

Tabstructor: Reconstructing Latent Borders in Borderless and Partially Ruled Tables

Alvin Sarraga Alon^{1,*}, Daniele De Rosa²,
Johannes Friedl³, Riza Batista-Navarro¹

¹Department of Computer Science, University of Manchester, United Kingdom

²Department of Agriculture, Forestry, Food and Environmental Sciences,
University of Basilicata, Potenza, Italy

³University of Natural Resources and Life Sciences, Vienna, Austria

*Correspondence: alvin.alon@manchester.ac.uk

Abstract

Borderless and partially ruled scientific tables often omit visual boundaries that mark logical rows, columns, and spanning regions. We introduce Tabstructor, a vision–language framework for latent-border reconstruction: predicting a table bounding box and horizontal and vertical segments representing logical boundaries even when visually absent or incomplete. On a held-out external test set, we compare task-specific prompting of a frozen base model, supervised fine-tuning, structure-preserving augmentation, and optional OCR-assisted geometry refinement. We introduce the Latent Border Reconstruction Score (LBRS; 0–100, higher is better) for evaluating valid structural and geometric border recovery. Task-specific prompting improves the frozen model, while fine-tuning provides the dominant gain, increasing LBRS from 38.552 to 90.014 (paired gain 51.462; 95% paired-bootstrap CI: 45.583–57.259). The augmented predictor reaches 93.487, a further 3.473-point gain, although it receives additional cumulative training exposure. OCR-assisted refinement improves boundary localisation and extent but adds only 0.467 points on average (95% CI: –1.968–2.615) and can cause boundary-count regressions. Thus, fine-tuning performs the principal reconstruction, while refinement is an optional, non-uniform geometry adjustment.

1 Introduction

Many scientific tables express rows, columns, grouped headers, and spanning cells through whitespace, alignment, typography, and partial rules rather than a complete grid. In complex layouts with multi-level headers, merged cells, stub columns, partial separators, or local panels, some

logical boundaries are not visibly drawn, complicating recovery of row, column, and logical-cell membership from a table image.

Table structure recognition (TSR) already recovers cells, relations, separators, or markup from document images (Khang and Hong, 2025; Xue et al., 2021; Lin et al., 2022). Representative TSR systems instead predict markup or layout pointers, cell locations and relations, or row and column separation lines and masks (Khang and Hong, 2025; Xue et al., 2021; Lin et al., 2022; Zhang et al., 2024). We examine a narrower representational gap by making the count, position, and bounded extent of latent structural borders an explicit and editable prediction target.

We call this task **latent-border reconstruction**. Given a single-table image, the objective is to predict its boundary and the horizontal and vertical segments supported by the logical table organisation, including boundaries that are only partially drawn or absent from the visible ink. The target therefore requires inference from layout and context rather than only visible-line detection. It provides geometric evidence for subsequent row, column, and logical-cell recovery, but does not constitute complete semantic table understanding.

We introduce **Tabstructor**, a vision–language framework for predicting this latent-border geometry. Throughout this paper, *Tabstructor* denotes the overall task framework; S3–S4 are its learned predictors, S4 is the proposed primary predictor, and S5 is the separately reported refinement analysis. A vision–language model (VLM) is a plausible basis for the task because missing boundaries must be inferred jointly from visual layout and textual context (Zhang, 2024; Alonso et al., 2026; Chen et al., 2026); however, general multimodal

capability does not establish reliable fine-grained geometric prediction. This limitation motivates a controlled evaluation of task-specific model adaptation. We treat structure-preserving augmentation as a distinct training condition and OCR-assisted geometry refinement as an optional downstream analysis, rather than as part of the primary learned reconstruction.

The study addresses three research questions: **RQ1** asks how task-specific prompting and fine-tuning affect latent-border reconstruction performance; **RQ2** asks whether the evaluated structure-preserving augmented training condition improves reconstruction, while recognising its additional cumulative training exposure; and **RQ3** asks whether OCR-assisted geometry refinement adds meaningful benefit beyond the raw fine-tuned prediction.

The controlled comparisons distinguish gains from learned reconstruction from changes due to the optional downstream refinement stage.

The contributions of this work are threefold:

1. **Latent-border task and resources.** We formulate latent-border reconstruction as explicit prediction of normalised table bounds and logically supported border segments, and provide an annotation protocol and reviewed completed-border references constructed with TeybolStruct, our human-in-the-loop editor for creating and reviewing table bounding boxes and logical horizontal and vertical border segments.
2. **Controlled VLM evaluation.** We define five frozen conditions that support paired, held-out comparisons of model adaptation, training data, and downstream processing, and introduce the Latent Border Reconstruction Score (LBRS) as a bounded measure of valid structural and geometric border recovery.
3. **Reconstruction and refinement evidence.** We provide component-level and per-table analyses that separate primary learned predictions from optional geometry adjustments.

2 Related Work

Table analysis commonly separates table-region detection from table structure recognition (TSR), which recovers rows, columns, cells, spans, relations, or markup (Prajapati and Maurya, 2026). Direct formulations predict cell relationships or serialised layouts (Xue et al., 2021; Khang and Hong, 2025), whereas separator-based formulations infer

row and column divisions from image evidence (Lin et al., 2022; Zhang et al., 2024). Both families recover table structure; the distinction relevant here is representational. Latent-border reconstruction makes the count, position, and bounded extent of inferred structural boundaries an explicit prediction target, rather than claiming that prior TSR does not infer missing structure.

Borderless and partially ruled tables are difficult because whitespace, alignment, typography, and local rules must jointly indicate divisions that a complete rendered grid would otherwise provide. Prior work addresses complex layouts through feature aggregation, spanning-cell recovery, and learned structure prediction (Hong et al., 2024; Nguyen et al., 2023; Yu et al., 2025). These approaches establish that structure can be inferred without tracing only visible rules. A remaining representational issue arises when a logical boundary is absent or valid over only part of the table: a full-width separator can split a spanning cell, while a single boundary plane may require disjoint segments. The present task targets these latent logical segments explicitly, including their local extents, while remaining complementary to cell-, relation-, and markup-based TSR.

Multimodal models have also been developed for document understanding and reasoning (Zhang, 2024), visual table understanding (Alonso et al., 2026), and instruction-guided TSR (Chen et al., 2026). Generative interfaces support constrained outputs such as JSON or markup, but prior success at document or table understanding does not by itself establish reliable fine-grained border geometry. This study therefore examines whether task-specific prompting is sufficient for latent-border reconstruction and how much task-specific fine-tuning changes structured geometric prediction; the base VLM is an enabling component rather than the claimed novelty.

Geometric priors and layout-aware optimisation have been used to improve row and column localisation in table structure recognition (Hashmi et al., 2021). Motivated by the broader value of geometric constraints, we separately evaluate whether inference-time refinement can improve the raw learned prediction. Strong regularisation may nevertheless be mismatched to irregular or partial boundaries, so this secondary refinement analysis does not access reference borders. Overall, prior TSR recovers complex structure through several effective representations, while VLMs provide a

flexible structured-generation interface; the open space examined here is explicit reconstruction of bounded latent borders in borderless and partially ruled scientific tables, together with the relative roles of prompting, fine-tuning, and downstream refinement.

3 Task Definition and Data

3.1 Latent-Border Reconstruction

Let I denote a single-table image of a borderless or partially ruled table. Locating a table among multiple page regions is outside the task; the predicted table box localises the table content within the supplied image. The target is the structured geometric representation

$$Y = (B, H, V), \quad (1)$$

where B is the table bounding box and H and V are sets of horizontal and vertical border segments. Box coordinates are normalised to the image and segment coordinates to B . The model emits these components as structured JSON; Appendix C specifies the exact schema, coordinate ordering, and transforms.

We call a boundary *latent* when it is supported by the table’s adjudicated row, column, or spanning structure rather than merely traced from visible ink. The *completed-border target* contains all such supported boundaries: it retains visible structural rules and adds missing portions or wholly implicit separators, while excluding decorative rules. A boundary may comprise disjoint collinear segments when a full-span separator would cross a grouped or spanning region. Figure 1 illustrates the resulting image-to-geometry task. The direct prediction is only this geometry; OCR transcription, cell identities and relations, semantic labels, HTML generation, and subsequent geometry refinement are outside the target.

