves structure recovery. Among the five models with parse rate $\geq 90\%$ in both modes, all achieve higher TED(family) when receiving field visualizations (VLM) than equivalent numerical tables (LLM). The largest gain is observed for GPT-5.2 (+0.191).

and numerical cues from (I_1, I_2, m) *without* consulting s^*. Stage 3 uses s^* to identify expected observable signatures (e.g., radial symmetry, oscillatory patterns). Stages 4–5 align observation-based cues with these signatures and construct a parameter-derivation narrative that is consistent with the ground truth. Stages 6–7 assemble the final rationale and validate that no ground-truth expressions or parameter values appear verbatim in the trace (see Appendix E for further discussion).

The resulting rationale follows a structured inference pathway: starting from visual pattern recognition, then forming a solution-family hypothesis, deriving parameters from observable quantities, and finally verifying consistency. Each trace is 300–800 tokens. At inference time, the model must reproduce this reasoning from observations alone, without any ground-truth access.

3.5 Model Training

We fine-tune Qwen3-VL (8B) (Yang et al., 2025) on 1,500 triplets $(\mathcal{X}, r^*, s^*)$ using standard next-token prediction on the concatenated sequence $[r^*; s^*]$. The loss is computed over both the reasoning trace and the final expression, enabling the model to learn the full inference pathway.

Training uses bfloat16 precision with learning rate 2×10^{-5}, batch size 1 with 4-step gradient accumulation (effective batch 4), 4 epochs, and maximum sequence length 8192 tokens.

4 Experiments

4.1 Main Results

Tables 1 and Figure 4 summarize our main comparisons. All zero-shot baselines are evaluated on the same 30-sample test set to ensure fair comparison across modalities; extended 150-sample LLM results are reported in Appendix B.

Among models with high parse rates ($\geq 90\%$), VLM mode (receiving field images) yields consistently higher structural similarity than LLM mode (receiving equivalent numerical tables) (Figure 4). The largest gains are observed for GPT-5.2 (+0.191) and Claude-Opus-4.5 (+0.117), confirming that field visualizations carry structural cues that are difficult to extract from numerical tables alone.

Even without task-specific training, GPT-5.2 in VLM mode achieves TED(family) of 0.676, substantially exceeding PySR (0.380). In text-only mode, five out of six high-parse-rate models still exceed or match PySR on structure. This suggests that pretrained VLMs already possess inductive biases relevant to solution-form identification, and that visual input further amplifies this capability. However, zero-shot VLMs exhibit high NRMSE (0.298–1.001), indicating that while they identify correct functional forms, their coefficient estimates are imprecise. Conversely, PySR's NRMSE (0.024) surpasses all LLM/VLM methods in numerical accuracy, illustrating the complementary strengths of the two paradigms.

ViSA-R2 achieves TED(family) of 0.826, exceeding the best zero-shot VLM (GPT-5.2: 0.676) by 22% and all other baselines. Classical SR methods achieve lower NRMSE (PySR: 0.024) reflecting their direct numerical optimization, while VLMs prioritize the solution form. We additionally report NRMSE after post-hoc L-BFGS-B coefficient refinement ($\text{NRMSE}_{\text{opt}}$), which preserves the predicted symbolic structure while fitting only the numerical constants. This two-stage approach substantially closes the numerical gap: ViSA-R2 improves from 0.229 to 0.103, and GPT-5.2 VLM from 0.298 to 0.108, validating a workflow where VLMs identify structure and lightweight optimization recovers coefficients.

Notably, extending PySR's operator set with special functions (Bessel, Airy, tanh, etc.) degrades rather than improves structural recovery (TED drops from 0.380 to 0.079). With high-expressivity operators available, PySR exploits them as generic approximators, achieving low NRMSE in $\sim 60\%$ of cases while producing structurally unrelated expressions (e.g., nested compositions of `airybi`/`sinh` to fit a logarithmic field). This illustrates that merely expanding the operator library does not solve the structure-recovery problem—the search algorithm

