Table2LaTeX-RL: High-Fidelity LaTeX Code Generation from Table Images via Reinforced Multimodal Language Models

Jun Ling[1] cs.lingjun@gmail.com **Yao Qi**[2] **Tao Huang**[1] taohuang0313@gmail.com **Shibo Zhou**[2]

Yanqin Huang[2] **Jiang Yang**[2] **Ziqi Song**[2] **Ying Zhou**[2] **Yang Yang**[1]

Heng Tao Shen[1,3] **Peng Wang**[1]* p.wang6@hotmail.com

[1]School of Computer Science and Engineering,
University of Electronic Science and Technology of China
[2]Research Center for Scientific Data Hub, Zhejiang Lab, Hangzhou, China
[3]School of Computer Science and Technology, Tongji University

Abstract

In this work, we address the task of table image to LaTeX code generation, with the goal of automating the reconstruction of high-quality, publication-ready tables from visual inputs. A central challenge of this task lies in accurately handling complex tables—those with large sizes, deeply nested structures, and semantically rich or irregular cell content—where existing methods often fail. We begin with a comprehensive analysis, identifying key challenges and highlighting the limitations of current evaluation protocols. To overcome these issues, we propose a reinforced multimodal large language model (MLLM) framework, where a pre-trained MLLM is fine-tuned on a large-scale table-to-LaTeX dataset. To further improve generation quality, we introduce a dual-reward reinforcement learning strategy based on Group Relative Policy Optimization (GRPO). Unlike standard approaches that optimize purely over text outputs, our method incorporates both a structure-level reward on LaTeX code and a visual fidelity reward computed from rendered outputs, enabling direct optimization of the visual output quality. We adopt a hybrid evaluation protocol combining TEDS-Structure and CW-SSIM, and show that our method achieves state-of-the-art performance, particularly on structurally complex tables, demonstrating the effectiveness and robustness of our approach. Code and dataset are available at https://github.com/newLLing/Table2LaTeX-RL.

1 Introduction

Tables are essential components of scientific and technical documents, providing a structured and concise format for presenting quantitative data, experimental results, and complex relationships. As document digitization becomes increasingly prevalent, the ability to automatically generate table code from images is critical for enabling content reuse and high-quality reproduction. However, most existing methods focus on generating HTML representations [1–5] , which lack the structural

*Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

expressiveness and typographic precision required for complex tables—especially those with nested headers, merged cells, or mathematical content. In contrast, LaTeX is the standard in scientific publishing, offering the flexibility and fidelity needed for professional-grade tables. Despite its practical importance, the task of directly generating LaTeX code from table images has received limited attention in prior work [6, 7].

In this work, we study the task of table image to LaTeX generation and provide a comprehensive analysis of its challenges. Through empirical observations, we find that the primary difficulty lies in handling complex tables, which are often large, deeply nested, and semantically rich—structures naturally suited to LaTeX but difficult for models to predict accurately. These challenges affect both the vision encoder, which must extract fine-grained visual and structural cues, and the language decoder, which must generate long, syntax-sensitive LaTeX sequences. Errors in either stage often lead to hallucinated, malformed output or even compilation errors. To enable finer-grained evaluation and better understand the current research gaps, we propose splitting the dataset into simple, medium, and complex subsets based on structural complexity.

To tackle these challenges, we leverage pre-trained multimodal large language models (MLLMs), which demonstrate strong capabilities in visual recognition, cross-modal reasoning, and LaTeX fluency. We fine-tune an MLLM on a large-scale image-to-LaTeX dataset harvested from scientific documents on arXiv. To further improve performance—particularly for complex tables—we introduce a dual-reward reinforcement learning strategy built on Group Relative Policy Optimization (GRPO) [8], termed VSGRPO. While standard GRPO methods optimize text generation quality based solely on textual output, we go a step further: we render the generated LaTeX code into images and directly evaluate visual fidelity using CW-SSIM. This image-based reward complements a structure-level reward computed from the LaTeX source, allowing us to jointly optimize for both structural accuracy and rendered appearance. This novel visual-in-the-loop reinforcement design significantly enhances the model's ability to produce faithful, high-fidelity LaTeX code for structurally rich and visually complex tables.

From an evaluation perspective, existing metrics are limited. TEDS [9], a widely used structure-based metric, lacks sensitivity to fine-grained errors and suffers from mismatches between HTML and LaTeX. On the other hand, rendered image comparison metrics focus on local visual similarity but ignore global structural correctness. To overcome this, we adopt a hybrid evaluation strategy that combines TEDS-Structure [10] for structural fidelity and CW-SSIM for robust visual similarity.

Under this framework, our method achieves state-of-the-art performance on the table image to LaTeX generation task, with particularly strong improvements on complex tables. This demonstrates the effectiveness of combining MLLM fine-tuning with targeted reinforcement learning for high-fidelity, publication-ready table generation.

- We delve deep into the under-explored task of table image to LaTeX code generation, offering a comprehensive analysis of its core challenges—particularly for structurally complex tables—and introducing a complexity-based data split for fine-grained evaluation.
- We develop a reinforced MLLM framework, where a pre-trained MLLM is fine-tuned on a large-scale image-to-LaTeX dataset harvested from arXiv, effectively bridging visual input and LaTeX code generation.
- We propose VSGRPO, a novel dual-reward reinforcement learning strategy based on GRPO, which jointly optimizes structure-level accuracy and visual fidelity by incorporating both LaTeX-based and rendered-image-based feedback.
- We introduce a hybrid evaluation strategy combining TEDS-Structure and CW-SSIM to better reflect both structural and visual correctness. Extensive experiments demonstrate state-of-the-art performance of our approach, especially on complex tables.

2 Related Work

Table Structure Recognition. Existing table recognition approaches fall into two main categories: detection-based and image-to-text–based methods. Detection-based methods first predict the physical structure—such as grid lines or cell bounding boxes—and then infer logical relationships. Grid-line-based approaches [11–17] segment tables along detected rows and columns and merge regions to

reconstruct cells. Cell-bounding-box methods [18–21] treat detected cells as graph nodes, using GNNs to infer row/column associations.

Image-to-text–based table structure recognition (TSR) decomposes the task into predicting structural layout and transcribing cell content, which are then fused into a full table representation. Encoder–decoder models [1–3] generate structure tokens (e.g., HTML tags) and content separately. TableFormer [3] improves this with Transformer-based decoding and regression for bounding boxes. VAST [4] frames coordinate prediction as sequence generation and adds a visual alignment loss. DRCC [5] adopts a hybrid decoding scheme to reduce error accumulation.

Most detection and TSR methods target HTML outputs, which are not well-suited for LaTeX due to syntactic and semantic differences. Recently, end-to-end LaTeX generation approaches have emerged. LaTeXNet [7] uses specialized submodules for equations, tables, and text. Nougat [22] bypasses OCR entirely to generate LaTeX directly. LATTE [6] introduces iterative refinement via localization and correction models. However, these methods do not explicitly address the combined challenges of large-scale layout and deeply nested LaTeX structures.

Multimodal Large Language Models with Reinforced Fine-Tuning. Pre-trained multimodal large language models (MLLMs) learn joint visual–text representations from large-scale image–text corpora, equipping them with strong capabilities in visual understanding and LaTeX code generation. While recent works such as Nougat [22] and LATTE [6] employ multimodal architectures, they largely underutilize pre-trained priors, relying instead on from-scratch training.

To further improve performance, especially for complex tables, we apply reinforced fine-tuning using the Group Relative Policy Optimization (GRPO) framework [8]. Compared to earlier reinforcement methods such as RLHF[23] and DPO[24], GRPO eliminates the need for a value network and uses correctness-based rewards to guide learning with reduced computational overhead. Unlike prior works that apply reinforcement learning purely in the text domain [25, 26], our method designs task-specific reward signals: we compile the generated LaTeX into HTML for TEDS-Structure evaluation and into images for CW-SSIM computation, enabling joint optimization of both structural accuracy and rendered visual fidelity. This visual-in-the-loop RL approach is particularly effective for high-fidelity LaTeX generation on complex table structures.

3 Insight into the Task

We provide key insights into the task of table-to-LaTeX generation, focusing on two critical aspects: the challenge of handling complex tables and the limitations of current evaluation protocols.

One of the central challenges in this task lies in accurately processing complex tables, which serve as a meaningful indicator of a model's true capability. Complex tables are prevalent in modern documents, often used to convey large volumes of structured information compactly. However, their intricate layouts, large dimensions, and diverse content introduce significant difficulties for vision encoders—leading to higher computational costs, reduced performance, and increased inference latency. Despite their importance, complex tables are often underrepresented or overlooked in evaluation. To address this, we propose categorizing tables into three complexity levels—simple, medium, and complex—to enable a more realistic and fine-grained evaluation of model performance.