3.2 Target Annotation Policy

Every target contains a closed outer rectangle. Within it, annotators retain the minimum set of full-span or bounded separators needed to express the adjudicated logical partition. Separators are interrupted where they would divide grouped headers, spanning cells, or panel-local regions, and visible decorative rules are excluded. When multiple geometries are plausible, the simplest topology supported by row and column alignment, spans, and local structure is selected. Thus, each annotation

is one adjudicated geometric interpretation rather than a claim that the layout has a unique semantic analysis.

One primary annotator created the targets with **TeybolStruct**, whose automated initialisation remained subject to manual correction. Every serialised-target rendering was reviewed before acceptance; unresolved cases received collaborative review and, when necessary, team adjudication. Labels also passed schema and coordinate round-trip validation. Because labels were not independently duplicated, formal inter-annotator agreement was not measured. The resulting annotations serve as training targets and as validation and test references; Appendix C gives the full protocol.

3.3 Dataset and Splits

The dataset comprises single-table images derived from published scientific literature, including borderless and partially ruled layouts, paired with one completed-border annotation per image. The development corpus contains 757 original training samples and 84 unaugmented validation samples. The augmented training condition adds 754 accepted variants derived only from the original training images; each variant preserves its source image dimensions and structural target (Table 1).

All 84 validation samples are used for token-loss monitoring, while a fixed 32-sample subset of them—not a third data split—is used for structural checkpoint selection. Validation data do not update model parameters, and the remaining 52 samples contribute only to loss monitoring, not structural checkpoint ranking. For the labels used below, S1–S2 are frozen prompting baselines, S3–S4 are fine-tuned predictors, S4 is the primary augmented Tabstructor predictor, and S5 refines S4. The separately annotated 100-table test set is external to the development corpus, uses the same representation, annotation policy, and review procedure, and is reserved for final evaluation; it is not used for training, loss monitoring, or checkpoint selection.

Ancestry checks found no overlap between original or augmented training records and validation records. Augmented records exactly retained their source targets, and test references were finalised without access to system predictions or scores. Appendices D and G provide the augmentation-integrity and selection details.

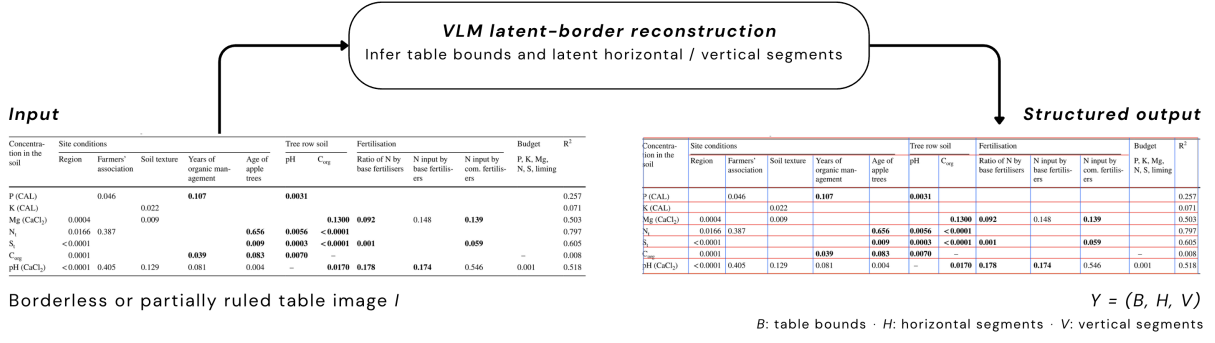


Figure 1: Latent-border reconstruction task. A vision–language model (VLM) takes a borderless or partially ruled table image *I* (left) and predicts the structured border geometry $Y = (B, H, V)$, visualised over the table on the right. *B* is the table bounding box, and *H* and *V* are the horizontal (red) and vertical (blue) border segments that recover the latent row and column separations; arrows show the input–model–output flow.

4 Tabstructor Framework

4.1 Model and Structured Generation

Tabstructor is the overall framework for mapping a single-table image to the latent-border representation defined in Section 3: a table bounding box and horizontal and vertical border segments serialised as JSON. It organises five controlled systems rather than treating every condition as a component of one deployed pipeline. Systems S1–S2 are frozen base-VLM prompting baselines, systems S3–S4 are learned Tabstructor predictors obtained by parameter-efficient fine-tuning of the same base VLM, and system S5 is an optional inference-time refinement analysis of S4. The augmented fine-tuned system S4 is the proposed primary predictor; its output is the raw learned prediction, before any refinement.

For S3–S4, Low-Rank Adaptation (LoRA) adapts the unsloth/Qwen3.5–9B base VLM to generate task-specific latent geometry from the table image and instruction, rather than merely tracing visible rules. The two fine-tuned systems share the base-model revision, adapter configuration, full prompt, and inference protocol, and differ principally in their training data. This separates learned reconstruction from the frozen prompting baselines and from downstream post-processing. Appendix D reports the exact model, LoRA, optimisation, length, compute, and run settings; Appendix F reports the shared inference and validation settings.

4.2 Minimal and Full Prompts

S1 and S2 keep the base VLM frozen and differ only in prompting: S1 uses a minimal schema-oriented instruction, whereas S2 uses the task-

specific full prompt. This contrast tests what prompting alone contributes under matched inputs, decoding, parsing, and evaluation. The fine-tuned systems S3–S4 use the same frozen full prompt, preventing prompt variation from confounding the fine-tuning and augmentation comparisons. S5 adds no generation prompt because it post-processes S4’s output. Appendix B preserves the exact prompt and output protocol.

4.3 Structure-Preserving, Typography-Matched Text Augmentation

Augmentation is used only during S4 training to vary textual and typographic appearance without changing the latent-border geometry. The off-line procedure replaces table text with typography-matched synthetic content while preserving image dimensions and the source target; reference geometry is used only to construct and verify training variants and is unavailable at inference. S3 trains on the original data, whereas S4 adds the accepted variants; validation and test data remain unaugmented. Thus, S4–S3 tests structure-preserving augmentation, although the contrast includes additional cumulative training exposure and does not isolate text diversity alone. Appendix D gives the exact data counts, exposure calculation, and augmentation procedure.

4.4 OCR-Assisted Geometry Refinement

S5 is a separate inference-time post-processing condition that takes the exact raw S4 prediction and corresponding source image. It uses zai-org/GLM-OCR (Duan et al., 2026) once as the OCR component to derive content-occupancy evidence, which guides frozen border alignment and conservative local repair; OCR transcription is not

Table 1: Dataset splits and experimental roles, grouped by training, validation, and evaluation data. Source tables are unique original table images, whereas records are dataset instances, including accepted variants in the augmented training set. The fixed 32-table subset of the 84-table validation set is used for checkpoint selection by Latent-Border Reconstruction Score (LBRS), and the separate 100-table external test set is held out for final evaluation.

Split / condition	Source tables	Records	Purpose
<i>Training data</i>			
Original training set	757	757	Non-augmented fine-tuning
Accepted augmented variants	—	754	Added augmented instances
Augmented training set	757	1,511	Tabstructor (S4) fine-tuning
<i>Validation data</i>			
Full validation set	84	84	Validation during fine-tuning
Fixed validation subset (of 84)	32	32	LBRS-based checkpoint selection
<i>Evaluation data</i>			
External test set	100	100	Held-out final evaluation (S1–S5)

added to the prediction. The OCR evidence comes from a learned model, but the subsequent geometry transformations are deterministic conditional on the accepted evidence. Refinement performs neither further VLM training nor a second border-generation pass, and it reads no reference borders, test annotations, or metric outputs. Consequently, S5–S4 isolates whether downstream geometry refinement is necessary beyond the primary raw S4 prediction, rather than evaluating a new end-to-end architecture. Appendix E gives the ordered algorithm and frozen parameters.

5 Experimental Design

5.1 Research Questions

The experiments operationalise the three RQs introduced in Section 1 through four paired comparisons defined before external-test evaluation:

- **RQ1: Prompting and fine-tuning.** Compare S2–S1 to measure the full-prompt effect and S3–S2 to measure the fine-tuning effect beyond prompting.
- **RQ2: Augmentation.** Compare S4–S3 to measure the evaluated augmented training condition under the same validation protocol; unequal cumulative exposure precludes attribution to text diversity alone.
- **RQ3: OCR-assisted geometry refinement.** Compare S5–S4 to isolate the frozen refinement-stage effect on the same underlying predictions.