Method	Input	N	TED(family)↑	NRMSE↓	$\text{NRMSE}_{\text{opt}}$ ↓
Classical symbolic regression					
PySR (Cranmer, 2023)	Numerical	30	0.380	**0.024**	—
PySR + SpecialFns	Numerical	30	0.079	0.223	—
Zero-shot LLMs (text-only, 30 samples)					
GPT-5.2 (OpenAI, 2025)	Text	30	0.485	0.407	0.257
Claude-Opus-4.5 (Anthropic, 2025b)	Text	30	0.420	0.384	0.265
Grok-4-Fast (xAI, 2025)	Text	29	0.406	1.171	1.268
Gemini-Flash (Google DeepMind, 2025)	Text	29	0.391	0.399	0.335
Claude-Sonnet-4.5 (Anthropic, 2025c)	Text	28	0.404	0.504	0.420
Claude-Haiku-4-5 (Anthropic, 2025a)	Text	30	0.347	0.481	0.440
Gemini-3-Pro (Google DeepMind, 2025)	Text	6	†0.864	†0.107	—
Qwen3-VL-8B (Yang et al., 2025)	Text	5	†0.346	†0.282	—
Zero-shot VLMs (with images, 30 samples)					
GPT-5.2 (OpenAI, 2025)	Image	30	0.676	0.298	0.108
Claude-Opus-4.5 (Anthropic, 2025b)	Image	30	0.537	0.586	0.545
Grok-4-Fast (xAI, 2025)	Image	30	0.441	1.001	0.545
Claude-Sonnet-4.5 (Anthropic, 2025c)	Image	29	0.453	0.303	0.268
Claude-Haiku-4-5 (Anthropic, 2025a)	Image	28	0.446	0.672	0.664
†Qwen3-VL-8B (Yang et al., 2025)	Image	19	†0.484	†0.308	—
†Gemini-3-Pro (Google DeepMind, 2025)	Image	18	†0.612	†0.124	—
Ours (5 runs, 150 samples)					
ViSA-R2	Image	150	**0.826**	0.229	**0.103**

Table 1: Main results on ViSA-Bench (30 test samples). All zero-shot baselines are evaluated on the same 30-sample test set; ViSA-R2 is evaluated on 150 samples (5 instances per scenario). N: number of successfully parsed outputs out of 30. †Parse rate $<90\%$; metrics computed only on parsed subset. $\text{NRMSE}_{\text{opt}}$: after L-BFGS-B coefficient refinement preserving predicted structure. Full 150-sample LLM results are in Appendix.

Variant	TED↑	NRMSE↓	$\text{NRMSE}_{\text{opt}}$↓
Input ablation			
ViSA-R2 (full: $I_1{+}I_2{+}m$)	**0.826±.007**	0.229±.015	**0.103±.018**
w/o numerical ($I_1{+}I_2$)	0.664±.092	0.630±.023	0.224±.066
Scalar only ($I_1{+}m$)	0.799±.053	0.226±.048	0.127±.037
Gradient only (I_2)	0.603±.044	0.394±.065	0.211±.073
Robustness (perturbation at test time)			
+ Gaussian noise (10%)	0.813±.040	0.227±.050	0.107±.035
+ Measurement noise	0.773±.021	0.262±.019	0.158±.023
+ Low resolution	0.797±.036	0.227±.033	0.123±.045
+ Color permutation	0.750±.030	0.267±.043	0.154±.049
+ Aspect ratio change	0.813±.024	0.232±.040	0.099±.044

Table 2: Ablation study (5 runs, mean±std). I_1: field heatmap; I_2: gradient image; m: numerical metadata. $\text{NRMSE}_{\text{opt}}$: after L-BFGS-B coefficient refinement.

cannot distinguish between good numerical fit and correct functional form.

4.2 Ablation Study

Table 2 presents component ablations evaluated over 5 independent runs (mean±std reported).

Removing all numerical information ($I_1{+}I_2$ only) reduces TED from 0.826 to 0.664 and increases NRMSE from 0.229 to 0.630. Using only the gradient image further drops TED to 0.603, confirming the scalar field heatmap I_1 carries the most critical structural information. Conversely, removing I_2 (scalar only) barely affects TED (0.799), suggesting gradient visualizations provide supplementary rather than essential cues.

Post-hoc L-BFGS-B optimization consistently reduces NRMSE across all variants (e.g., 0.229→0.103 for the full system), confirming that the model recovers correct structure but imprecise constants. The largest relative gain occurs for the no-numerical variant (0.630→0.224), indicating that even without numerical input the predicted structure is partially correct and recoverable.

ViSA-R2 is robust to image perturbations at test time. We evaluate five corruption types simulating realistic degradation: (1) *Gaussian noise*: additive noise with $\sigma{=}25$ on RGB channels; (2) *Measurement noise*: a composite perturbation combining light Gaussian noise ($\sigma{=}10$), salt-and-pepper corruption (0.5% of pixels), and horizontal stripe artifacts (5 random scan lines with ± 40 intensity shifts), simulating sensor readout artifacts in scientific instruments; (3) *Low resolution*: 4× bicubic downscale followed by upscale to original size; (4) *Color permutation*: R↔G channel swap; (5) *Aspect ratio change*: width compressed to 80% with white padding. Gaussian noise, resolution reduction, and aspect-ratio distortion all yield TED≥0.797 with overlapping confi-

dence intervals to the baseline. Measurement noise causes a mild drop (TED 0.773±0.021), while color permutation produces the largest degradation (TED 0.750±0.030), suggesting partial reliance on learned color semantics. After coefficient refinement, all perturbation variants achieve $\text{NRMSE}_{\text{opt}} \leq 0.158$.