In addition to data-level challenges, the evaluation of LaTeX code generation remains underdeveloped, as shown in Appendix A. Existing metrics generally fall into two categories: **Text-based metrics**, such as TEDS [9] and BLEU [27] , compare the predicted and ground-truth LaTeX code at the token level. However, they fail to account for the inherent syntactic ambiguity of LaTeX and ignore structural semantics, often penalizing semantically equivalent but syntactically different outputs. **Visual-based metrics**, such as CW-SSIM [28], evaluate similarity between rendered images of the generated and ground-truth tables. While useful for natural scenes, standard CW-SSIM is less effective on binary, high-contrast table images, where sharp edges and sparse textures dominate. Alternative pixel-level metrics, such as Edit (column-wise normalized edit distance) and Match (binary pixel-wise agreement), also fall short in capturing higher-level structural or semantic similarity. To address these limitations, we adopt a modified version of CW-SSIM, tuned for binary table images with high visual sparsity, to better assess rendering fidelity. However, since CW-SSIM primarily focuses on local visual similarity, we complement it with TEDS-Structure [10] to evaluate the global structure and layout alignment.

4 Method

4.1 Large-Scale Table2LaTeX Collection

Due to the lack of publicly available large-scale datasets containing LaTeX table code, we propose a dataset construction pipeline. Specifically, we develop a web crawler to scrape the LaTeX source files of scientific papers from the open-access arXiv repository. We use regular expressions to extract LaTeX code corresponding to table environments. To ensure data quality, we further clean the extracted code by removing references, color settings, and other LaTeX control commands. Through this process, we collect a dataset comprising 1,209,986 table–LaTeX pairs.To classify table complexity, tables with 2 or more `\multirow` or `\multicolumn` commands and 100–160 cells are defined as medium tables, while those with over 160 cells are labeled complex tables. All others are considered simple. Within the training set, simple tables account for approximately 94%, while medium and complex tables each represent about 3% of the data.

4.2 Supervised Fine-Tuning

To enable general multimodal large language models (MLLMs) to acquire preliminary capability for handling the task of table-to-LaTeX generation, we design a second-stage supervised fine-tuning (SFT) process. During this process, we perform SFT using data collected in stage 1. The input consists of a table image and the prompt: ‘‘`Convert this table to LaTeX`’’, while the ground-truth LaTeX code serves as the response. Thus, our dataset is structured as input-response pairs, formally expressed as: $\mathcal{D} = \left\{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\right\}_{i=1}^{N}$ where each $\mathbf{x}^{(i)}$ is an input and $\mathbf{y}^{(i)}$ the corresponding target response. During training we optimize θ to maximize the conditional likelihood of $\mathbf{y}^{(i)}$ given $\mathbf{x}^{(i)}$. Equivalently, we minimize the negative log-likelihood over the dataset:

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\sum_{i=1}^{N} \log p_\theta\big(\mathbf{y}^{(i)} \mid \mathbf{x}^{(i)}\big). \tag{1}$$

However, as shown in Table 4, SFT alone is insufficient to fully unlock the model's potential. A key limitation stems from the widespread use of teacher forcing, where the model is trained to predict the next token given the prefix. Yet, LaTeX code is inherently ambiguous—different syntactic forms (e.g., control sequences) may yield identical visual outputs. This mismatch between training supervision and evaluation objectives leads to inefficient generalization, particularly for structurally complex tables.

4.3 Reinforced Fine-Tuning via VSGRPO

As analyzed above, the next-token prediction paradigm used in SFT is limited in its ability to model the semantic structure and syntactic dependencies embedded in long LaTeX sequences. Moreover, the SFT objective focuses solely on text-level alignment and completely ignores the visual similarity between the rendered LaTeX output and the original table image—despite visual appearance being a direct and critical indicator of generation quality. However, since LaTeX rendering is a non-differentiable operation, it cannot be directly incorporated into gradient-based supervised training.

To address these limitations, we propose a novel reinforced fine-tuning framework that introduces rendered image feedback as an explicit optimization signal. Drawing inspiration from Group Relative Policy Optimization (GRPO) [8], we extend its scope beyond standard textual quality assessment and design a dual-reward mechanism that jointly promotes structural accuracy and visual fidelity. While conventional GRPO-based methods focus solely on improving text generation quality, our framework leverages both the LaTeX code structure and its rendered appearance, offering a more task-aligned supervision signal, as shown in Figure 1. We select 5,936 complex tables from the training dataset as the training set for VSGRPO, whose ground-truth LaTeX code contains fewer than 3,000 characters to balance complexity and computational feasibility.

Visual Reward. We compile and convert the set of predicted table LaTeX code—generated by the model from a single table image input—by embedding them into LaTeX fragments with standard macro packages, producing a group of predicted table images. At the same time, we compile the ground-truth LaTeX code to obtain the ground-truth table image. If the compilation fails, the

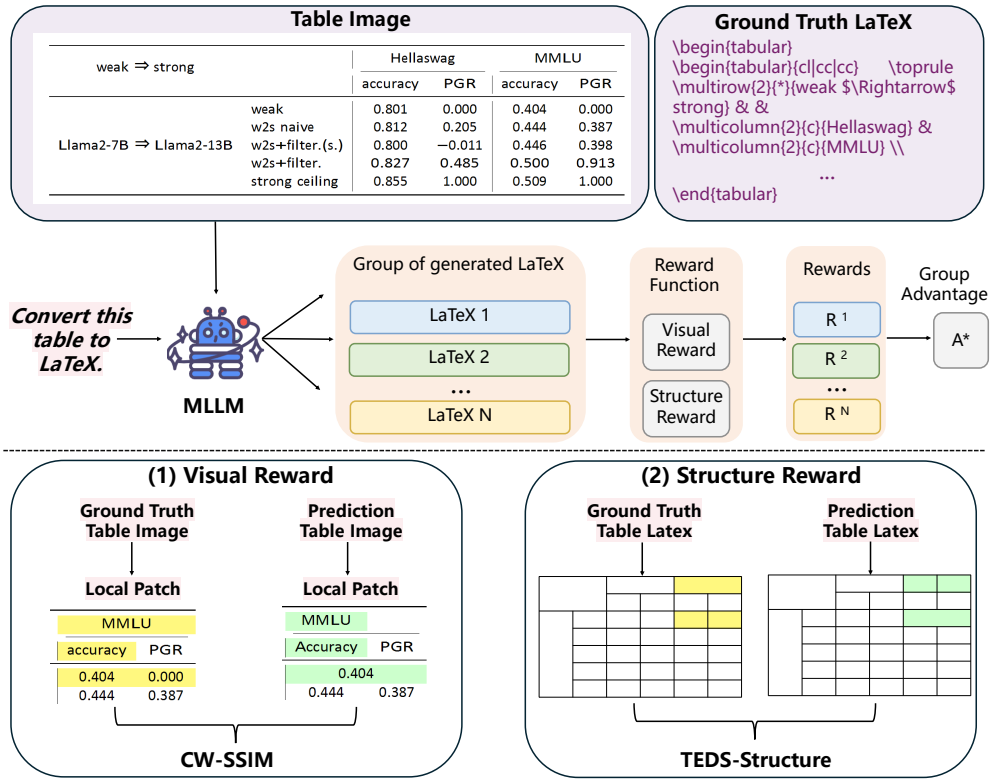


Figure 1: Demonstration of our proposed **VSGRPO** framework for table image to LaTeX code generation. The top section shows an example table image alongside its corresponding LaTeX code, representing the input-output pair used in training. The middle section illustrates the workflow of the VSGRPO framework. The bottom section highlights the dual-reward mechanism: a visual fidelity reward computed via CW-SSIM between the model-generated and ground-truth rendered images, and a structure-level reward based on TEDS-Structure computed from the table's structural elements.

corresponding reward is set to 0. We then compute the CW-SSIM between the ground-truth table image and each predicted table image. If the CW-SSIM of a predicted image exceeds the predefined threshold, it receives a reward of 1; otherwise, the reward is 0.

To accommodate black-and-white table images, we adopt the following CW-SSIM calculation process: the CW-SSIM algorithm preprocesses two table images by converting them to grayscale, resizing them to uniform dimensions, and aligning their rows and columns. It then divides the images into 2×2 pixel blocks and applies a simplified Haar wavelet transform [29] to decompose each block into four sub-bands: cA (low-frequency approximation), cH (horizontal), cV (vertical), and cD (diagonal high-frequency details). For each sub-band, the algorithm calculates SSIM metrics optimized for monochrome tables, incorporating pixel-level means, variances, covariance, and stabilizing constants C_1 and C_2. Finally, it averages the SSIM scores from all four sub-bands to generate the comprehensive CW-SSIM metric.

Structure Reward. We convert both the predicted table LaTeX code generated by the model and the ground-truth LaTeX into HTML in order to compute their TEDS-Structure. If the HTML conversion of a predicted table LaTeX fails, the reward is set to 0. For successfully converted predictions, if the TEDS-Structure similarity with the ground-truth exceeds a predefined threshold, the reward is set to 1; otherwise, it is 0.

TEDS-Structure computes the Minimum Tree Edit Distance between the two trees by applying unit-cost insertions, deletions, and structural-node substitutions to transform the predicted tree into the ground-truth tree. It then normalizes this distance by the larger of the two tree sizes and converts it into a similarity score.