5.2 Five Frozen Systems

Table 2 summarises five predefined conditions. S1 is the frozen base-VLM minimal-prompt baseline,

and S2 applies the full task prompt to the same frozen model. S3 fine-tunes that base model on the original training set, whereas the primary raw predictor, S4, uses the same full prompt and matched fine-tuning configuration with structure-preserving augmented training data. Augmentation is confined to training; validation and test inputs remain un-augmented. S5 is not separately trained: it applies frozen downstream refinement to the exact raw S4 predictions. All systems use the same output representation and external test set, and prompts, inference settings, and metric parameters are fixed before external-test evaluation.

5.3 Data Roles and Frozen Selection Protocol

S3 is trained on the 757 original training tables, while S4 adds the 754 accepted training-only variants; neither condition uses validation or test examples for parameter updates. Token-level loss is monitored on all 84 validation tables, and the unique maximum unrounded macro LBRS on a fixed 32-table subset of those tables selects one checkpoint per condition; the following subsection formally defines this metric. This subset is part of the validation split, not a third partition. After selection, checkpoints and all inference and evaluation settings are frozen, and only then are the five systems evaluated on the separately annotated 100-table external test set. Figure 4 in Appendix G visualises this validation-only selection and held-out evaluation protocol. The external test set is never used for training, loss monitoring, checkpoint selection, or configuration decisions. Appendix G also reports the selected checkpoint identifiers, complete validation rankings, and trajectories; Appendix D gives the training and checkpoint-

Table 2: Experimental configuration matrix for the five frozen systems evaluated on the same 100-table external test set. Rows group prompting-only base vision-language model (VLM) baselines (S1–S2), fine-tuned models (S3–S4), and refinement analysis (S5); columns indicate prompt type, use of fine-tuning, structure-preserving augmentation, and refinement, and each configuration’s comparison purpose. S4 is the proposed Tabstructor system, and S5 applies geometry refinement to S4’s raw predictions; boldface identifies the proposed system.

System	Prompt	Fine-tuned	Augmentation	Refinement	Purpose
<i>Base VLM baselines (prompting only)</i>					
S1 Base VLM – Minimal Prompt	Minimal	No	No	No	Minimal prompting baseline
S2 Base VLM – Full Prompt	Full	No	No	No	Effect of task-specific prompting
<i>Fine-tuned models</i>					
S3 Tabstructor predictor (no augmentation)	Full	Yes	No	No	Effect of fine-tuning beyond prompting
S4 Tabstructor primary predictor (Ours)	Full	Yes	Yes	No	Effect of structure-preserving augmentation
<i>Refinement analysis</i>					
S5 Refinement analysis (from S4)	Full	Yes	Yes	Yes	Effect of geometry refinement on S4 outputs

saving configurations.

5.4 Latent Border Reconstruction Score

Each system’s predicted table box and horizontal and vertical latent-border segments are compared with the adjudicated reference geometry. The primary metric, the *Latent Border Reconstruction Score* (LBRS), rewards valid structured outputs that recover the number, position, and bounded extent of the reference boundaries together with table localisation. Parse and schema validity form a hard gate; table-box intersection over union (IoU), boundary-count agreement, snap-position agreement, and interval Dice describe complementary structural and geometric quality. Snap is a tolerance-weighted measure of matched boundary-plane position accuracy, while interval Dice compares the unioned extents of matched collinear border segments. Line-position mean absolute error (MAE) is reported only as a secondary diagnostic. This combination is appropriate because a usable reconstruction must recover both the required boundary planes and their geometry. For table i ,

$$\begin{aligned} \text{LBRS}_i = 100G_i & [0.05B_i \\ & + 0.375R_{c,i} + 0.05R_{s,i} + 0.05R_{d,i} \\ & + 0.375C_{c,i} + 0.05C_{s,i} + 0.05C_{d,i}]. \end{aligned} \quad (2)$$

where G_i is the parse-and-schema validity gate, B_i is table-box IoU, and the R and C terms measure horizontal and vertical boundary-count recovery (c), position agreement (s), and interval coverage (d). Appendix F gives the complete matching rules, tolerances, clipping, and edge conventions.

The official evaluation-set score is the macro-

average of the per-table scores:

$$\text{LBRS}_{\text{macro}} = \frac{1}{N} \sum_{i=1}^N \text{LBRS}_i. \quad (3)$$

LBRS ranges from 0 to 100 and higher is better. The primary score is the macro-average of gated per-table scores; MAE is reported separately over matched planes and is not included in LBRS. Thus, LBRS measures border-structure and geometric reconstruction rather than complete cell topology or semantic table understanding. Appendix F specifies the aggregation procedure.

5.5 Statistical Reporting

Because all systems are evaluated on the same 100 tables, the principal analysis uses the four per-table paired contrasts defined above. We report macro means, medians, standard deviations, paired bootstrap 95% confidence intervals, and improved, unchanged, and degraded table counts. These intervals quantify variation in paired effects across test tables, not training-run variability; each fine-tuning condition is represented by one training run. Appendix F gives the resampling count, seed, and exact reporting conventions.

6 Results and Discussion

Across the paired external-test comparisons, task-specific prompting improves the frozen base VLM, fine-tuning produces the dominant gain, and the augmented training condition adds a smaller improvement. Refinement of the primary raw prediction usually improves geometry, but its mean benefit is small because a few large regressions offset the more frequent gains. Tables 3 and 5 report the supporting absolute and paired results.

Table 3: Macro-average latent-border reconstruction performance of five systems (S1–S5) on the fixed 100-table external test set, grouped as prompting-only vision-language model (VLM) baselines, fine-tuned Tabstructor variants, and refinement analysis. Columns report output validity and bounding-box (BBox) localisation, horizontal- and vertical-border recovery, mean absolute error (MAE), and the overall Latent-Border Reconstruction Score (LBRS). All values are percentages except MAE (pixels) and LBRS (0–100); $\uparrow$ indicates higher is better. S4 is **Tabstructor (Ours)**, and S5 refines S4 outputs. **Boldface** and underlining denote the best and second-best values, respectively; MAE † is unranked because it averages only finite per-table values and depends on parse/schema coverage.

System	Output & extent			Horizontal borders				Vertical borders				LBRS ↑
	Parse ↑	Schema ↑	BBox ↑	Count ↑	Snap ↑	Dice ↑	MAE [†]	Count ↑	Snap ↑	Dice ↑	MAE [†]	
<i>Base VLM baselines (prompting only)</i>												
S1 Base VLM – Minimal Prompt	55.00	34.00	33.16	7.09	9.99	6.50	4.54	25.31	12.08	8.09	7.60	15.64
S2 Base VLM – Full Prompt	<u>75.00</u>	<u>74.00</u>	72.50	32.25	30.02	26.31	6.50	48.58	25.43	10.55	8.22	38.55
<i>Fine-tuned models</i>												
S3 Tabstructor predictor (no augmentation)	100.00	100.00	98.48	90.00	89.28	91.23	4.44	<u>99.62</u>	43.59	55.52	11.51	90.01
S4 Tabstructor primary predictor (Ours)	100.00	100.00	<u>98.38</u>	<u>95.57</u>	<u>95.24</u>	<u>95.64</u>	3.38	99.92	<u>49.37</u>	<u>64.94</u>	13.89	<u>93.49</u>
<i>Refinement analysis</i>												
S5 Refinement analysis (from S4)	100.00	100.00	<u>98.38</u>	96.19	96.95	96.71	2.45	92.34	82.51	90.53	7.77	93.95

Metrics. H/V denote horizontal/vertical borders. MAE † averages finite per-table values and should be interpreted with parse/schema coverage.

Table 4: Post hoc sensitivity of macro-average Latent-Border Reconstruction Score (LBRS; 0–100, higher is better) for systems S1–S5 on the fixed 100-table external test set to four component-weighting schemes, including a geometry-balanced variant added for robustness. S1–S2 are prompting-only baselines, S3–S4 are fine-tuned systems, and S5 refines S4. Rows give the combined horizontal-and-vertical weights for boundary-count recovery (Count), snap-position accuracy (Snap), and interval-based segment coverage (Dice); the bounding-box (BBox) weight remains 5%. **Boldface** and underlining mark the best and second-best system scores within each row, respectively. $\Delta(S5-S4)$ is the LBRS-point difference between refined and unrefined Tabstructor, and * identifies the official weighting used in the primary evaluation. All system outputs are frozen; this analysis changes only the component weights.