4.3 Analysis

Why do VLMs excel at structure recovery? We attribute this to two factors. First, *visual inductive bias*: a heatmap makes structural properties (radial symmetry, oscillatory rings, logarithmic singularities, exponential decay) immediately perceptible—precisely the cues that guide ansatz selection. The gradient image amplifies this by rendering spatial derivatives as visual patterns. VLMs extract rich spatial structure from the same 400 data points that SR receives as an unordered table. Second, *unbounded function vocabulary*: classical SR searches a finite operator set ($\{+, -, \times, /, \sin, \cos, \exp, \log\}$), whereas VLMs can hypothesize any function expressible in natural language. When a VLM sees oscillatory rings with radial decay, it can directly produce $J_0(kr)$; PySR cannot unless `besselj` is in its library. As shown in Table 1, even adding special functions to PySR's operator set degrades performance (TED: 0.380→0.079) because the search algorithm exploits them as generic approximators rather than identifying correct structure.

Complementarity and the two-stage workflow. The TED-vs-NRMSE landscape reveals that VLMs and SR occupy complementary regions: VLMs recover correct structure with imprecise coefficients, while SR achieves low NRMSE with structurally incorrect approximations. This motivates the two-stage workflow validated by the $\text{NRMSE}_{\text{opt}}$ column: VLM proposes the parametric form, then L-BFGS-B fits constants. Both ViSA-R2 (0.229→0.103) and GPT-5.2 VLM (0.298→0.108) benefit substantially, confirming that correct structure enables effective coefficient recovery.

5 Discussion

5.1 Why Structure Recovery Matters Beyond Numerical Fit

One might ask why structural similarity matters when NRMSE already measures prediction quality. We argue that recovering the correct symbolic form provides value that numerical accuracy alone cannot: (1) *Interpretability*: a correctly identified Bessel function reveals the underlying wave physics, while a degree-8 polynomial with equivalent MSE does not; (2) *Generalization*: the correct functional form extrapolates beyond the training domain, whereas polynomial approximations diverge; (3) *Scientific utility*: downstream tasks (stability analysis, parameter sensitivity, dimensional analysis) require the symbolic form, not merely pointwise predictions. In our benchmark, PySR achieves NRMSE of 0.024 but TED(family) of only 0.380, meaning it fits the data well with structurally incorrect expressions (e.g., polynomials approximating Bessel functions within a bounded domain). This motivates our emphasis on TED(family) as the primary metric for evaluating whether a method has identified the correct solution form.

5.2 Scope and Limitations of the Comparison

Our benchmark includes more complex functions than standard SR benchmarks (special-function families) but is restricted to 2D static fields. We do not claim VLMs universally outperform SR; rather, we demonstrate a specific regime—special-function structure recovery from field visualizations—where VLMs have a decisive advantage. Extending to time-dependent, nonlinear, or higher-dimensional systems is future work.

6 Conclusion

We introduced visual-to-symbolic analytical solution inference as a VLM-centric capability: recovering an instance-specific SymPy expression directly from visualizations of physical fields and their derivatives. To make this setting measurable and reproducible, we constructed a VLM-ready benchmark spanning 30 parametric steady-state scenarios with verifiable ground truth, and evaluated predictions using structural similarity (TED) and numerical accuracy (NRMSE) metrics. We further proposed a programmatic framework for ground-truth-guided CoT synthesis and applied CoT alignment during fine-tuning, enabling ViSA-R2 to infer solution families and improve parameter estimation. Under a standardized protocol, ViSA-R2 outperforms both open-source and closed-source frontier VLMs, providing evidence that field visualizations and training-time reasoning alignment are effective for end-to-end analytical solution inference.

Limitations

ViSA-Bench uses clean synthetic data restricted to 2D steady-state scalar fields (predominantly linear, with some nonlinear stationary cases). Real scientific data involves measurement noise, partial observability, and rendering artifacts. Our robustness ablations (Table 2) suggest moderate resilience to perturbations, but systematic evaluation under realistic experimental conditions and extension to time-dependent or higher-dimensional systems remain future work.

References

Anthropic. 2025a. Claude haiku 4.5. Model release.

Anthropic. 2025b. Claude opus 4. Model release.

Anthropic. 2025c. Claude sonnet 4. Model release.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2024. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. arXiv:2308.12966.

Luca Biggio, Tommaso Bendinelli, Alexander Neitz, Aurelien Lucchi, and Giambattista Parascandolo. 2021. Neural symbolic regression that scales. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 936–945.

Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. 2016. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937.

Miles Cranmer. 2023. Interpretable machine learning for science with pysr and symbolicregression.jl. *arXiv preprint arXiv:2305.01582*.

Google DeepMind. 2025. Gemini 3 pro. Model release.

Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. 2018. DVQA: Understanding data visualizations via question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5648–5656.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. 2016. A diagram is worth a dozen images. In *Computer Vision – ECCV 2016*, volume 9908 of *Lecture Notes in Computer Science*, pages 235–251. Springer International Publishing.

Aniruddha Kembhavi, Minjoon Seo, Dustin Schwenk, Jonghyun Choi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4999–5007.

John R. Koza. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press.