During the RFT training process, we samples a group of generated output set $\{o_1, o_2, \cdots, o_N\}$ for each table image q from policy model $\pi_{\theta_{old}}$. Then RFT maximizes the following objective and optimizes the model π_θ. The specific formula is shown below:

$$J_{\mathrm{RFT}}(\theta) = \mathbb{E}_{q\sim P(Q),\,\{o_i\}_{i=1}^{N}\sim\pi_{\theta_{\mathrm{old}}}(O|q)} \left[\frac{1}{N}\sum_{i=1}^{N} \min\Big(\frac{\pi_\theta(o_i \mid q)}{\pi_{\theta_{\mathrm{old}}}(o_i \mid q)}\,A_i,\ \mathrm{clip}\Big(\frac{\pi_\theta(o_i \mid q)}{\pi_{\theta_{\mathrm{old}}}(o_i \mid q)}, 1-\varepsilon,\ 1+\varepsilon\Big)A_i\Big) - \beta\, D_{\mathrm{KL}}\big(\pi_\theta \,\big\|\, \pi_{\mathrm{ref}}\big)\right], \tag{2}$$

where ε and β denote the clipping threshold in Proximal Policy Optimization (PPO) and the coefficient controlling the Kullback–Leibler (KL) divergence penalty term, respectively [30, 31]. We set $\varepsilon = 0.2$ and $\beta = 0.02$ during training.

The advantage for the i-th sample is computed as

$$A_i = \frac{r_i - \mathrm{mean}\big(\{r_1, r_2, \ldots, r_N\}\big)}{\mathrm{std}\big(\{r_1, r_2, \ldots, r_N\}\big)}, \tag{3}$$

where $\{r_1, r_2, \ldots, r_N\}$ denotes the set of group rewards. The KL divergence between the current policy π_θ and the reference policy π_{ref} for the observation–action pair (q, o_i) is defined as

$$D_{\mathrm{KL}}\big(\pi_\theta \,\|\, \pi_{\mathrm{ref}}\big) = \frac{\pi_{\mathrm{ref}}(o_i \mid q)}{\pi_\theta(o_i \mid q)} - \log\Big(\frac{\pi_{\mathrm{ref}}(o_i \mid q)}{\pi_\theta(o_i \mid q)}\Big) - 1\,. \tag{4}$$

5 Experiments

In this section, we present our experimental results. Specifically, Section 5.1 details the datasets, implementation settings, and evaluation metrics used in our study. Section 5.2 reports quantitative comparisons against state-of-the-art baselines. To further assess whether the generated table images align with human perception, we conduct a human evaluation, presented in Section 5.3. Finally, Section 5.4 provides an ablation analysis to highlight the key components and contributions of our proposed method.

5.1 Experimental Setup

In this section, we first describe the detailed construction process and composition of the training and testing datasets. We then present the implementation details for both the SFT and reinforced fine-tuning (RFT) phases.

Training Dataset. We collect LaTeX source code from arXiv papers published between October 2017 and April 2023, extracting a total of 1,209,986 table entries using regular expression matches between `\begin{tabular}` and `\end{tabular}`. After filtering out references, color commands, and other non-structural LaTeX elements, we classify the tables into three categories. This full dataset is used for SFT.

Testing Dataset. We construct the testing dataset using the same processing pipeline as the training set. Specifically, we crawl 101,469 LaTeX table entries from arXiv papers published between January and November 2024, covering a diverse range of scientific domains. From this pool, we randomly sample 496 simple, 354 medium, and 361 complex tables to form the final testing set.

Implementation Details. We adopt full-parameter fine-tuning for all training phases. During SFT, all models are trained for one epoch with a maximum output length of 4096 tokens. For Nougat [22], training is conducted on 4 nodes (each with 8×A100 GPUs) using a batch size of 2. InternVL2-1B is trained on 4 nodes with a batch size of 4 and gradient accumulation steps set to 2. Qwen2.5-VL-3B [32] is trained on 2 nodes, also with a batch size of 4 and gradient accumulation steps of 2.

For reinforced fine-tuning (RFT), InternVL2-1B adopts the VLM-R1 framework [33] and is trained on 2 nodes with 8 sampled generations per input (num_gens = 8), a batch size of 8, and gradient

accumulation steps of 2. Qwen2.5-VL-3B uses the ms-swift infrastructure [34], trained on 2 nodes with 4 generations per input (num_gens = 4), a batch size of 4, and the same gradient step setting. The reward thresholds are set to 0.6 for CW-SSIM and 0.9 for TEDS-Structure.

During testing, we use a maximum output length of 8192 tokens, a batch size of 1, and a temperature of 0. All testing is conducted within a `texlive-full` Docker environment to ensure LaTeX rendering fidelity. For metrics, we use Python-based implementations of CW-SSIM, TEDS-Structure, and TEDS for performance evaluation. The scores range from 0 to 1, and the exact formulas are shown in Appendix B.

5.2 Main Results

We compare VSGRPO with various solutions across different categories. In the commercial and paid domain, we evaluate it against the most powerful system to date, Mathpix [35]. To compare with current general-purpose multimodal large models, we include the closed-source GPT-4o [36], as well as the open-source Qwen2.5-VL-72B [37] and Intern2.5-VL-78B [38]. For specialized expert models, we compare against Nougat [22], a state-of-the-art open-source LaTeX generation system.

To more accurately evaluate the correctness of LaTeX generation, we assess model performance from two complementary perspectives: rendered image quality and LaTeX source fidelity. First, we evaluate the visual accuracy of the generated LaTeX by compiling it into table images. Two metrics are used: the compile ratio, which reflects the proportion of LaTeX outputs that can be successfully compiled using standard LaTeX packages, and CW-SSIM, which quantifies the visual similarity between the rendered output and the ground-truth image. These results are reported in Table 1. Second, we assess the semantic and structural correctness of the LaTeX source code itself. To this end, we compute TEDS-Structure, which measures cell-level structural alignment, and TEDS, which additionally considers the tabular content. These metrics provide a deeper view into how well the generated code captures the underlying table semantics, and are summarized in Table 2. To further evaluate the generalization ability of our method, we additionally test it on an external benchmark dataset introduced in [6], with the results shown in Table 3.

Table 1 shows that the CW-SSIM values of all models exhibit a decreasing trend as table complexity increases (from simple to complex). However, the proposed VSGRPO method achieves comprehensive improvements across two model families. Specifically, Intern2-VL-1B-VSGRPO sets a new record on the simple tables with a CW-SSIM of 0.8201, surpassing the previous best model by 0.049. Meanwhile, Qwen2.5-VL-3B-VSGRPO significantly outperforms baselines on the medium tables and complex tables, achieving CW-SSIM scores of 0.7236 (+0.1113) and 0.6145 (+0.0903), respectively. Furthermore, it attains a compile success rate of 0.9917 on the complex tables, exceeding Mathpix's 0.9889. These results demonstrate that the proposed VSGRPO strategy effectively enhances the robustness of complex tables reconstruction while maintaining high LaTeX compilability through visual- and structure-guided optimization.

Table 1: Model performance on CW-SSIM and compile ratio across three table complexity levels.

Models	Simple		Medium		Complex	
	CW-SSIM	Compile ratio	CW-SSIM	Compile ratio	CW-SSIM	Compile ratio
Commercial Tools						
Mathpix [35]	0.6884	**1.0000**	0.5647	0.9943	0.4862	0.9889
General VLMs						
GPT4o [36]	0.6792	0.9918	0.5612	**0.9972**	0.4747	0.9917
Qwen2.5-VL-72B [37]	0.7077	0.9858	0.6009	0.9887	0.5112	0.9335
Intern2.5-VL-78B [38]	0.7814	0.9959	0.6123	0.9773	0.5242	0.4515
Expert VLMs						
Nougat [22]	0.7401	0.7617	0.5505	0.1813	0.4699	0.3352
Our Results						
Intern2-VL-1B-VSGRPO	**0.8201**	0.9939	0.7185	0.9830	0.5899	0.9640
Qwen2.5-VL-3B-VSGRPO	0.8186	0.9980	**0.7236**	0.9943	**0.6145**	**0.9917**

Table 2 presents results for TEDS and TEDS-Structure metrics. The trend of TEDS scores largely mirrors that of TEDS-Structure, although the absolute values are consistently lower due to TEDS additionally accounting for cell content alignment. The commercial tool Mathpix demonstrates relatively stable performance across table types, achieving its highest TEDS-Structure score on

medium-complexity tables (0.8965). In the general-purpose VLM category, Qwen2.5-VL-72B shows consistently strong structural performance, with the highest TEDS-Structure score on simple tables (0.9400). However, it exhibits a gradual performance decline as complexity increases—TEDS drops from 0.8720 (simple) to 0.8090 (medium) and 0.7448 (complex). By contrast, other large-scale models such as Intern2.5-VL-78B experience a sharp drop on complex tables (TEDS: 0.3379), and the expert model Nougat collapses almost entirely (TEDS: 0.0424), revealing severe limitations in both structural and content-level generalization. In contrast, our proposed Qwen2.5-VL-3B-VSGRPO achieves consistently superior results across all levels of table complexity. Despite its compact size (3B parameters), it outperforms significantly larger models, reaching a TEDS score of 0.8673 on complex tables—0.1225 higher than the next-best model—and achieving a TEDS-Structure score of 0.9218, the first to surpass the 0.9 threshold on complex tables. These results underscore the effectiveness of our dual-reward optimization strategy, which integrates structural and visual supervision to enable robust, high-fidelity LaTeX code generation, especially for structurally rich and visually complex tables.