Weighting	Count (%)	Snap (%)	Dice (%)	LBRS $\uparrow$					$\Delta(S5-S4)$
				S1	S2	S3	S4	S5	
Count-heavy	80.0	7.5	7.5	15.991	39.419	91.260	94.559	<u>94.084</u>	−0.475
Official*	75.0	10.0	10.0	15.639	38.552	90.014	<u>93.487</u>	93.954	+0.467
Higher localization weighting	70.0	12.5	12.5	15.287	37.686	88.769	<u>92.415</u>	93.825	+1.410
Geometry-balanced	47.5	23.75	23.75	13.703	33.787	83.164	<u>87.591</u>	93.242	+5.651

6.1 RQ1: Prompting and Fine-tuning

The full prompt raises schema validity from 34% to 74% and LBRS from 15.639 to 38.552, a paired gain of 22.913 points (95% CI: 16.879–28.928). For the evaluated base VLM, task-specific instructions therefore improve output reliability and latent-border recovery, but 26 invalid S2 generations and weak vertical interval recovery show that prompting alone remains insufficient.

Fine-tuning produces the dominant improvement: S3 attains complete parse/schema validity and 90.014 LBRS, gaining 51.462 points over S2 (95% CI: 45.583–57.259). Vertical counts are nearly perfect, although placement and bounded extent remain weaker. Thus, within this model and test setting, prompting makes valid reconstruction more attainable, while the fine-tuned condition

accounts for most of the observed performance increase.

6.2 RQ2: Structure-Preserving Typography-Matched Augmentation

The primary raw predictor S4 reaches 93.487 LBRS and gains 3.473 points over S3 (95% CI: 2.061–4.986), a modest effect relative to fine-tuning. The augmented condition improves 69 tables and degrades 31, with its clearest component gain in vertical interval coverage; vertical snap improves whereas vertical MAE worsens, so the localisation evidence is metric-dependent. RQ2 therefore favours the evaluated augmented training condition on average, not augmentation universally. Because S4 receives approximately 15.6% more cumulative exposure, the comparison cannot attrib-

Table 5: Paired per-table changes in Latent Border Reconstruction Score (LBRS; 0–100, higher is better) between consecutive systems S1–S5 on the fixed 100-table external test set. Rows identify each system contrast and added intervention; columns report the mean later-minus-earlier change (Δ , in LBRS points), its 95% paired-bootstrap confidence interval (CI), and the numbers of tables that improved, remained unchanged, or degraded. Positive Δ values favour the later system; $\uparrow$ indicates higher is better.

Contrast	Intervention	Mean Δ LBRS $\uparrow$	95% CI	Improved	Same	Degraded
S1 $\rightarrow$ S2	Task-specific prompting	+22.913	[16.879, 28.928]	66	19	15
S2 $\rightarrow$ S3	Fine-tuning	+51.462	[45.583, 57.259]	98	0	2
S3 $\rightarrow$ S4	Structure-preserving augmentation	+3.473	[2.061, 4.986]	69	0	31
S4 $\rightarrow$ S5	Geometry refinement	+0.467	[−1.968, 2.615]	90	0	10

ute the gain specifically to text diversity rather than additional exposure.

6.3 RQ3: Geometry Refinement

Refinement improves 90 tables and degrades ten, with a median paired gain of 3.191 LBRS points, but its mean gain is only 0.467 points (95% CI: −1.968–2.615). Thus, refinement typically helps under these summaries, but severe regressions make its average benefit small and not reliably positive under the official weighting.

The benefit is principally better boundary placement and extent: vertical snap and interval coverage improve, while vertical-count accuracy falls from 99.923% to 92.340%. Eight regressions exceed 30 LBRS points and coincide with vertical-count collapse; the paired qualitative cases likewise show both improved alignment and damage to an otherwise strong raw reconstruction (Figures 9 and 10). Most measured structure is therefore already present in S4, making S5 an optional, non-uniform geometry adjustment rather than a necessary source of the main reconstruction gain.

The small S5–S4 effect is also weighting-sensitive: it is negative under count-heavy weighting and grows as localisation receives more weight (Table 4). As a robustness check, we additionally evaluate a geometry-balanced variant that retains 5% BBox weight and splits the remaining 95% equally between boundary-count recovery (47.5%) and geometry (47.5%: 23.75% Snap and 23.75% Dice). Under this deliberately more geometry-sensitive weighting, S3, S4, and S5 score 83.164, 87.591, and 93.242, respectively: the principal fine-tuning gain remains dominant, S4 still exceeds S3 by 4.427 points, and refinement benefits substantially because its strongest changes are geometric. The pre-specified official weighting remains primary; the balanced variant is a post-hoc robustness analysis rather than a replacement metric.

7 Conclusion

Tabstructor investigates explicit latent-border reconstruction for complex borderless tables and introduces the Latent Border Reconstruction Score (LBRS) as a bounded measure of valid structural and geometric border recovery. On the evaluated scientific-table test set, task-specific prompting improves the frozen base VLM, while supervised fine-tuning supplies the dominant reconstruction gain. The augmented training condition provides a further modest average improvement, although its additional cumulative exposure prevents attributing that gain to text diversity alone.

Downstream refinement of the primary raw fine-tuned system, S4, frequently improves boundary localisation and extent, but S5 has only a small, uncertain mean benefit and can introduce severe boundary-count regressions. Thus, in the evaluated setting, task-specific VLM adaptation performs the principal latent-border reconstruction, while refinement is an optional geometry adjustment rather than a necessary source of the main gain. These results establish latent borders as an explicit and measurable structural representation, but do not establish complete logical-cell topology, semantic understanding, downstream structured-format recovery, or cross-domain generalisation.

Limitations

The evidence is limited to the represented scientific-table distribution, modest development splits, and one held-out 100-table test set, so it does not establish performance on rare structures or other domains. We evaluate one LoRA-adapted model family and size, Qwen3.5-9B, with one run per fine-tuning condition; the paired intervals capture variation across test tables, not seeds, architectures, scales, or adaptation methods, and checkpoint choice may depend on the fixed 32-table

validation subset. Each table has one reviewed, adjudicated reference, although ambiguous layouts can admit other defensible geometries that single-reference scoring may penalise; labels were not independently duplicated. LBRS measures border geometry, not OCR accuracy, complete cell topology, semantic understanding, reasoning, or markup recovery. The S4–S3 contrast also combines one augmentation strategy with approximately 15.6% more training exposure, precluding attribution to text diversity or augmentation generally. Placement and extent errors remain in S4, while optional S5 adds OCR-dependent, rule-based processing that can over-regularise irregular boundaries; its small effect is non-uniform and sensitive to metric weighting.

Ethical Considerations

This study uses table images derived from published scientific literature, involves no research on human participants or collection of new personal data, and treats manual label creation and review as research annotation. Because access does not imply redistribution permission, cropped source images will be released only where source-specific review permits; project-created code and tools, derived annotations, split manifests, and provenance metadata will be released separately under applicable terms. Coverage of particular publishing styles and table structures may yield uneven reliability on other document types. Predictions can contain structural errors and establish neither OCR correctness nor the semantic or scientific validity of table contents, so accuracy-sensitive downstream applications should verify relevant outputs.

Acknowledgements

This work was supported by an Early Career Research Fellowship under the International Science Partnerships Fund (ISPF), delivered by the British Council, through the project “*Enhancing Environmental Resilience through AI-driven Analysis of Unstructured Data*.” The authors also acknowledge the RONIN cloud platform and the Research IT team at The University of Manchester for providing computational resources and technical support.

References

Iñigo Alonso, Imanol Miranda, Eneko Agirre, and Mirella Lapata. 2026. [TABLET: A large-scale data-](#)

[set for robust visual table understanding](#). In *Proceedings of the Fourteenth International Conference on Learning Representations (ICLR)*.

Boming Chen, Zining Wang, Zhentao Guo, Jianqiang Liu, Chen Duan, Yu Gu, Kai Zhou, and Pengfei Yan. 2026. [InstructTable: Improving table structure recognition through instructions](#). *arXiv preprint arXiv:2604.02880*.

Shuaiqi Duan, Yadong Xue, Weihang Wang, Zhe Su, Huan Liu, Sheng Yang, Guobing Gan, Guo Wang, Zihan Wang, Shengdong Yan, Dexin Jin, Yuxuan Zhang, Guohong Wen, Yanfeng Wang, Yutao Zhang, Xiaohan Zhang, Wenyi Hong, Yukuo Cen, Da Yin, and 4 others. 2026. [GLM-OCR technical report](#). *arXiv preprint arXiv:2603.10910*.