William La Cava, Patryk Orzechowski, Bogdan Burlacu, Fabricio Olivetti de Franca, Marco Virgolin, Ying Jin, Michael Kommenda, and Jason H. Moore. 2021. Contemporary symbolic regression methods and their relative performance. In *Proceedings of the NeurIPS 2021 Track on Datasets and Benchmarks*.

Da Li, Junping Yin, Jin Xu, Xinxin Li, and Juan Zhang. 2024. Visymre: Vision-guided multimodal symbolic regression. *arXiv preprint arXiv:2412.11139*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pages 34892–34916.

Jiaqi Liu, Songning Lai, Pengze Li, Di Yu, Wenjie Zhou, Yiyang Zhou, Peng Xia, Zijun Wang, Xi Chen, Shixiang Tang, Lei Bai, Wanli Ouyang, Mingyu Ding, Huaxiu Yao, and Aoran Wang. 2025. Mimicking the physicist's eye: A vlm-centric approach for physics formula discovery. arXiv:2508.17380.

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 (ICLR)*.

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 (NeurIPS)*.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279.

Matteo Merler, Katsiaryna Haitsiukevich, Nicola Dainese, and Pekka Marttinen. 2024. In-context symbolic regression: Leveraging large language models for function discovery. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 427–444. Association for Computational Linguistics.

Nitesh Methani, Pritha Ganguly, Mitesh M. Khapra, and Pratyush Kumar. 2020. PlotQA: Reasoning over scientific plots. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1527–1536.

OpenAI. 2023. GPT-4V(ision) system card. Technical report.

OpenAI. 2025. Gpt-5.2. Model release.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Jonathan Roberts, Kai Han, Neil Houlsby, and Samuel Albanie. 2024. Scifibench: Benchmarking large multimodal models for scientific figure interpretation. In *Advances in Neural Information Processing Systems*, volume 37. Datasets and Benchmarks Track.

Michael Schmidt and Hod Lipson. 2009. Distilling free-form natural laws from experimental data. *Science*, 324(5923):81–85.

Parshin Shojaee, Kazem Meidani, Shashank Gupta, Amir Barati Farimani, and Chandan K. Reddy. 2025. LLM-SR: Scientific equation discovery via programming with large language models. In *International Conference on Learning Representations (ICLR)*.

Encheng Su, Jianyu Wu, Chen Tang, Lintao Wang, Pengze Li, Aoran Wang, Jinouwen Zhang, Yizhou Wang, Yuan Meng, Xinzhu Ma, Shixiang Tang, and Houqiang Li. 2026. Sciif: Benchmarking scientific instruction following towards rigorous scientific intelligence. arXiv:2601.04770.

Gemini Team. 2023. Gemini: A family of highly capable multimodal models. arXiv:2312.11805.

Silviu-Marian Udrescu, Andrew Tan, Jiahai Feng, Orisvaldo Neto, Tailin Wu, and Max Tegmark. 2020. AI Feynman 2.0: Pareto-optimal symbolic regression exploiting graph modularity. arXiv:2006.10782.

Silviu-Marian Udrescu and Max Tegmark. 2020. AI Feynman: A physics-inspired method for symbolic regression. *Science Advances*, 6(16):eaay2631.

Silviu-Marian Udrescu and Max Tegmark. 2021. Symbolic pregression: Discovering physical laws from distorted video. *Physical Review E*, 103:043307.

Lintao Wang, Encheng Su, Jiaqi Liu, Pengze Li, Jiabei Xiao, Wenlong Zhang, Xinnan Dai, Xi Chen, Yuan Meng, Lei Bai, Wanli Ouyang, Shixiang Tang, Aoran Wang, and Xinzhu Ma. 2025. Physunibench: An undergraduate-level physics reasoning benchmark for multimodal models. arXiv:2506.17667.

xAI. 2025. Grok 4. Model release.

Hengrui Xing, Ansaf Salleb-Aouissi, and Nakul Verma. 2021. Automated symbolic law discovery: A computer vision approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 660–668.

Xin Xu, Qiyun Xu, Tong Xiao, Tianhao Chen, Yuchen Yan, Jiaxin Zhang, Shizhe Diao, Can Yang, and Yang Wang. 2025. Ugphysics: A comprehensive benchmark for undergraduate physics reasoning with large language models. In *International Conference on Machine Learning (ICML)*. ICML 2025 poster. arXiv:2502.00334.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. arXiv:2505.09388.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, and 3 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 (CVPR)*, pages 9556–9567.

Hengzhe Zhang, Qi Chen, Bing Xue, Wolfgang Banzhaf, and Mengjie Zhang. 2025a. RAG-SR: Retrieval-augmented generation for neural symbolic regression. In *International Conference on Learning Representations (ICLR)*.

Kaizhong Zhang and Dennis Shasha. 1989. Simple fast algorithms for the editing distance between trees and related problems. *SIAM Journal on Computing*, 18(6):1245–1262.