Table 2: Performance of different models on TEDS and TEDS-Structure across three table complexity levels.

Models	Simple		Medium		Complex	
	TEDS	TEDS-Structure	TEDS	TEDS-Structure	TEDS	TEDS-Structure
Commercial Tools						
Mathpix [35]	0.7804	0.8701	0.8044	0.8965	0.7176	0.8100
General VLMs						
GPT4o [36]	0.8259	0.9117	0.6986	0.8451	0.5865	0.7745
Qwen2.5-VL-72B [37]	0.8720	0.9400	0.8090	0.8920	0.7448	0.8334
Intern2.5-VL-78B [38]	0.8368	0.8795	0.7123	0.7652	0.3379	0.3735
Expert VLMs						
Nougat [22]	0.3856	0.4308	0.1193	0.1357	0.0424	0.0527
Our Results						
Intern2-VL-1B-VSGRPO	0.8959	0.9358	0.8604	0.8988	0.8054	0.8625
Qwen2.5-VL-3B-VSGRPO	**0.8997**	**0.9405**	**0.9004**	**0.9427**	**0.8673**	**0.9218**

To evaluate the generalization capability of our method, we conduct additional experiments on an external benchmark dataset introduced in [6]. The results are presented in Table 3. Manual inspection reveals that this dataset is primarily composed of simple tables with limited structural complexity. Consequently, the performance trends largely mirror those observed in the simple-table subsets reported in Table 1 and Table 2.

Once again, our method, Qwen2.5-VL-3B-VSGRPO, achieves superior performance, outperforming both task-specific baselines for table image to LaTeX generation[2] and general-purpose multimodal large language models. These results underscore the model's strong generalization capability.

Table 3: Experimental comparison on external dataset [6] on CW-SSIM and TEDS-Structure.

Models	CW-SSIM	TEDS-Structure
LATTE [6]	0.7615	0.9445
GPT4o [36]	0.6897	0.8568
Qwen2.5-VL-72B [37]	0.7176	0.8915
Intern2.5-VL-78B [38]	0.7696	0.9009
Qwen2.5-VL-3B-VSGRPO	**0.8225**	**0.9461**

5.3 Human Evaluation

To complement automated metrics and better capture perceived visual quality, we conduct a human preference study on 200 randomly selected tables (50 simple, 50 medium, 100 complex), as shown in Appendix C. For each case, rendered outputs from four models are displayed anonymously alongside the ground-truth image. Multiple human assessors independently vote on the most visually similar result, and the final decision is determined by majority voting. As shown in Table 4, Qwen2.5-VL-3B-VSGRPO receives the highest number of votes across all difficulty levels, clearly outperforming other models in terms of visual and structural fidelity.

[2]The LATTE model proposed in [6] is not publicly available. We compute results based on the authors' released outputs and apply our own metric calculations.

Table 4: Results of human evaluation.

Models	Simple	Medium	Complex
GPT4o	5	2	2
Mathpix	19	2	10
Qwen2.5-VL-3B-SFT	29	28	56
Qwen2.5-VL-3B-VSGRPO	**42**	**37**	**70**

5.4 Ablation Study

To validate the effectiveness and robustness of our proposed method, we conduct a series of ablation studies focusing on three aspects: the impact of data selection strategies, the contribution of individual reward components, and the necessity of staged training. All experiments are evaluated on the complex table subset.

Evaluation on the Dataset Selection Strategy for VSGRPO. As shown in Table 5, we compare different strategies for constructing the reinforcement learning (RL) training set. Specifically, we evaluate three variants of Qwen2.5-VL-3B fine-tuned with VSGRPO: (1) using only simple tables (`-Simple`), (2) using a balanced mixture of simple, medium, and complex tables (`-Mixed-Data`), and (3) using only complex tables (`-VSGRPO`). The results clearly demonstrate that restricting the RL fine-tuning data to complex tables leads to the best overall performance across all metrics. This validates our design choice of focusing on structurally difficult examples during reinforcement learning to better generalize across complexity levels.

Table 5: Ablation experiments on the dataset selection for VSGRPO.

Models	CW-SSIM	Compile ratio	TEDS	TEDS-Structure
Qwen2.5-VL-3B-VSGRPO-Simple	0.5993	0.9861	0.8614	0.9113
Qwen2.5-VL-3B-VSGRPO-Mixed-Data	0.6107	0.9861	0.8614	0.9136
Qwen2.5-VL-3B-VSGRPO	**0.6145**	**0.9917**	**0.8673**	**0.9218**

Evaluation on the Reward Design in VSGRPO. Table 6 presents the effectiveness of the two reward components used in our RL framework—TEDS-Structure and CW-SSIM. Adding either reward individually to the base model leads to noticeable performance gains over the SFT-only baseline, demonstrating that both structure-level accuracy and visual similarity are important for improving LaTeX generation quality. The best performance is achieved when both reward signals are combined, suggesting they are complementary in guiding the model toward faithful and well-aligned outputs.

Table 6: Ablation experiments on the reward design.

Models	CW-SSIM	Compile ratio	TEDS	TEDS-Structure
Qwen2.5-VL-3B-SFT	0.5806	0.9889	0.8481	0.9047
Qwen2.5-VL-3B-GRPO-TEDS-Structure	0.5925	0.9889	0.8608	0.9155
Qwen2.5-VL-3B-GRPO-CW-SSIM	0.6064	0.9889	0.8607	0.9133
Qwen2.5-VL-3B-VSGRPO	**0.6145**	**0.9917**	**0.8673**	**0.9218**

Evaluation on the Necessicity of SFT. To verify the necessity of SFT before reinforcement fine-tuning, we perform one epoch of reinforcement learning directly on the pre-trained Qwen2.5-VL-3B model (VSGRPO without SFT). As shown in Table 7, the performance of the model without SFT initialization is significantly lower across all metrics, indicating that SFT is essential to provide a reasonable starting point for subsequent RL-based optimization.

6 Conclusion and Limitations

Our work tackled the challenge of converting table images into syntactically correct, publication-quality LaTeX code by integrating vision-language modeling with targeted reinforcement learning. We leveraged a pre-trained multimodal large language model (MLLM), fine-tuned it on a diverse corpus of scientific table images, and further enhanced it through a dual-reward scheme: one reward evaluated structural integrity using TEDS-Structure, while the other measured visual fidelity via a refined CW-SSIM on rendered outputs. By jointly optimizing these objectives, the model was able to accurately capture complex table layouts—including nested headers, merged cells, and mathematical expressions—and produce outputs that closely matched the original visual appearance.

Table 7: Ablation experiments on the effectiveness of SFT.

Models	CW-SSIM	Compile ratio	TEDS	TEDS-Structure
Qwen2.5-VL-3B-VSGRPO w/o SFT	0.4695	0.9668	0.6884	0.8167
Qwen2.5-VL-3B-VSGRPO	**0.6145**	**0.9917**	**0.8673**	**0.9218**

Limitations. Although VSGRPO effectively improved MLLM performance on complex tables, it introduced notable computational overhead during training. Specifically, each LaTeX output had to be rendered into a PDF and then converted to a PNG image for CW-SSIM computation—a time-consuming process that created a training bottleneck, even with multi-threading. Due to this overhead and limited GPU resources, we trained VSGRPO on only 5,936 complex tables.

References

[1] Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. Image-based table recognition: data, model, and evaluation. In *ECCV*. Springer, 2020.

[2] Jiaquan Ye, Xianbiao Qi, Yelin He, Yihao Chen, Dengyi Gu, Peng Gao, and Rong Xiao. Pingan-vcgroup's solution for icdar 2021 competition on scientific literature parsing task b: table recognition to html. *arXiv*, 2021.

[3] Ahmed Nassar, Nikolaos Livathinos, Maksym Lysak, and Peter Staar. Tableformer: Table structure understanding with transformers. In *CVPR*, 2022.

[4] Yongshuai Huang, Ning Lu, Dapeng Chen, Yibo Li, Zecheng Xie, Shenggao Zhu, Liangcai Gao, and Wei Peng. Improving table structure recognition with visual-alignment sequential coordinate modeling. In *CVPR*, 2023.

[5] Huawen Shen, Xiang Gao, Jin Wei, Liang Qiao, Yu Zhou, Qiang Li, and Zhanzhan Cheng. Divide rows and conquer cells: Towards structure recognition for large tables. In *IJCAI*, 2023.

[6] Nan Jiang, Shanchao Liang, Chengxiao Wang, Jiannan Wang, and Lin Tan. Latte: Improving latex recognition for tables and formulae with iterative refinement. In *AAAI*, 2024.

[7] Renqiu Xia, Hongbin Zhou, Ziming Feng, Huanxi Liu, Boan Chen, Bo Zhang, and Junchi Yan. Latexnet: A specialized model for converting visual tables and equations to latex code. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025.