Khurram Azeem Hashmi, Didier Stricker, Marcus Liwicki, Muhammad Noman Afzal, and Muhammad Zeshan Afzal. 2021. [Guided table structure recognition through anchor optimization](#). *IEEE Access*, 9:113521–113534.

Xuanrui Hong, Kai Zha, Ziming Feng, Zhexuan Chen, Xiaodong Du, and Min Liu. 2024. [DAFA: A dual-awareness feature aggregator for table structure recognition on medical examination reports](#). *IEEE Access*, 12:1321–1329.

Minsoo Khang and Teakgyu Hong. 2025. [TFLOP: Table structure recognition framework with layout pointer mechanism](#). *arXiv preprint arXiv:2501.11800*.

Weihong Lin, Zheng Sun, Chixiang Ma, Mingze Li, Jiawei Wang, Lei Sun, and Qiang Huo. 2022. [TS-RFormer: Table structure recognition with transformers](#). In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 6473–6482.

Nam Quan Nguyen, Anh Duy Le, Anh Khoa Lu, Xuan Toan Mai, and Tuan Anh Tran. 2023. [Formerge: Recover spanning cells in complex table structure using transformer network](#). In *Document Analysis and Recognition—ICDAR 2023*, volume 14191 of *Lecture Notes in Computer Science*, pages 522–534. Springer Nature Switzerland.

Aditi Prajapati and Swati Maurya. 2026. [Table structure recognition: A review of current approaches and future opportunities](#). *IET Conference Proceedings*, 2025(43):86–91.

Wenyuan Xue, Baosheng Yu, Wen Wang, Dacheng Tao, and Qingyong Li. 2021. [TGRNet: A table graph reconstruction network for table structure recognition](#). In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1275–1284.

Chenglong Yu, Weibin Li, Zixuan Zhu, Wei Li, Jianchao Du, and Shiwei Zhang. 2025. [TableStructureFormer: An improved masked-attention mask transformer model with long-distance feature aggregation and deep detail supervision for table structure recognition](#). *Complex & Intelligent Systems*, 11(8):343.

Jinxu Zhang. 2024. [Read and think: An efficient step-wise multimodal language model for document understanding and reasoning](#). *arXiv preprint arXiv:2403.00816*.

Zhenrong Zhang, Pengfei Hu, Jiefeng Ma, Jun Du, Jianshu Zhang, Baocai Yin, Bing Yin, and Cong Liu. 2024. [SEMV2: Table separation line detection based on instance segmentation](#). *Pattern Recognition*, 149:110279.

A Reproducibility Statement

Section 3 and Appendix C document the data, split roles, annotation target, TeybolStruct protocol, and ambiguity policy. Appendix B preserves the frozen prompts and output protocol; Appendix D records the exact base-model revision and the LoRA, optimisation, augmentation, seed, environment, hardware, and runtime configurations; Section 5 defines S1–S5; and Appendices G, F, and E specify check-point selection, LBRS evaluation, and the GLM-OCR-assisted refinement from raw S4 to S5, respectively. Appendix H describes the planned machine-readable predictions and provenance records. Subject to licensing review, the planned project release covers fixed split manifests, derived annotations, selected adapters, and the annotation, training, inference, augmentation, evaluation, and refinement resources; cropped source images will be included only where redistribution is permitted.

B Frozen Prompt and Output Protocol

B.1 Minimal Prompt

The frozen textual instruction supplied with each table image in S1 was exactly:

```
Reconstruct the completed table borders from this scientific
table image.

Return exactly one valid JSON object and nothing else.

Use this exact schema:
{"table_bbox_norm": [x_min, y_min, x_max, y_max],
  "completed_borders": {"horizontal_segments_norm": [[x_start,
    x_end, y]], "vertical_segments_norm": [[x, y_start, y_end]]}}

The table bounding box coordinates must be normalized relative
to the full image.
The border-segment coordinates must be normalized relative to
the table bounding box.
Every coordinate must be between 0.000000 and 1.000000 and use
exactly six decimal places.

Infer the outer table border and the horizontal and vertical
separators required to represent the table.

Horizontal segments must use [x_start, x_end, y].
Vertical segments must use [x, y_start, y_end].

Sort horizontal segments from top to bottom, then left to
right.
Sort vertical segments from left to right, then top to bottom.

Return only the JSON object.
```

For both prompt conditions, the fixed message-role wrapper and exact image-token placement are specified by the frozen inference implementation.

B.2 Full Prompt

The frozen textual instruction supplied with each table image in S2, S3, and S4 was exactly:

```
Reconstruct the completed table borders from this scientific
table image.

Return exactly one valid JSON object and nothing else.
Do not return markdown, code fences, comments, OCR text, table
text, cell contents, or explanations.

Use this exact schema:
{"table_bbox_norm": [x_min, y_min, x_max, y_max],
  "completed_borders": {"horizontal_segments_norm": [[x_start,
    x_end, y]], "vertical_segments_norm": [[x, y_start, y_end]]}}

Coordinate rules:

* table_bbox_norm is normalized relative to the full image.
* completed_borders coordinates are normalized relative to the
  table bounding box.
* Every coordinate value must be a float from 0.000000 to
  1.000000.
* Use exactly six decimal places.
* Never output pixel coordinates.
* Never output image-size coordinates.
* Never output 0-100, 0-999, or 0-1000 coordinates.
* Any value greater than 1.000000 is invalid.

Task:
Complete the logical border structure of the table, not only
the visibly printed rules. Add the missing outer border,
column separators, row separators, partial header rules,
and merged-cell boundaries needed to express the
intended table topology.

Highest-priority geometry rules:
1. A predicted border is invalid if it crosses, touches, or
  visually overlaps table text glyphs.
2. Place horizontal row separators in the whitespace gap
  between adjacent visual text rows. Never place them on a
  text baseline, through descenders, through ascenders,
  through superscripts/subscripts, through punctuation, or
  through the middle of text.
3. Place vertical separators in the whitespace gap between
  adjacent text columns. Never place them through letters,
  digits, symbols, units, formulas, superscripts, or
  subscripts.
4. Geometry has priority over uniform spacing. Do not evenly
  distribute borders unless the text bands, alignment, and
  visible rules support that placement.
5. If a separator cannot be placed without touching text,
  choose the nearest existing printed rule or the widest
  available whitespace gap.
6. Do not duplicate nearby horizontal or vertical rules. If a
  visible printed rule already represents the logical
  boundary, use that boundary once.

Critical row-topology rules:
1. A horizontal separator is valid only where adjacent cells
  actually split at that y-position.
2. A full-width horizontal separator is valid only when every
  column participates in the same row split.
3. If a cell visually spans multiple subordinate rows, the
  separator between those subordinate rows must not pass
  through the spanning cell. Start or stop the separator
  at the column boundary of the cells that actually split.
4. Multiple printed lines inside one cell are not multiple
  rows. Do not draw a horizontal separator between wrapped
  lines, continued labels, units, formulas, coordinates,
  parenthetical text, or multi-line descriptions unless
  the surrounding columns also show a consistent row split.
5. Section-like, bold, italic, or standalone text does not
  automatically create a full-width merged row. Use a
  merged/full-width row only when the visual layout shows
  that the row owns the full table width.
6. Prefer partial horizontal segments over full-width
  horizontal segments when only part of the table splits
```