Xinyu Zhang, Yuxuan Dong, Yanrui Wu, Jiaxing Huang, Chengyou Jia, Basura Fernando, Mike Zheng Shou, Lingling Zhang, and Jun Liu. 2025b. Physreason: A comprehensive benchmark towards physics-based reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16593–16615, Vienna, Austria. 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*.

A Case Study: Bessel Function Recovery

We illustrate ViSA-R2's reasoning on a Bessel function scenario (instance 4_4_0005). The ground-truth solution is:

$$u^*(x, y) = 0.75 \cdot J_0\big(2.05\sqrt{x^2 + y^2}\big)$$

Visual cues. The field heatmap exhibits concentric oscillatory rings centered at the origin with radially decaying amplitude, the hallmark of a zeroth-order Bessel function. The gradient images show azimuthally uniform radial derivatives, confirming exact radial symmetry.

Model reasoning (abridged). The model's CoT identifies: (1) radial symmetry from the circular contour pattern; (2) oscillatory behavior with $\sim$2 zero-crossings in the visible domain, suggesting a frequency parameter $k \approx 1.5$–2.1; (3) a central peak value ≈ 0.75 from the colorbar, giving the amplitude. It hypothesizes $u = A \cdot J_0(kr)$ and estimates parameters from multiple sources.

Predicted expression.

$$\hat{u}(x, y) = 0.75 \cdot J_0\big(1.58\sqrt{x^2 + y^2}\big)$$

The model correctly identifies the solution family (J_0), the radial argument structure ($\sqrt{x^2 + y^2}$), and the amplitude (0.75 exactly). The frequency parameter is estimated at 1.58 vs. the true 2.05, a 23% error that post-hoc L-BFGS-B optimization corrects to within 1%.

This example demonstrates the two-stage workflow: the VLM provides the correct structural ansatz, and numerical optimization recovers precise coefficients. A classical SR method without `besselj` in its operator set cannot produce this form regardless of computational budget.

B Full 150-Sample LLM Evaluation

Table 3 reports LLM-only (text-mode) results on the full 150-sample evaluation set. Models with parse rate below 90% are marked with †; their metrics should be interpreted cautiously as they are computed only on successfully parsed outputs.

The 30-sample subset used in the main paper (Table 1) produces consistent rankings with the full 150-sample evaluation. Models with low parse rates (Gemini-3-Pro: 22.7%; Qwen3-VL-8B: 20.0%) achieve high metrics on their small parsed subset, but these results may not generalize. Grok-4-Fast exhibits extreme NRMSE outliers on the full set due to numerically divergent predictions.

Method	N	TED(family)↑	NRMSE↓
GPT-5.2	150	0.455	0.344
Claude-Opus-4.5	149	0.406	0.386
Claude-Sonnet-4.5	144	0.395	0.670
Claude-Haiku-4-5	143	0.400	0.598
Gemini-Flash	140	0.394	0.440
Grok-4-Fast	138	0.369	36.34
†Gemini-3-Pro	34	0.795	0.072
†Qwen3-VL-8B	30	0.458	0.278

Table 3: LLM-only results on 150 evaluation samples. N: number of successfully parsed outputs. †Parse rate <90%; metrics computed only on parsed subset.

C Relationship to Standard SR Benchmarks

Standard symbolic regression benchmarks, such as SRBench (La Cava et al., 2021), the Feynman equations (Udrescu and Tegmark, 2020), and Nguyen benchmarks, evaluate methods on discovering expressions from unstructured numerical tables across diverse function classes and dimensionalities. Our benchmark, ViSA-Bench, differs in several key respects:

- **Input modality.** ViSA-Bench provides field visualizations (heatmaps and gradient images) as primary input, whereas SR benchmarks provide only numerical point sets. This difference is central to our investigation: we study whether visual structure aids solution-form identification.
- **Domain restriction.** ViSA-Bench is restricted to 2D steady-state scalar fields (predominantly from linear PDEs, with selected nonlinear stationary extensions), which is narrower than SR benchmarks that span arbitrary dimensionality and function classes. This restriction enables exact ground truth from analytic PDE solutions and produces fields with rich visual structure.
- **Function complexity.** ViSA-Bench deliberately includes special-function families (Bessel, Airy, Yukawa) that are absent from most SR operator libraries but common in physics. Standard SR benchmarks typically use elementary functions.

We do not evaluate on standard SR benchmarks because: (1) they do not provide field visualizations, removing the visual modality that is the focus of our study; (2) many SR benchmark expressions

are not solutions to 2D PDEs and thus cannot be rendered as physically meaningful heatmaps; and (3) our structural metric (TED with family grouping) is designed for the function families in our benchmark and would require adaptation for arbitrary expressions. The comparison in our main results (Table 1) evaluates SR methods on our benchmark under matched conditions, providing a fair assessment of complementary strengths.

D Metric Justification

We evaluate with TED(family) and NRMSE rather than the exact recovery rate or R^2 commonly used in symbolic regression benchmarks. We explain the rationale below.