[8] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Jun-Mei Song, Mingchuan Zhang, Y. K. Li, Yu Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *ArXiv*, 2024.

[9] Xu Zhong, Elaheh Shafieibavani, and Antonio Jimeno-Yepes. Image-based table recognition: data, model, and evaluation. In *ECCV*, 2019.

[10] Yongshuai Huang, Ning Lu, Dapeng Chen, Yibo Li, Zecheng Xie, Shenggao Zhu, Liangcai Gao, and Wei Peng. Improving table structure recognition with visual-alignment sequential coordinate modeling. *CVPR*, 2023.

[11] Sebastian Schreiber, Stefan Agne, Ivo Wolf, Andreas Dengel, and Sheraz Ahmed. Deepdesrt: Deep learning for detection and structure recognition of tables in document images. In *ICDAR*, 2017.

[12] Shubham Singh Paliwal, D Vishwanath, Rohit Rahul, Monika Sharma, and Lovekesh Vig. Tablenet: Deep learning model for end-to-end table detection and tabular data extraction from scanned document images. In *ICDAR*, 2019.

[13] Chris Tensmeyer, Vlad I Morariu, Brian Price, Scott Cohen, and Tony Martinez. Deep splitting and merging for table structure decomposition. In *ICDAR*, 2019.

[14] Zengyuan Guo, Yuechen Yu, Pengyuan Lv, Chengquan Zhang, Haojie Li, Zhihui Wang, Kun Yao, Jingtuo Liu, and Jingdong Wang. Trust: An accurate and end-to-end table structure recognizer using splitting-based transformers. *arXiv*, 2022.

[15] Zhenrong Zhang, Jianshu Zhang, Jun Du, and Fengren Wang. Split, embed and merge: An accurate table structure recognizer. *PR*, 2022.

[16] Chixiang Ma, Weihong Lin, Lei Sun, and Qiang Huo. Robust table detection and structure recognition from heterogeneous document images. *PR*, 2023.

[17] Pengyuan Lyu, Weihong Ma, Hongyi Wang, Yuechen Yu, Chengquan Zhang, Kun Yao, Yang Xue, and Jingdong Wang. Gridformer: Towards accurate table structure recognition via grid prediction. In *ACM MM*, 2023.

[18] Sachin Raja, Ajoy Mondal, and CV Jawahar. Table structure recognition using top-down and bottom-up cues. In *ECCV*, 2020.

[19] Hao Liu, Xin Li, Bing Liu, Deqiang Jiang, Yinsong Liu, Bo Ren, and Rongrong Ji. Show, read and reason: Table structure recognition with flexible context aggregator. In *ACM MM*, 2021.

[20] Jianqiang Wan, Sibo Song, Wenwen Yu, Yuliang Liu, Wenqing Cheng, Fei Huang, Xiang Bai, Cong Yao, and Zhibo Yang. Omniparser: A unified framework for text spotting key information extraction and table recognition. In *CVPR*, 2024.

[21] Zewen Chi, Heyan Huang, Heng-Da Xu, Houjin Yu, Wanxuan Yin, and Xian-Ling Mao. Complicated table structure recognition. *arXiv*, 2019.

[22] Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. Nougat: Neural optical understanding for academic documents. *arXiv*, 2023.

[23] Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liangyan Gui, Yu-Xiong Wang, Yiming Yang, Kurt Keutzer, and Trevor Darrell. Aligning large multimodal models with factually augmented rlhf. *ArXiv*, 2023.

[24] Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, and Tat-Seng Chua. Rlhf-v: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. *CVPR*, 2023.

[25] Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaoshen Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *ArXiv*, 2025.

[26] Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, Ruochen Xu, and Tiancheng Zhao. Vlm-r1: A stable and generalizable r1-style large vision-language model. *ArXiv*, 2025.

[27] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Annual Meeting of the Association for Computational Linguistics*, 2002.

[28] Yang Gao, Abdul Rehman, and Zhou Wang. Cw-ssim based image classification. In *2011 18th IEEE International Conference on Image Processing*, 2011.

[29] Erhard Schmidt. Zur theorie der linearen und nichtlinearen integralgleichungen. *Mathematische Annalen*, 1907.

[30] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv*, 2024.

[31] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[32] Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, 2024.

[33] Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, Ruochen Xu, and Tiancheng Zhao. Vlm-r1: A stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*, 2025.

[34] Yuze Zhao, Jintao Huang, Jinghan Hu, Xingjun Wang, Yunlin Mao, Daoze Zhang, Zeyinzi Jiang, Zhikai Wu, Baole Ai, Ang Wang, Wenmeng Zhou, and Yingda Chen. Swift:a scalable lightweight infrastructure for fine-tuning, 2024.

[35] Mathpix, Inc. Mathpix — ai-powered text recognition. `https://mathpix.com/`, 2025.

[36] OpenAI. Gpt-4o: Gpt-4 with vision capabilities. `https://openai.com/research/gpt-4o`, 2025.

[37] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[38] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please refer to the Introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to the Limitations.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: This work does not include a theoretical result.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have demonstrated all technical details to reproduce the results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Justification: The data and the code would be released after acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Please refer to the experimental setup.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Due to resource limitations, we do not repeat one of the experiments. Running a complete experiment takes one week.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We include the device information in the experimental setup.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeuroIPS Code of Ethics and checked the paper in every respect.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: This work is not related to any private or personal data, and there's no explicit negative social impacts.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: No such models or datasets are involved.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have cited the original papers that offer the original idea or technical details.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: We invite 5 graduate students majoring in computer science to participate in the volunteer-based evaluation. The instructions below are provided for each evaluator: at the top of the page, a reference table image is presented, followed by four model-generated table images. You are asked to anonymously select the image that best matches the reference. Preferably, choose only one. If multiple candidates appear equally similar, multiple selections are allowed. In cases where no image is clearly similar, prioritize those with the most similar structural layout.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: We use the pretrained MLLM in the task of table image to LaTeX code generation.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Case Study

In this section, we demonstrate the limitations of relevant metrics through result visualizations, and the capability of our method VSGRPO to generate such complex table and analyze the effectiveness of our approach.

The Limitation of Metric. We illustrate LaTeX-level ambiguity with two visually identical table renderings whose TEDS scores nonetheless differ. In Figure 2 (TEDS 0.8047), the ground-truth code wraps every cell's contents in an empty group { }, whereas the model output omits these no-op braces. Although neither variation alters the final rendering, they change the underlying token sequence and thus reduce the TEDS score. In both figures, the relevant LaTeX code differences are highlighted in yellow. In Figure 3 (TEDS 0.8983), the sole divergence lies in the use of different bold commands.

	Train	Dev	Test
¿=2	2,745	213	248
¿=5	1,477	138	144
¿=10	793	64	66
¿=20	459	0	0
¿=50	163	0	0

LaTeX

```
\\begin{tabular}{lccc} \hline
 & {Train} & {Dev} & {Test} \\ \hline
 {>=2} & {2,745} & {213} & {248}\\
 {>=5} & {1,477} & {138} & {144}\\
 {>=10} & {793} & {64} & {66}\\
 {>=20} & {459} & {0} & {0}\\
 {>=50} & {163} & {0} & {0}\\ \hline
\end{tabular}
```

(a) Ground Truth

	Train	Dev	Test
¿=2	2,745	213	248
¿=5	1,477	138	144
¿=10	793	64	66
¿=20	459	0	0
¿=50	163	0	0

LaTeX

```
\begin{tabular}{cccc} \hline
 & Train & Dev & Test \\ \hline
 >=2 & 2,745 & 213 & 248 \\
 >=5 & 1,477 & 138 & 144 \\
 >=10 & 793 & 64 & 66 \\
 >=20 & 459 & 0 & 0 \\
 >=50 & 163 & 0 & 0 \\ \hline
\end{tabular}
```

(b) Qwen2.5-VL-3B-VSGRPO **Metric:** CW-SSIM=0.9816, TEDS-Structure=1, TEDS=0.8047

Figure 2: Example 1: LaTeX code ambiguity.

Visualisation of Complex Tables. As shown in Figure 4 and Figure 5, they are the complex table image from the ground truth and the table image rendered from the LaTeX generated by our method VSGRPO, respectively.

Visualisation of effectiveness. As shown in Figure 6, our method VSGRPO improves the quality of the LaTeX generated by SFT, and the CW-SSIM score also reflects the visual similarity between the images.