at that y-position. Annotation policy: 1. Preserve all table text. Borders may overlap existing printed rules, but must not cross, touch, or obscure text glyphs. 2. Use the existing top rule, header rule, bottom rule, alignment, spacing, and content ownership to infer the table bounding box and borders. 3. Do not default to a full rectangular internal grid. Use a complete grid only when the text alignment and visible structure show that all rows and columns share the same boundaries. 4. Do not create structure from typography alone. Bold, italic, superscripts, subscripts, significance marks, chemical notation, scientific symbols, and font weight are cell content only. 5. Wrapped or multi-line text inside one visual cell remains one cell. Do not add a horizontal border merely because a label, phrase, coordinate, unit, taxonomic name, parenthetical description, or long heading wraps onto multiple printed lines. 6. A new row exists only when there is a separate aligned record, a repeated row pattern, or a clear cell ownership change across columns. 7. Repeated prefixes, repeated units, repeated child labels, repeated statistic labels, or semantic similarity between columns do not create grouped headers unless the visual header layout shows a parent-child structure. 8. Isolated punctuation, dashes, or stray marks between header groups do not create columns or merged cells unless they align with a real body column or header hierarchy. 9. Shared unit, scale, normalization, or measurement annotations may appear as extra header rows beneath otherwise separate column headers. If such an annotation applies across multiple measurement columns, treat it as a merged header cell spanning only the applicable columns. 10. For grouped headers, preserve each visible parent-header underline as its own partial horizontal segment. Do not merge adjacent header underlines into one longer segment if there is a visual gap, a parent-group boundary, or a row-spanning header cell between them. 11. For hierarchical headers, parent headers span exactly their visible child columns. Child separators must stop at the parent-header boundary and restart only below the parent header. 12. Descriptor columns may rowspan across multiple header levels when they do not participate in grouped subheaders. 13. Repeated child labels under different parent headers are distinct columns. 14. Do not invent missing child columns under shortened header groups. 15. Unlabeled structural columns are valid. If a body column contains labels, terms, formulas, coefficient names, row descriptors, explanatory text, or statistic labels but its header cell is blank, preserve it as an explicit column. 16. If an unlabeled body column contains consistently aligned row descriptors, statistic labels, coefficients, category labels, units, or repeated label patterns, treat it as an explicit structural column even when its header cell is blank. 17. A header cell may span multiple underlying body columns even when no child header labels are printed. 18. Adjacent aligned components such as estimate, uncertainty, error, confidence interval, standard deviation, standard error, or significance-letter columns should be preserved as separate columns when they form consistent vertical alignments. 19. Section or explanatory rows should span the full table width only when they visually act as full-width section dividers, panel titles, block labels, or standalone descriptive rows. 20. Internal vertical borders must stop through full-width merged section rows and restart after them. 21. Do not merge a section-like label across the full table width merely because it is bold, italic, standalone, or followed by related rows. 22. Italic, bold, or standalone section-like labels do not automatically span the full table width. If the numeric columns remain part of the same row structure, keep the vertical column separators and treat the numeric-side cells as blank cells. 23. Panel header rows may combine a section or panel label in the first descriptor column with repeated column headers	in the remaining columns. 24. Section-like body rows may be local to a column group or panel rather than full-width. 25. A full-width section row may reset the local column structure below it. 26. Body rowspans are allowed when a label visually owns multiple subordinate rows, especially when repeated parent labels are omitted. 27. When descriptor cells such as region names, category names, sample counts, orchard counts, treatment labels, site labels, or group names visually own multiple subordinate rows, treat them as row-spanning cells. 28. Horizontal separators between subordinate rows must not pass through row-spanning descriptor cells. They should begin at the first column that actually splits into subordinate rows. 29. Body rowspans are not limited to descriptor columns. 30. Body records may contain nested subrows. Only the columns whose values change should split into subordinate rows. 31. Nested subrows may split independently by column. 32. Explicitly repeated values or row labels in separate aligned rows do not imply merging. 33. Empty cells, blank regions, placeholders, missing-value symbols, and missing-value text remain valid cells. 34. Sparse tables, triangular matrices, and partially populated regions must preserve the full latent grid. 35. Output only border segments that define logical cell topology. Keep outer borders, row separators, column separators, merged-cell boundaries, grouped-header boundaries, section/panel boundaries, and necessary partial segments. Remove decorative rules, duplicate nearby rules, stray marks, text underlines, and line fragments that do not separate cells or define ownership. 36. Preserve partial-line semantics. Output partial borders as separate segments. 37. Prefer the simplest valid topology that preserves visible grouping, row ownership, column ownership, merged cells, rowspans, panel-local grids, shared headers, and empty cells. Body-row placement rules: 1. For body rows, place each horizontal separator between the lower edge of the text in the row above and the upper edge of the text in the row below. 2. Do not place a separator through the center, baseline, descenders, superscripts, subscripts, or punctuation of either adjacent row. 3. For body rows with long descriptor text, parenthetical units, coordinates, soil names, taxonomic names, chemical notation, or wrapped descriptions, treat all wrapped lines as one cell unless the adjacent columns also show a consistent separate row split. 4. Do not add extra horizontal borders inside a single wrapped cell. 5. If a row contains blank cells in some columns, preserve those blank cells rather than merging the row unless a visual full-width section or rowspan is clearly indicated. Segment format: Horizontal segments must be [x_start,x_end,y]. Vertical segments must be [x,y_start,y_end]. Sorting: Sort horizontal segments from top to bottom, then left to right. Sort vertical segments from left to right, then top to bottom. Final coordinate check before answering: All numbers in table_bbox_norm, horizontal_segments_norm, and vertical_segments_norm must be between 0.000000 and 1.000000. Do not output any number greater than 1.000000. Do not output pixel coordinates. Return only the JSON object.
---	--

S5 has no additional VLM border-generation prompt; its refinement protocol is described in Section 4.3 and Appendix E.

B.3 Parsing and Validation

The evaluation expects the top-level keys `table_bbox_norm` and `completed_borders`; the latter must contain `horizontal_segments_norm` and `vertical_segments_norm`. The generated text is trimmed to the first complete JSON object before parsing. Parse success and schema validity are reported separately; both are required by the LBRS validity gate defined in Section 5.4. Table 6 reports the shared frozen decoding settings.

Table 6: Frozen vision–language model (VLM) decoding settings shared by the generation-producing systems S1–S4. Rows report the maximum output length in tokens, sampling status, beam count, and repetition penalty; S5 performs no additional VLM generation.

Setting	Value
Maximum new tokens	5,120
Sampling	Disabled
Number of beams	1
Repetition penalty	1.02

C TeybolStruct Annotation Protocol and Ambiguity Policy

C.1 Editor Representation

TeybolStruct represents the main-body target using a pixel-coordinate table box, complete horizontal and vertical boundary axes, bounded partial segments, and erased portions. The editor retains these objects and assisted evidence for reversible addition, removal, trimming, and erasure.

The target contains only `table_bbox_norm` and the `horizontal_segments_norm` and `vertical_segments_norm` arrays under `completed_borders`; OCR, cell text, dimensions, and debug fields are excluded. For image width W , height H , and pixel box $[l, t, r, b]$,

$$B_{\text{norm}} = \left[\frac{l}{W}, \frac{t}{H}, \frac{r}{W}, \frac{b}{H} \right], \quad (4)$$

while a horizontal pixel segment $[x_0, x_1, y]$ is converted to

$$\left[\frac{x_0 - l}{r - l}, \frac{x_1 - l}{r - l}, \frac{y - t}{b - t} \right], \quad (5)$$

while a vertical segment $[x, y_0, y_1]$ becomes

$$\left[\frac{x - l}{r - l}, \frac{y_0 - t}{b - t}, \frac{y_1 - t}{b - t} \right]. \quad (6)$$

Values are clamped to $[0, 1]$ and serialised to six decimal places.

C.2 Assisted Initialisation

Visible-line candidates are extracted by gray-scale conversion, adaptive mean thresholding, horizontal/vertical morphological opening, and the fixed length filters in Table 7. Margin-adjacent candidates may be supplied to the optional Qwen/Qwen3-VL-8B-Instruct assistant, which proposes only editable top, bottom, left, and right coordinates; it neither generates internal annotations nor participates in the five evaluated systems.

At export, detected rules exceeding the overlap threshold may enter the target. An erased portion marks its axis as non-complete, preventing automatic reintroduction as a full line; manual correction therefore takes precedence over assisted evidence.

C.3 Human Editing Procedure

The annotation procedure was:

1. load the extracted table image and establish or correct the outer table region;
2. inspect detected rule candidates and, when used, the editable four-boundary VLM proposal;
3. add, remove, or reposition complete horizontal and vertical boundaries;
4. use bounded partial segments, including erasures from otherwise complete axes, to represent local separators without dividing grouped, merged, or spanning regions;
5. trim overhangs and resolve near-duplicate coordinates;
6. export the normalised target and inspect the rendering generated from the exact validated JSON.

Figures 2 and 3 show this procedure sequentially on one representative example.

C.4 Export Validation and Quality Assurance

The exporter normalises the final pixel geometry, denormalises it, and blocks export if any reconstructed box or segment coordinate differs by more than one pixel. Validation also requires finite values in $[0, 1]$, increasing endpoints, exactly the expected keys, and full-width/full-height outer borders.

Before JSONL writing, the exact validated JSON is rendered for review. Export tests cover horizontal and vertical segmented-axis suppression and preservation of the outer rectangle.