Why not exact recovery rate? Exact recovery (checking whether the predicted expression is symbolically equivalent to the ground truth) is a binary metric: either the expression matches exactly or it does not. In our setting, this metric is too coarse. A prediction that identifies the correct function family but has slightly different coefficients (e.g., $0.48\, J_0(2.1r)$ vs. $0.5\, J_0(2r)$) would score 0 under exact recovery, discarding valuable information about partial structural correctness. TED(family) provides a continuous measure of structural similarity, rewarding predictions that are "close" to the ground truth in tree structure even when not symbolically identical.

Why not R^2? R^2 and MSE-based metrics measure numerical fit on evaluation points but are insensitive to structural correctness. A degree-8 polynomial can achieve $R^2 > 0.99$ on a Bessel function within a bounded domain, yet it captures none of the underlying physics. NRMSE serves a similar purpose to R^2 (measuring numerical accuracy) but is normalized by the range of the ground-truth field, making it comparable across scenarios with different magnitudes.

Why family-level grouping in TED? Raw TED (without family grouping) penalizes substitutions between closely related functions (e.g., $\sin \to \cos$, or $J_0 \to Y_0$) as heavily as unrelated substitutions (e.g., $\sin \to \exp$). Since our focus is on identifying the solution *form* rather than exact function choice within a family, we group functions by family (trig, hyperbolic, Bessel, Airy) so that within-family substitutions incur zero cost. This better reflects whether a model has identified the correct physical mechanism.

Complementarity. TED(family) and NRMSE together capture two orthogonal dimensions: a method may achieve high structural similarity with imprecise coefficients (VLMs), or low numerical error with a structurally different approximation (SR). Reporting both avoids the failure mode of either metric alone.

E Role of Ground Truth in CoT Synthesis

Our CoT synthesis pipeline uses ground-truth access during trace construction, which warrants careful discussion. Stages 3–5 explicitly use s^* to (i) identify which observable signatures to emphasize, (ii) select which visual cues support a coherent derivation, and (iii) ensure the reasoning pathway leads to the correct solution family. The resulting traces are therefore *oracle-guided*: they represent the reasoning a model should produce given the observations, reverse-engineered from knowledge of the answer. The key constraint is that the final trace r^* references only quantities derivable from the observation (I_1, I_2, m); Stage 7 explicitly validates that no ground-truth expression or parameter values appear verbatim in the training data. The ground truth is used to *select and organize* reasoning steps, not to provide answers directly. At inference time, the model receives only observations and must reproduce the reasoning pathway without any oracle access—it has no channel through which ground-truth information could leak.

This design is analogous to teacher-forced training in sequence-to-sequence models, or knowledge distillation where a teacher with privileged information constructs training targets for a student that operates without such access. The CoT-aligned model (TED=0.826) substantially outperforms the zero-shot base model (TED=0.484), demonstrating that this guided supervision is effective for learning the observation-to-solution mapping.

Category	Scenario Name	Operator	Sing.	#Samples
Electrostatics	Point Charge Potential	Laplacian	✓	500
	Electric Dipole Potential	Laplacian	✓	500
	Coulomb Potential Field	Laplacian	×	500
Heat Transfer	Gaussian Heat Source	Laplacian	×	500
	Multi-Gaussian Heat Sources	Laplacian	×	500
	Thermal Boundary Layer	Laplacian	×	500
	Radial Heat Conduction	Laplacian	×	500
	Anisotropic Heat Transfer	Aniso. Laplacian	×	500
	Thermal Conduction Rings	Laplacian	×	500
Fluid Dynamics	Potential Flow Superposition	Laplacian	×	500
	Point Source/Sink Flow	Laplacian	✓	500
	Point Vortex Streamfunction	Laplacian	✓	500
	Doublet Flow Pattern	Laplacian	×	500
Quantum Mechanics	Harmonic Oscillator Eigenmode	Schrödinger	×	500
	Square Well Eigenstate	Schrödinger	×	500
	Helmholtz Wave Function	Helmholtz	×	500
	Bessel Scattering State	Helmholtz	×	500
	Airy Function Solution	Schrödinger (1D)	×	500
Other PDEs	Poisson with Poly. Source	Laplacian	×	500
	Poisson Mixed BCs	Laplacian	×	500
	Helmholtz Plane Wave	Helmholtz	×	500
	Poisson with Two Point Sources	Laplacian	✓	500
	Yukawa Screened Potential	Mod. Helmholtz	×	500
	Mod. Helmholtz Eigenmode	Mod. Helmholtz	×	500
	Helmholtz Standing Wave	Helmholtz	×	500
	Reaction-Diffusion Equilibrium	Nonlinear R-D	×	500
	Fokker-Planck Stationary	Fokker-Planck	×	500
	Line Mass Gravity Potential	Laplacian	✓	500
	Helmholtz Mode Superposition	Helmholtz	×	500
Screened Potentials	Yukawa Potential	Mod. Helmholtz	✓	500
Total				**15,000**

Table 4: Distribution of scenarios in the training dataset by operator type and physical domain.