B Metric

CW-SSIM. The specific formula is as follows:

$$\text{CW-SSIM}(X, Y) = \frac{1}{4} \sum_{i \in \{A,H,V,D\}} \text{SSIM}\big(c_X^i,\, c_Y^i\big), \tag{5}$$

where for each sub-band i:

$$\text{SSIM}(c_X^i, c_Y^i) = \frac{\big(2\mu_{X_i}\mu_{Y_i} + C_1\big)\,\big(2\sigma_{X_iY_i} + C_2\big)}{\big(\mu_{X_i}^2 + \mu_{Y_i}^2 + C_1\big)\,\big(\sigma_{X_i}^2 + \sigma_{Y_i}^2 + C_2\big)}. \tag{6}$$

Problem	UMAD 0.1	Bandit	GESMR	SAMR
Nguyen1	**50**	**50**	47	47
Nguyen2	**50**	**50**	48	34
Nguyen3	**48**	**43**	12	9
Nguyen4	34	**43**	14	2
Nguyen5	**50**	**42**	24	23
Nguyen6	**50**	**30**	6	23
Nguyen7	**8**	2	0	**3**
Nguyen8	0	0	0	0

LaTeX

```
\begin{tabular}{c|c|c|c|c}\toprule
Problem & UMAD 0.1 & Bandit & GESMR & SAMR
\\\midrule
Nguyen1&{\bf 50}&{\bf 50}&47&47\\
Nguyen2&{\bf 50}&{\bf 50}&48&\underline{34}\\
…
Nguyen7 & {\bf 8} &  2 & 0& {\bf 3}\\
Nguyen8 & 0 &  0 & 0 & 0\\\bottomrule
\end{tabular}
```

(a) Ground Truth

Problem	UMAD 0.1	Bandit	GESMR	SAMR
Nguyen1	**50**	**50**	47	47
Nguyen2	**50**	**50**	48	34
Nguyen3	**48**	**43**	12	9
Nguyen4	34	**43**	14	2
Nguyen5	**50**	**42**	24	23
Nguyen6	**50**	**30**	6	23
Nguyen7	**8**	2	0	**3**
Nguyen8	0	0	0	0

LaTeX

```
\begin{tabular}{c|c|c|c|c}\toprule
Problem & UMAD 0.1 & Bandit & GESMR & SAMR\\
\midrule
Nguyen1 & \textbf{50} & \textbf{50} & 47 & 47 \\
Nguyen2 & \textbf{50} & \textbf{50} & 48 &
\underline{34} \\
…
Nguyen7 & \textbf{8} & 2 & 0 & \textbf{3} \\
Nguyen8 & 0 & 0 & 0 & 0 \\
\bottomrule
\end{tabular}
```

(b) Qwen2.5-VL-3B-VSGRPO **Metric**: CW-SSIM=1, TEDS-Structure=1, TEDS=0.8983

Figure 3: Example 2: LaTeX code ambiguity.

Let X and Y be two aligned grayscale table images trimmed to even dimensions. Apply a one-level Haar wavelet to each, yielding four sub-bands c_X^i and c_Y^i for $i = A$ (approximation), H (horizontal detail), V (vertical detail), and D (diagonal detail). For each sub-band i, let μ_{X_i} and μ_{Y_i} be the pixel-wise means, $\sigma_{X_i}^2$ and $\sigma_{Y_i}^2$ the variances, and $\sigma_{X_iY_i}$ the covariance. Constants $C_1 = (K_1 L)^2$ and $C_2 = (K_2 L)^2$ stabilize the denominator (L is 255.0, K_1 is 0.01 and K_2 is 0.03).

TEDS-Structure. The specific formula is as follows:

$$\text{TEDS-Structure} = 1 - \frac{\text{TED}_{\text{structure}}}{\max\big(|T_{\text{pred}}|,\ |T_{\text{gt}}|\big)}, \tag{7}$$

$|T_{\text{pred}}|$ denotes the total number of nodes in the structural tree parsed from the predicted table, and $|T_{\text{gt}}|$ denotes the total number of nodes in the structural tree parsed from the ground-truth table.

TEDS. The Tree Edit Distance–based Similarity (TEDS) computation extends TEDS-Structure by first calculating the total edit distance: $\text{TED} = \text{TED}_{\text{structure}} + \text{TED}_{\text{content}}$, and finally normalizing by the larger of the two tree sizes to yield the similarity score. The specific formula is as follows:

$$\text{TEDS} = 1 - \frac{\text{TED}}{\max\big(|T_{\text{pred}}|,\ |T_{\text{gt}}|\big)}. \tag{8}$$

C Human Evaluation

To evaluate whether the table images rendered from the model-generated LaTeX better align with human preferences, we place the ground truth table image at the top and display the model's predicted table image below it, side by side. The order is randomly shuffled, and the names are hidden. We place the ground truth table image at the top, allowing humans to select one or more images based on subjective similarity. As shown in Figure 7.

D_I	Model	Score	w/o SACP \ w/ SACP					
			$\alpha = 0.05$			$\alpha = 0.1$		
			Coverage	Size (↓)	SSCV (↓)	Coverage	Size (↓)	SSCV (↓)
Indian Pines	1D-CNN	APS	0.95 \ 0.95	3.68 \ 2.28	0.41 \ 0.28	0.90 \ 0.90	2.52 \ 1.75	0.51 \ 0.45
		RAPS	0.95 \ 0.94	4.09 \ 2.29	0.54 \ 0.57	0.90 \ 0.90	2.54 \ 1.74	0.88 \ 0.30
		SAPS	0.95 \ 0.95	6.68 \ 4.31	2.48 \ 1.83	0.90 \ 0.90	5.56 \ 3.02	4.84 \ 1.95
	3D-CNN	APS	0.94 \ 0.95	5.73 \ 3.27	0.47 \ 0.38	0.90 \ 0.90	2.85 \ 2.06	0.35 \ 0.34
		RAPS	0.95 \ 0.95	4.12 \ 3.92	0.10 \ 0.21	0.90 \ 0.90	3.21 \ 2.38	0.11 \ 0.25
		SAPS	0.95 \ 0.95	6.40 \ 5.44	0.98 \ 0.95	0.91 \ 0.90	4.97 \ 3.72	1.61 \ 1.51
	HybridSN	APS	0.95 \ 0.95	5.56 \ 4.83	0.20 \ 0.16	0.90 \ 0.90	3.28 \ 2.74	0.92 \ 0.15
		RAPS	0.95 \ 0.95	7.34 \ 7.12	0.40 \ 0.96	0.90 \ 0.90	4.29 \ 3.95	0.98 \ 0.75
		SAPS	0.95 \ 0.95	6.79 \ 6.07	0.19 \ 0.41	0.90 \ 0.90	4.03 \ 3.40	0.64 \ 0.61
	SSTN	APS	0.95 \ 0.95	2.81 \ 1.73	0.42 \ 0.18	0.90 \ 0.90	2.00 \ 1.38	0.41 \ 0.47
		RAPS	0.95 \ 0.95	2.52 \ 1.62	0.29 \ 0.30	0.90 \ 0.90	1.87 \ 1.36	0.29 \ 0.34
		SAPS	0.95 \ 0.95	6.98 \ 4.16	4.38 \ 1.74	0.90 \ 0.90	5.33 \ 3.25	6.35 \ 3.09
Pavia University	1D-CNN	APS	0.95 \ 0.95	2.26 \ 1.92	0.39 \ 0.37	0.90 \ 0.90	1.65 \ 1.57	0.40 \ 0.27
		RAPS	0.95 \ 0.95	2.00 \ 1.83	0.23 \ 0.31	0.90 \ 0.90	1.59 \ 1.54	0.40 \ 0.29
		SAPS	0.95 \ 0.95	3.92 \ 3.99	1.77 \ 2.21	0.90 \ 0.90	3.44 \ 3.04	3.62 \ 2.68
	3D-CNN	APS	0.94 \ 0.95	2.77 \ 2.34	1.04 \ 0.69	0.89 \ 0.89	2.14 \ 1.79	0.94 \ 0.66
		RAPS	0.94 \ 0.94	2.57 \ 2.28	0.41 \ 0.46	0.89 \ 0.89	2.04 \ 1.76	0.64 \ 0.56
		SAPS	0.94 \ 0.94	4.80 \ 4.31	4.56 \ 3.85	0.89 \ 0.89	4.04 \ 3.32	6.56 \ 4.23
	HybridSN	APS	0.95 \ 0.95	4.79 \ 4.59	4.59 \ 3.58	0.90 \ 0.90	3.38 \ 3.01	2.39 \ 1.33
		RAPS	0.94 \ 0.95	5.50 \ 5.36	0.47 \ 0.36	0.89 \ 0.90	3.70 \ 3.74	0.81 \ 0.07
		SAPS	0.95 \ 0.95	5.57 \ 5.44	1.98 \ 2.35	0.90 \ 0.90	3.99 \ 3.71	3.33 \ 3.47
	SSTN	APS	0.95 \ 0.95	1.75 \ 1.24	0.22 \ 0.26	0.90 \ 0.90	1.39 \ 1.11	0.23 \ 0.29
		RAPS	0.95 \ 0.95	1.60 \ 1.22	0.20 \ 0.20	0.90 \ 0.90	1.33 \ 1.10	0.13 \ 0.23
		SAPS	0.95 \ 0.95	3.26 \ 2.24	2.07 \ 0.92	0.90 \ 0.90	2.75 \ 1.91	2.96 \ 1.64
Salinas	1D-CNN	APS	0.95 \ 0.95	1.40 \ 1.20	0.15 \ 0.15	0.90 \ 0.90	1.20 \ 1.09	0.18 \ 0.25
		RAPS	0.95 \ 0.95	1.37 \ 1.20	0.06 \ 0.20	0.90 \ 0.90	1.19 \ 1.07	0.23 \ 0.25
		SAPS	0.95 \ 0.95	3.63 \ 1.71	1.20 \ 0.16	0.90 \ 0.90	2.97 \ 1.28	2.04 \ 0.16
	3D-CNN	APS	0.95 \ 0.95	1.48 \ 1.25	0.20 \ 0.19	0.90 \ 0.90	1.17 \ 1.08	0.12 \ 0.16
		RAPS	0.95 \ 0.95	1.47 \ 1.24	0.13 \ 0.18	0.90 \ 0.90	1.15 \ 1.07	0.15 \ 0.17
		SAPS	0.95 \ 0.95	2.94 \ 2.02	0.58 \ 0.28	0.90 \ 0.90	2.39 \ 1.29	1.07 \ 0.13
	HybridSN	APS	0.95 \ 0.95	1.37 \ 1.09	0.18 \ 0.07	0.90 \ 0.90	1.10 \ 1.03	0.18 \ 0.23
		RAPS	0.95 \ 0.95	1.20 \ 1.07	0.12 \ 0.10	0.90 \ 0.90	1.06 \ 1.00	0.12 \ 0.31
		SAPS	0.95 \ 0.95	1.90 \ 1.37	0.42 \ 0.33	0.90 \ 0.90	1.66 \ 1.18	0.69 \ 0.31
	SSTN	APS	0.95 \ 0.95	1.70 \ 1.19	0.21 \ 0.16	0.90 \ 0.90	1.37 \ 1.08	0.23 \ 0.10
		RAPS	0.95 \ 0.95	1.60 \ 1.18	0.12 \ 0.14	0.90 \ 0.90	1.29 \ 1.06	0.10 \ 0.11
		SAPS	0.95 \ 0.95	5.42 \ 2.65	3.58 \ 0.86	0.90 \ 0.90	3.91 \ 2.07	3.64 \ 1.26