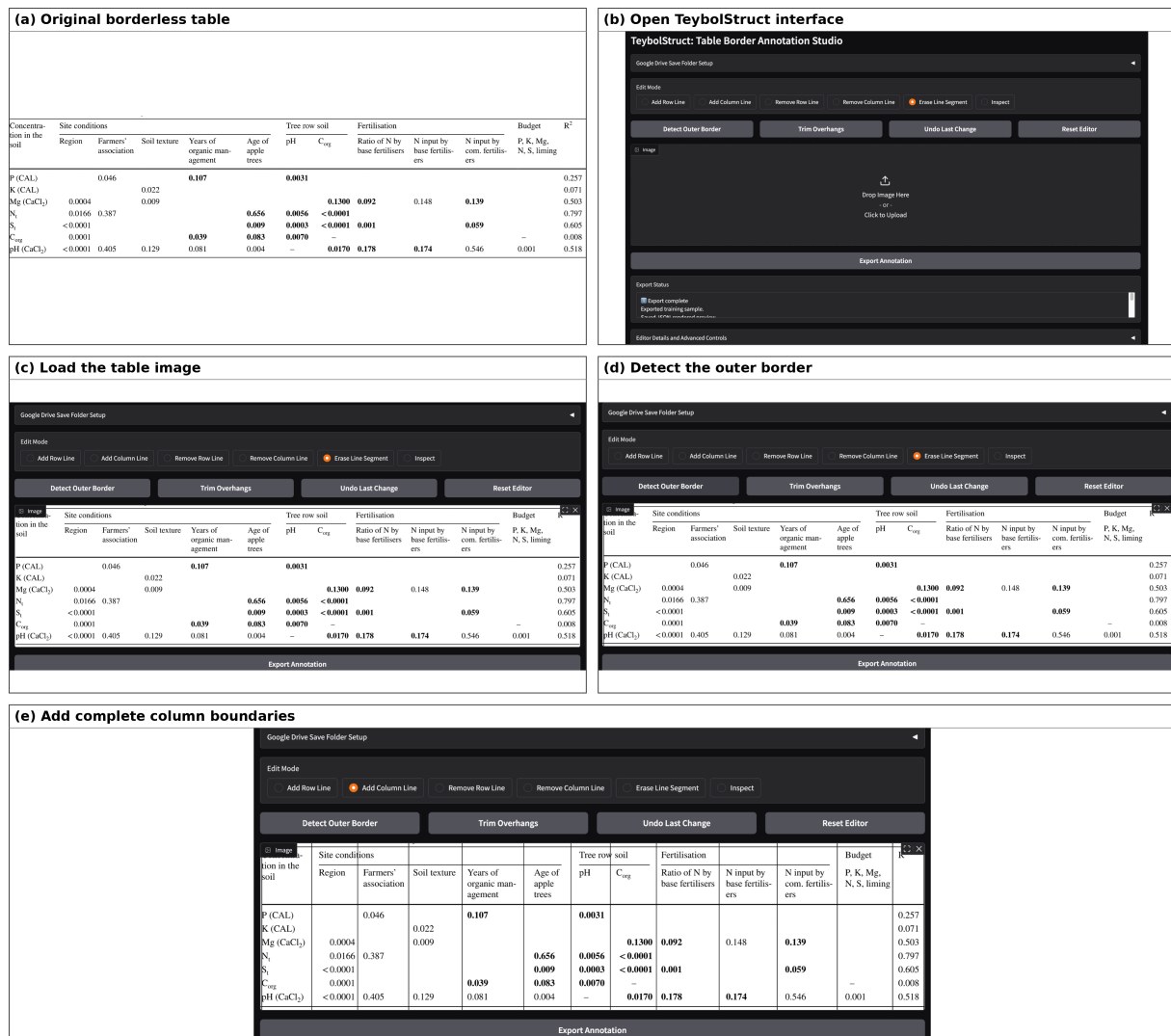


Figure 2: Early stages of the TeybolStruct annotation workflow on a representative borderless table: (a) the original table image; (b) the annotation interface; (c) the loaded image; (d) the detected and, if needed, corrected outer table border; and (e) complete column boundaries added only where they form valid full-table splits.

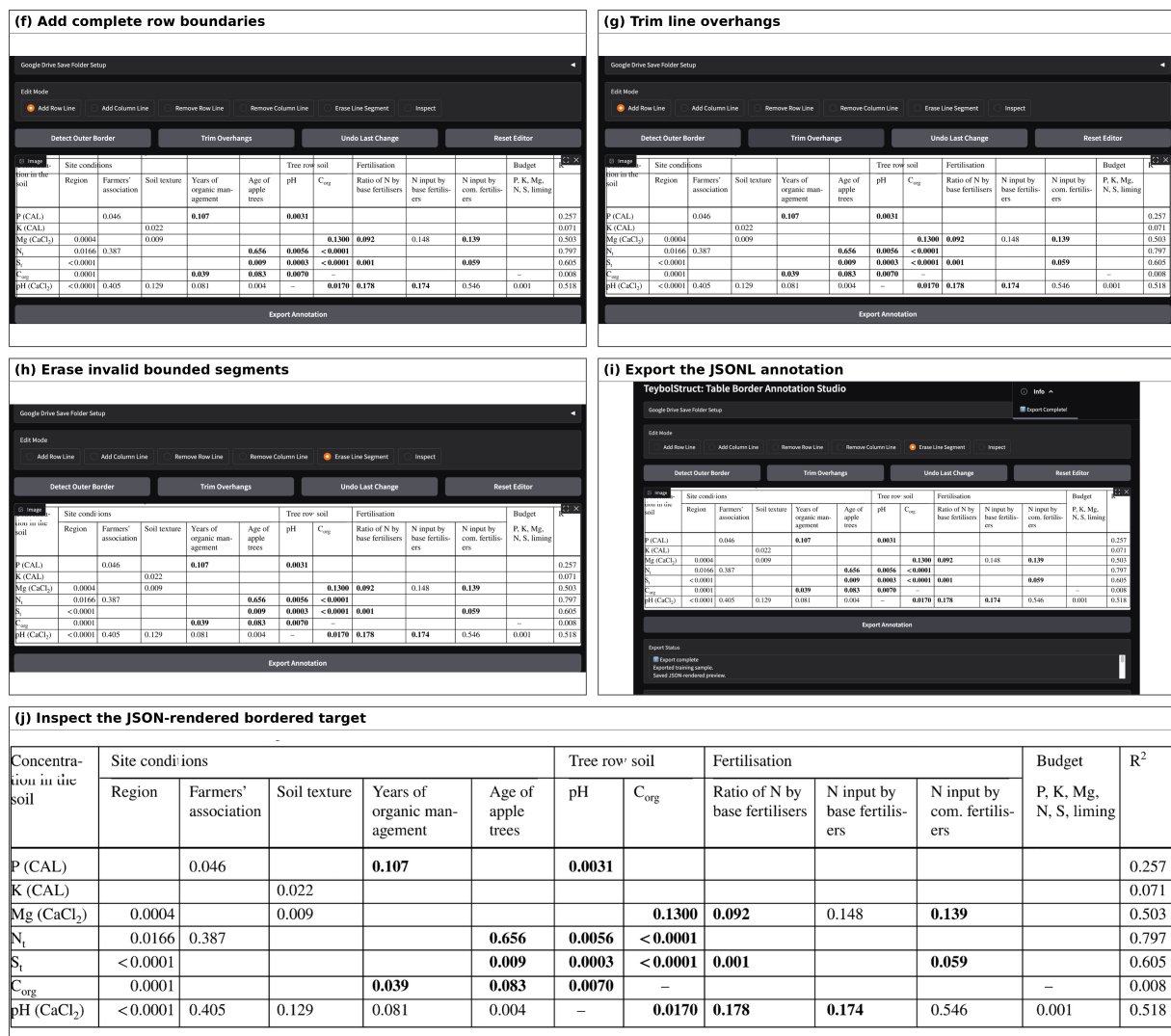


Figure 3: Later stages of the TeybolStruct annotation workflow on a representative borderless table: (f) add complete row boundaries; (g) trim line overhangs; (h) erase invalid bounded segments that would split grouped or spanning regions; (i) export the annotation as JSON Lines (JSONL); and (j) inspect the bordered-table rendering generated from the exported JSON. The panels proceed from geometry editing to export and final verification.