Stage 1: Visual Feature Observation

Purpose: Extract visual patterns from scalar field and gradient visualizations.
Prompt:

You are analyzing a scalar field $u(x, y)$ and its gradient components over the domain $[\mathrm{x}_{\mathrm{min}}, \mathrm{x}_{\mathrm{max}}, \mathrm{y}_{\mathrm{min}}, \mathrm{y}_{\mathrm{max}}]$.

Task: Carefully observe the provided images and describe all visual features you can identify.

Images provided:

- Scalar field visualization ($u(x, y)$)
- Gradient x-component ($\partial u/\partial x$)
- Gradient y-component ($\partial u/\partial y$)

Instructions:

1. Describe the overall pattern/shape of the scalar field
2. Identify any special features: symmetries, extrema, zeros, singularities, boundaries
3. Read values from the colorbars to estimate:
   - Maximum and minimum values of $u(x, y)$
   - Approximate values of $\partial u/\partial x$ and $\partial u/\partial y$
4. Note any special behaviors: radial patterns, linear gradients, oscillations, decay patterns

Output format: SUMMARY: - Pattern: [concise description]

- Symmetry: [none/radial/translational/rotational/other]

- Max value: [estimate from colorbar]

- Min value: [estimate from colorbar]

- Gradient type: [uniform/varying/radial/other]

- Special features: [list key features]

Figure 5: Stage 1: Visual Feature Observation Prompt for CoT Generation

Stage 2: Numerical Data Analysis

Purpose: Verify visual observations with quantitative evidence from CSV data.
Prompt:

You are analyzing numerical data for a scalar field $u(x, y)$ over domain $[\mathrm{x}_{\min}, \mathrm{x}_{\max}, \mathrm{y}_{\min}, \mathrm{y}_{\max}]$.

Previous analysis (Stage 1): [Visual observations summary]

Task: Analyze the provided CSV data and statistical features to extract quantitative evidence.

Data provided:

- Field CSV data (20×20 grid, 400 points)
- Gradient CSV data (20×20 grid, 400 points)
- Statistical features:
  - domain, grid_shape
  - extrema: max_value, max_location, min_value, min_location
  - radial: radial_symmetry_score, decay_rate, radial_profile
  - symmetry: x_axis, y_axis, rotational_180
  - zero_crossings: has_zero_crossings, crossing_count
  - gradient: mean, std, range, magnitude, uniformity
  - boundary: left, right, top, bottom values

Instructions:

1. Verify the image observations with numerical data
2. Calculate key numerical metrics (range, gradients, extrema locations)
3. Identify numerical patterns (decay rates, oscillation frequencies, radial profiles)
4. Look for quantitative clues about the solution form

Output format:

`NUMERICAL_EVIDENCE:`

- Max: [value] at approximately [location]
- Min: [value] at approximately [location]
- Gradient magnitude: [range]
- Decay/growth rate: [estimate if applicable]
- Oscillation frequency: [estimate if applicable]
- Key ratios: [any useful ratios between quantities]

Figure 6: Stage 2: Numerical Data Analysis Prompt for CoT Generation

Stage 3: Ground Truth Feature Extraction

Purpose: Identify all theoretical features from the analytical solution.
Prompt:

You are analyzing the ground truth solution to identify its theoretical features.

Ground truth solution: $u(x, y) =$ [analytical expression]

Physical/Mathematical Context:

- PDE equation: [e.g., $(\Delta + k^2)\psi = 0$]
- Operator type: [e.g., Helmholtz, Laplacian, Schrödinger]
- Scenario: [e.g., Cylindrically Symmetric Scattering State]

Task: Extract all theoretical features from this analytical solution that could be observable in images or numerical data.

Instructions:

1. Identify the solution family/type (polynomial, exponential, trigonometric, Bessel, etc.)
2. Extract parameters and their meanings
3. List observable features:
   - Symmetries (radial, translational, rotational)
   - Extrema locations and values
   - Boundary behavior
   - Decay/growth rates
   - Oscillation patterns
   - Zero crossings
4. Identify which features are most distinctive and easiest to observe

Output format:

GTFEATURES:

- Solution family: [type]

- Parameters: [list with meanings]

- Observable features:

* Feature 1: [description + how to observe]

* Feature 2: [description + how to observe]

...

- Most distinctive features: [rank top 3]

- Verification signatures: [what would confirm this?]

Figure 7: Stage 3: Ground Truth Feature Extraction Prompt

Stage 4: Feature Matching and Verification

Purpose: Match observed features against theoretical predictions.
Prompt:

You are matching observed features with theoretical predictions to filter noise and confirm patterns.