Ground Truth Table Image

Figure 4: Ground truth example of complex table.

D_I	Model	Score	w/o SACP \ w/ SACP					
			$\alpha = 0.05$			$\alpha = 0.1$		
			Coverage	Size (↓)	SSCV (↓)	Coverage	Size (↓)	SSCV (↓)
Indian Pines	1D-CNN	APS	0.95 \ 0.95	3.68 \ 2.28	0.41 \ 0.28	0.90 \ 0.90	2.52 \ 1.75	0.51 \ 0.45
		RAPS	0.95 \ 0.94	4.09 \ 2.29	0.54 \ 0.57	0.90 \ 0.90	2.54 \ 1.74	0.88 \ 0.30
		SAPS	0.95 \ 0.95	6.68 \ 4.31	2.48 \ 1.83	0.90 \ 0.90	5.56 \ 3.02	4.84 \ 1.95
	3D-CNN	APS	0.94 \ 0.95	5.73 \ 3.27	0.47 \ 0.38	0.90 \ 0.90	2.85 \ 2.06	0.35 \ 0.34
		RAPS	0.95 \ 0.95	4.12 \ 3.92	0.10 \ 0.21	0.90 \ 0.90	3.21 \ 2.38	0.11 \ 0.25
		SAPS	0.95 \ 0.95	6.40 \ 5.44	0.98 \ 0.95	0.91 \ 0.90	4.97 \ 3.72	1.61 \ 1.51
	HybridSN	APS	0.95 \ 0.95	5.56 \ 4.83	0.20 \ 0.16	0.90 \ 0.90	3.28 \ 2.74	0.92 \ 0.15
		RAPS	0.95 \ 0.95	7.34 \ 7.12	0.40 \ 0.96	0.90 \ 0.90	4.29 \ 3.95	0.98 \ 0.75
		SAPS	0.95 \ 0.95	6.79 \ 6.07	0.19 \ 0.41	0.90 \ 0.90	4.03 \ 3.40	0.64 \ 0.61
	SSTN	APS	0.95 \ 0.95	2.81 \ 1.73	0.42 \ 0.18	0.90 \ 0.90	2.00 \ 1.38	0.41 \ 0.47
		RAPS	0.95 \ 0.95	2.52 \ 1.62	0.29 \ 0.30	0.90 \ 0.90	1.87 \ 1.36	0.29 \ 0.34
		SAPS	0.95 \ 0.95	6.98 \ 4.16	4.38 \ 1.74	0.90 \ 0.90	5.33 \ 3.25	6.35 \ 3.09
Pavia University	1D-CNN	APS	0.95 \ 0.95	2.26 \ 1.92	0.39 \ 0.37	0.90 \ 0.90	1.65 \ 1.57	0.40 \ 0.27
		RAPS	0.95 \ 0.95	2.00 \ 1.83	0.23 \ 0.31	0.90 \ 0.90	1.59 \ 1.54	0.40 \ 0.29
		SAPS	0.95 \ 0.95	3.92 \ 3.99	1.77 \ 2.21	0.90 \ 0.90	3.44 \ 3.04	3.62 \ 2.68
	3D-CNN	APS	0.94 \ 0.95	2.77 \ 2.34	1.04 \ 0.69	0.89 \ 0.89	2.14 \ 1.79	0.94 \ 0.66
		RAPS	0.94 \ 0.94	2.57 \ 2.28	0.41 \ 0.46	0.89 \ 0.89	2.04 \ 1.76	0.64 \ 0.56
		SAPS	0.94 \ 0.94	4.80 \ 4.31	4.56 \ 3.85	0.89 \ 0.89	4.04 \ 3.32	6.56 \ 4.23
	HybridSN	APS	0.95 \ 0.95	4.79 \ 4.59	4.59 \ 3.58	0.90 \ 0.90	3.38 \ 3.01	2.39 \ 1.33
		RAPS	0.94 \ 0.95	5.50 \ 5.36	0.47 \ 0.36	0.89 \ 0.90	3.70 \ 0.74	0.81 \ 0.07
		SAPS	0.95 \ 0.95	5.57 \ 5.44	1.98 \ 2.35	0.90 \ 0.90	3.99 \ 3.71	3.33 \ 3.47
	SSTN	APS	0.95 \ 0.95	1.75 \ 1.24	0.22 \ 0.26	0.90 \ 0.90	1.39 \ 1.11	0.23 \ 0.29
		RAPS	0.95 \ 0.95	1.60 \ 1.22	0.20 \ 0.20	0.90 \ 0.90	1.33 \ 1.10	0.13 \ 0.23
		SAPS	0.95 \ 0.95	3.26 \ 2.24	2.07 \ 0.92	0.90 \ 0.90	2.75 \ 1.91	2.96 \ 1.64
Salinas	1D-CNN	APS	0.95 \ 0.95	1.40 \ 1.20	0.15 \ 0.15	0.90 \ 0.90	1.20 \ 1.09	0.18 \ 0.25
		RAPS	0.95 \ 0.95	1.37 \ 1.20	0.06 \ 0.20	0.90 \ 0.90	1.19 \ 1.07	0.23 \ 0.25
		SAPS	0.95 \ 0.95	3.63 \ 1.71	1.20 \ 0.16	0.90 \ 0.90	2.97 \ 1.28	2.04 \ 0.16
	3D-CNN	APS	0.95 \ 0.95	1.48 \ 1.25	0.20 \ 0.19	0.90 \ 0.90	1.17 \ 1.08	0.12 \ 0.16
		RAPS	0.95 \ 0.95	1.47 \ 1.24	0.13 \ 0.18	0.90 \ 0.90	1.15 \ 1.07	0.15 \ 0.17
		SAPS	0.95 \ 0.95	2.94 \ 2.02	0.58 \ 0.28	0.90 \ 0.90	2.39 \ 1.29	1.07 \ 0.13
	HybridSN	APS	0.95 \ 0.95	1.37 \ 1.09	0.18 \ 0.07	0.90 \ 0.90	1.10 \ 1.03	0.18 \ 0.23
		RAPS	0.95 \ 0.95	1.20 \ 1.07	0.12 \ 0.10	0.90 \ 0.90	1.06 \ 1.00	0.12 \ 0.31
		SAPS	0.95 \ 0.95	1.90 \ 1.37	0.42 \ 0.33	0.90 \ 0.90	1.66 \ 1.18	0.69 \ 0.31
	SSTN	APS	0.95 \ 0.95	1.70 \ 1.19	0.21 \ 0.16	0.90 \ 0.90	1.37 \ 1.08	0.23 \ 0.10
		RAPS	0.95 \ 0.95	1.60 \ 1.18	0.12 \ 0.14	0.90 \ 0.90	1.29 \ 1.06	0.10 \ 0.11
		SAPS	0.95 \ 0.95	5.42 \ 2.65	3.58 \ 0.86	0.90 \ 0.90	3.91 \ 2.07	3.64 \ 1.26

Qwen2.5-VL-3B-VSGRPO

Figure 5: Prediction example of complex table.