Task: Systematically compare observed features against theoretical predictions to identify:

1. Which GT features are clearly present in observations
2. Which GT features are ambiguous or hard to confirm
3. Any contradictions that need resolution

Instructions:

1. For each GT feature, check if it appears in Stage 1 or Stage 2 observations
2. Rate the match quality: STRONG MODERATE WEAK ABSENT CONTRADICTORY
3. Identify which parameters can be estimated from which observations
4. Flag any inconsistencies between images and numerical data

Output format:

FEATUREMATCHING:

Confirmed features (STRONG match): - [Feature]: [evidence from Stage 1 2]

Probable features (MODERATE match): - [Feature]: [evidence from Stage 1 2]

Unclear features (WEAK/ABSENT): - [Feature]: [why unclear]

Contradictions: - [Any contradictions and potential resolutions]

PARAMETER OBSERVABILITY:

- Parameter 1: Observable from [source1, source2, ...]

- Parameter 2: Observable from [source1, source2, ...]

Figure 8: Stage 4: Feature Matching and Verification Prompt

Stage 5: Multi-Source Parameter Estimation

Purpose: Estimate solution parameters from multiple independent sources with cross-validation.
Prompt:

You are estimating solution parameters from multiple independent sources.

Task: For each parameter identified in Stage 4, estimate its value from ≥ 2 independent sources.

Instructions:

1. For each parameter, identify 2-3 estimation methods:
   - From colorbar readings
   - From extrema values/locations
   - From gradient magnitudes
   - From decay rates
   - From zero crossings
   - From boundary values
2. Show explicit calculations for each method
3. Compare estimates for consistency
4. Compute weighted average if consistent, or flag conflicts

Example format for one parameter:

Parameter: λ (decay constant)

Method 1 (Colorbar): - Center value u(0,0) $\approx$ 2.5 (from colorbar) - If u = A*exp($-\lambda$*r), and at r=0: A $\approx$ 2.5 - At r=3, u $\approx$ 0.5 (from colorbar) - 0.5 = 2.5*exp($-\lambda$*3) $\rightarrow$ $\lambda \approx 0.54$

Method 2 (Gradient): [similar detailed calculation]

Method 3 (Numerical): [similar detailed calculation]

Consistency: Methods agree within 8% Final estimate: $\lambda \approx 0.54 \pm 0.03$

Output format:

PARAMETER ESTIMATES:

Parameter 1: [value] and [uncertainty] (sources: [list], consistency: [%])

Parameter 2: [value] and [uncertainty] (sources: [list], consistency: [%])

Figure 9: Stage 5: Multi-Source Parameter Estimation Prompt

Stage 6: Chain-of-Thought Generation

Purpose: Generate natural reasoning from observations to solution.
Prompt:

You are generating a Chain-of-Thought (CoT) reasoning process for predicting a PDE solution.

Ground Truth Solution: [for reference, NOT to be mentioned in CoT]

Previous Stages Summary: [Stage 1-5 summaries provided]

Task: Generate a natural, logical Chain-of-Thought that:

1. Starts from observations (what you see in images/data)
2. Identifies patterns and makes hypotheses about solution type
3. Estimates parameters through explicit calculations
4. Arrives at the final solution
5. Verifies the solution makes sense

Critical requirements:

- Use natural language (like human reasoning, not JSON)
- Show explicit arithmetic calculations
- Use multi-source parameter verification
- DO NOT say "the ground truth is..." or "comparing with GT..."
- Make it seem like independent reasoning from observations
- The CoT should lead naturally to the GT solution

Style reference:

```
<thinking>
Looking at the scalar field image, I observe a strong radial symmetry centered at the
origin. The colorbar shows the field decays from approximately 2.5 at the center to near
0 at the boundaries.

Let me estimate parameters:
1. From colorbar: center value A ≈ 2.5
2. From decay: at r≈3, u≈0.5, so 0.5 = 2.5*exp(-3λ) → λ ≈ 0.536
3. Verification from gradient: ...
[continue reasoning with calculations]
</thinking>

<solution>[final SymPy expression]</solution>
```

The CoT should be 300–800 words, showing detailed step-by-step reasoning.

Figure 10: Stage 6: Chain-of-Thought Generation Prompt

Test Prompt: Direct Symbolic Regression from Field Data

Purpose: Evaluate model's ability to derive symbolic expressions from PDE solution data.
Prompt:

You are an expert in analyzing scientific field data and deriving symbolic mathematical expressions.

Input Data

1. Scalar Field Visualization

The first image shows the scalar field $u(x, y)$ as a heatmap.

2. Gradient Components Visualization

The second image shows the gradient components $\partial u/\partial x$ and $\partial u/\partial y$.

3. Field Data (CSV)

```
Data shape: (400, 3)
Columns: x, y, u
Value ranges:
  x: [xmin, xmax]
  y: [ymin, ymax]
  u: [umin, umax]
First 10 rows:
  x      y      u
[data rows...]
```

4. Gradient Data (CSV)

```
Data shape: (400, 4)
Columns: x, y, du_dx, du_dy
Value ranges:
  x: [xmin, xmax]
  y: [ymin, ymax]
  du_dx: [grad_x_min, grad_x_max]
  du_dy: [grad_y_min, grad_y_max]
First 10 rows:
  x      y     du_dx    du_dy
[data rows...]
```

Figure 11: Test/Inference Prompt for Symbolic Regression Evaluation