Data	Model	χ^2_{min}	H_0	Ω^0_m	M	ΔBIC
OHD	LDEM	30.19	$65.49^{+0.87}_{-0.89}$	$0.2902^{+0.0185}_{-0.0175}$	—	−2.23
	ΛCDM	32.42	$69.93^{+1.04}_{-1.06}$	$0.2658^{+0.0171}_{-0.0160}$	—	
OHD + SNe Ia	LDEM	1778.67	65.82 ± 0.72	$0.2830^{+0.0136}_{-0.0132}$	-19.46 ± 0.02	−20.42
	ΛCDM	1799.09	$67.16^{+0.82}_{-0.81}$	$0.3163^{+0.0137}_{-0.0133}$	-19.44 ± 0.02	
OHD + SNe Ia + BAO	LDEM	1780.53	66.04 ± 0.51	0.2800 ± 0.01	-19.46 ± 0.02	−20.50
	ΛCDM	1803.71	$68.08^{+0.64}_{-0.63}$	$0.2990^{+0.0090}_{-0.0088}$	-19.42 ± 0.02	
OHD + SNe Ia + CMB	LDEM	1778.67	65.83 ± 0.72	0.2800 ± 0.01	-19.46 ± 0.02	−4.81
	ΛCDM	1800.04	67.75 ± 0.56	$0.3052^{+0.0071}_{-0.0069}$	-19.42 ± 0.02	
OHD + SNe Ia + BAO + CMB	LDEM	1778.67	65.83 ± 0.72	0.2800 ± 0.01	-19.46 ± 0.02	−2.70
	ΛCDM	1803.74	$68.02^{+0.53}_{-0.52}$	$0.3000^{+0.0060}_{-0.0059}$	-19.42 ± 0.02	

(a) Ground Truth

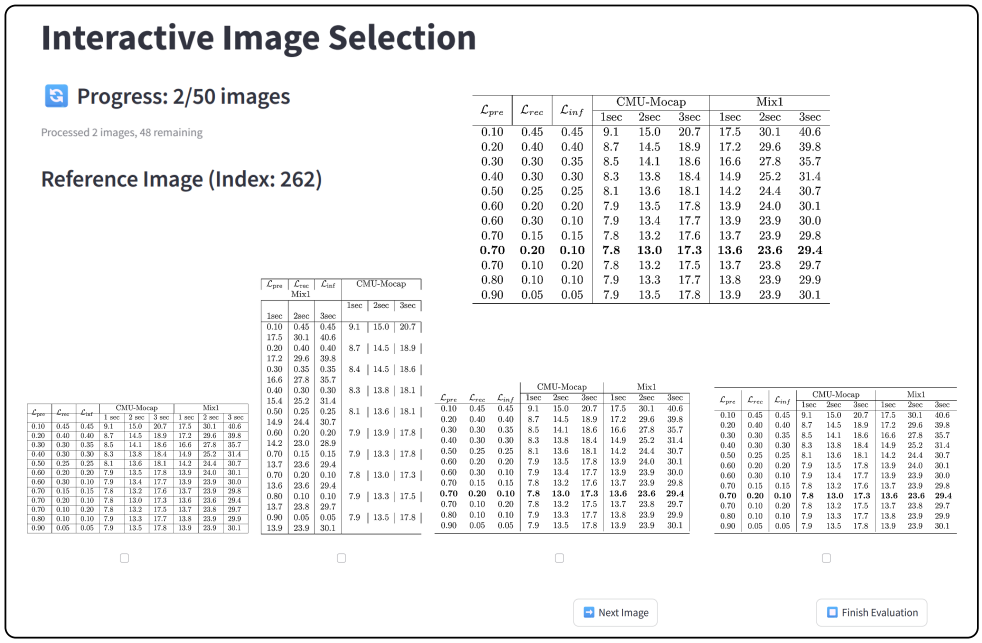
Data	Model	χ^2_{min}	H_0	Ω^0_m	M	ΔBIC
OHD	LDEM	30.19	$65.49^{+0.87}_{-0.89}$	$0.2902^{+0.0185}_{-0.0175}$	—	−2.23
	ΛCDM	32.42	$69.93^{+1.04}_{-1.06}$	$0.2658^{+0.171}_{-0.160}$	—	
OHD + SNe Ia	LDEM	1778.67	65.82 ± 0.72	$0.2830^{+0.136}_{-0.132}$	-19.46 ± 0.02	−20.42
	ΛCDM	1799.09	$67.16^{+0.82}_{-0.81}$	$0.3163^{+0.0127}_{-0.0133}$	-19.44 ± 0.02	
OHD + SNe Ia + BAO	LDEM	1780.53	66.04 ± 0.51	0.2800 ± 0.01	-19.46 ± 0.02	−20.50
	ΛCDM	1803.71	$68.08^{+0.64}_{-0.63}$	$0.2990^{+0.0090}_{-0.0088}$	-19.42 ± 0.02	
OHD + SNe Ia + CMB	LDEM	1778.67	65.83 ± 0.72	0.2800 ± 0.01	-19.46 ± 0.02	−4.81
	ΛCDM	1800.04	67.75 ± 0.56	$0.3052^{+0.0074}_{-0.0069}$	-19.42 ± 0.02	
OHD + SNe Ia + BAO + CMB	LDEM	1778.67	65.83 ± 0.72	0.2800 ± 0.01	-19.46 ± 0.02	−2.70
	ΛCDM	1803.74	$68.02^{+0.53}_{-0.52}$	$0.3000^{+0.0060}_{-0.0059}$	-19.42 ± 0.02	

(b) Qwen2.5-VL-3B-VSGRPO (CW-SSIM:0.9876)

Data	Model	χ^2_{min}	H_0	Ω^0_m	M	ΔBIC
OHD	LDEM	30.19	$65.49^{+0.87}_{-0.89}$	$0.2902^{+0.0185}_{-0.0175}$	—	−2.23
	ΛCDM	32.42	$69.93^{+1.04}_{-1.06}$	$0.2658^{+0.0171}_{-0.0160}$	—	
OHD + SNe Ia	LDEM	1778.67	65.82 ± 0.72	$0.2830^{+0.0136}_{-0.0132}$	-19.46 ± 0.02	−20.42
	ΛCDM	1799.09	$67.16^{+0.82}_{-0.81}$	$0.3163^{+0.0137}_{-0.0133}$	-19.44 ± 0.02	
OHD + SNe Ia + BAO	LDEM	1780.53	66.04 ± 0.51	0.2800 ± 0.01	-19.46 ± 0.02	−20.50
	ΛCDM	1803.71	$68.08^{+0.64}_{-0.63}$	$0.2990^{+0.0090}_{-0.0088}$	-19.42 ± 0.02	
OHD + SNe Ia + CMB	LDEM	1778.67	65.83 ± 0.72	0.2800 ± 0.01	-19.46 ± 0.02	−4.81
	ΛCDM	1800.04	67.75 ± 0.56	$0.3052^{+0.0074}_{-0.0069}$	-19.42 ± 0.02	
OHD + SNe Ia + BAO + CMB	LDEM	1778.67	65.83 ± 0.72	0.2800 ± 0.01	-19.46 ± 0.02	−2.70
	ΛCDM	1803.74	$68.02^{+0.53}_{-0.52}$	$0.3000^{+0.0060}_{-0.0059}$	-19.42 ± 0.02	

(c) Qwen2.5-VL-3B-SFT (CW-SSIM:0.6092)

Figure 6: Visualization of result comparisons. (a) Ground Truth refers to the ground truth table image from the simple testing dataset; (b) Qwen2.5-VL-3B-VSGRPO represents the table image rendered from LaTeX generated by the Qwen2.5-VL-3B model trained with our VSGRPO method; (c) Qwen2.5-VL-3B-SFT represents the table image rendered from LaTeX generated by the Qwen2.5-VL-3B model trained with SFT. The corresponding CW-SSIM scores are reported. Blue boxes highlight examples where Qwen2.5-VL-3B-VSGRPO differs from the ground truth, and red boxes highlight examples where Qwen2.5-VL-3B-SFT differs from the ground truth.



$\mathcal{L}_{pre}$	$\mathcal{L}_{rec}$	$\mathcal{L}_{inf}$	CMU-Mocap			Mix1		
			1sec	2sec	3sec	1sec	2sec	3sec
0.10	0.45	0.45	9.1	15.0	20.7	17.5	30.1	40.6
0.20	0.40	0.40	8.7	14.5	18.9	17.2	29.6	39.8
0.30	0.30	0.35	8.5	14.1	18.6	16.6	27.8	35.7
0.40	0.30	0.30	8.3	13.8	18.4	14.9	25.2	31.4
0.50	0.25	0.25	8.1	13.6	18.1	14.2	24.4	30.7
0.60	0.20	0.20	7.9	13.5	17.8	13.9	24.0	30.1
0.60	0.30	0.10	7.9	13.4	17.7	13.9	23.9	30.0
0.70	0.15	0.15	7.8	13.2	17.6	13.7	23.9	29.8
0.70	0.20	0.10	7.8	13.0	17.3	13.6	23.6	29.4
0.70	0.10	0.20	7.8	13.2	17.5	13.7	23.8	29.7
0.80	0.10	0.10	7.9	13.3	17.7	13.8	23.9	29.9
0.90	0.05	0.05	7.9	13.5	17.8	13.9	23.9	30.1

Figure 7: The table image selection page for human evaluation.