Table 7: Fixed TeybolStruct settings and values for the annotation workflow, grouped by line extraction, candidate filtering and merging, and export and geometry validation. Pixel-based values are reported in px; ratios are dimensionless.

Setting	Value
<i>Line extraction</i>	
Adaptive threshold block size	25
Adaptive threshold constant	15
Horizontal / vertical kernel divisor	30 / 30
Minimum detected length	15 px and 0.03 of image axis
<i>Candidate filtering & merging</i>	
Outer-candidate margin ratio	0.18
Outer-candidate minimum span ratio	0.25
Line-merge / duplicate tolerance	3 px
Partial-segment anchor snap tolerance	12 px
Erase-selection tolerance	8 px
<i>Export & geometry validation</i>	
Export detected-rule overlap ratio	0.20 horizontal / 0.20 vertical
Coordinate precision	6 decimal places
Maximum geometry round-trip error	1 px

One primary annotator prepared the labels, and each serialised-target rendering was reviewed before dataset admission. Ambiguity review and acceptance are specified below. Primary annotation averaged approximately five minutes per image (roughly 70 hours for 841 tables); correction and review time were not recorded separately.

C.5 Ambiguity and Adjudication

When multiple geometries are plausible, annotators select the simplest topology that preserves observable row and column ownership, grouped headers, spanning cells, panel-local structure, and empty structural fields. The decision rules are:

1. Use a complete separator only when every region along its extent participates in the same split; otherwise use the minimum bounded extent that expresses the split.
2. Treat aligned, repeated fields as structural even when their header or body entry is blank. Do not create a boundary from typography, whitespace, or a decorative rule alone.
3. Keep wrapped or multiline text in one cell unless consistent neighbouring evidence establishes another row or column. Where text obscures or interrupts an otherwise aligned boundary, join

the fragments only if the shared plane is supported on both sides and no content crosses the inferred corridor.

4. Preserve merged cells, rowspans, colspans, grouped headers, and local panels by omitting or bounding separators that would split them. A panel-local separator does not extend through neighbouring panels without evidence of the same split.
5. Use the minimal outer rectangle enclosing the logical table content and rules; exclude captions, footnotes, and decorative material that do not participate in the table grid.

These rules apply equally to fully borderless, partially ruled, and broken-rule regions: visible ink is evidence, whereas the target records the supported logical boundary. If the rules do not yield one defensible geometry, the case is escalated rather than resolved by visual regularity alone.

The primary annotator inspected the source and serialised-target rendering; unresolved cases were reviewed weekly and, if necessary, escalated to the biweekly full-team meeting. The agreed geometry was revised before admission, yielding one reference per table. This was collaborative review rather than independent duplicate annotation, so formal inter-annotator agreement is unavailable. External-test references were reviewed and frozen before S1–S5 evaluation without access to predictions, LBRS scores, or model-specific errors. Table 7 was fixed across all splits, but assisted settings never overrode manual correction and export validation.

D Training and Augmentation Configuration

This appendix records the matched fine-tuning configurations for S3 (non-augmented) and S4 (augmented). Shared settings are reported once; condition-specific data and run records are identified explicitly below.

D.1 Frozen Augmentation Procedure

For each selected source record, the released augmentation module executes the following ordered procedure:

1. Validate the source image path and completed-border JSON, then convert the image-relative table box and table-relative border segments to pixels.

2. Merge nearby axes, construct completed-border edge arrays, optionally repair outer edges, and recover cells by joining adjacent elementary rectangles whose shared border is absent.
3. Remove known rule pixels, skip near-empty crops, and prepare usable crops by border wiping, background normalisation, contrast-limited adaptive histogram equalisation (CLAHE), light denoising, and unsharp masking.
4. Transcribe cells in size-aware batches, recursively splitting CUDA out-of-memory batches and retrying irreducible crops at smaller scales. Normalise scripts without dataset-specific corrections and use geometric detection for missed dash-only placeholders.
5. Assign detected visual words to recovered cells, estimate base font size and cell-level boldness, and deterministically replace letters and digits with case-, width-, and shape-compatible alternatives while preserving punctuation, spacing, subscripts, and superscripts.
6. Mask safely assigned non-dash words while protecting rules, inpaint their pixels, and fit replacement strings to the original word boxes using the configured font-size and horizontal-scale ranges.
7. Accept a variant only if dimensions are unchanged, at least one safe replacement exists, the saved image and per-sample JSONL pass integrity checks, and the raw and parsed assistant targets exactly equal the source targets.

As described in Section 4.3, reference geometry is confined to this offline procedure and is not an inference input; exact target equality is enforced by the final acceptance check. Table 8 records the fixed pipeline parameters.

Here, *graph-guided* means that the recovered cell-adjacency graph—formed by joining adjacent elementary rectangles when their shared border is absent—guides cell assignment and text replacement; it is not a separate learned model.

Cell transcription in this offline augmentation procedure uses the same GLM-OCR model identified in the main text. This training-data preparation role is separate from its use for content-geometry validation in the S5 inference-time refinement stage.

Variant acceptance and rejection. A variant is rejected if validation, image loading, cell recovery,

Table 8: Fixed configuration of the offline typography-matched text-augmentation pipeline used to create S4 training variants. Rows report parameter–value pairs grouped by pipeline stage; OCR denotes optical character recognition, and px denotes pixels.

Parameter	Value
<i>Structural preparation</i>	
Axis merge tolerance	5 px
Edge coverage tolerance	8 px
Outer-edge repair	Enabled
Header rows	2
Row-header columns	1
<i>OCR preparation</i>	
OCR model	zai-org/GLM-OCR
OCR scale	2×
OCR padding	12 px
Minimum prepared crop	224 × 80 px
Maximum crop side	768 px
OCR token limit	64
OCR batch size	8
Padded-pixel limit	2,400,000
Preparation buffer	32
Minimum ink ratio	0.0012
Crop enhancement	Enabled
Dash fallback	Enabled
<i>Text and rule masking</i>	
Dark threshold	250
Erase threshold	253
Visual-word linking kernel	7 × 2 px
Minimum word ink	6 px
Normal-word height	6 px
Rule-protection thickness	4 px
Text-mask dilation	1 px
Inpainting radius	2 px
<i>Typography synthesis</i>	
Base font-size search range	14–52 px
Horizontal rendering scale range	0.70–1.35
<i>Reproducibility</i>	
Global seed	42
Variants per image	1

OCR/rendering, or a final integrity check fails. The run accepted 754 of 757 requested variants; Table 9 places these records in the matched S3–S4 data and model configuration, and the released rejection log records the three case-specific causes.

Table 10 reports the shared optimisation settings and the condition-specific run, termination, and compute records.

Token audit. Inputs reach at most 13,027 tokens and assistant targets at most 3,783 tokens; no record exceeds the respective 15,360-token sequence or 5,120-token generation limit.

Run identification. The retained manifests identify S3 with `augmentation_enabled=false`

Table 9: Matched data, model, and integrity configuration for S3 (unaugmented) and S4 (augmented) fine-tuning. S4 adds 754 accepted graph-guided, typography-matched variants to the 757 original training records; validation and test data remain unaugmented. Sequence and generation lengths are in tokens. LoRA, MLP, and rsLoRA denote Low-Rank Adaptation, multilayer perceptron, and rank-stabilized LoRA, respectively.

Configuration	Value
<i>S3-S4 data comparison</i>	
Training / validation split	757 / 84
S3 training records	757 total
S4 training records	757 originals + 754 accepted variants
S4 rejected variants	3
S3 augmentation	Disabled
S4 augmentation	Graph-guided, typography-matched text randomisation
<i>Shared model configuration</i>	
Base model	unsloth/Qwen3.5-9B
Model revision	005429cee5cb648998cf2b70eebdd83175989c9a
Adaptation	4-bit LoRA; full fine-tuning disabled
Gradient checkpointing	Unsloth
LoRA rank r	64
LoRA α	64
LoRA dropout	0.02
LoRA bias	none
rsLoRA	False
Trainable parameters	203,913,216 (3.43%)
Targeted modules	Vision, language, attention, and MLP
Sequence length	15,360
Generation allowance	5,120
<i>Integrity controls</i>	
Validation / test augmentation	Disabled
Ancestry and target checks	Passed

and S4 with `augmentation_enabled=true`; original manifests and hashes provide traceability.

Table 10: Shared optimisation settings, observed run and termination records, compute usage, and software versions for S3 (fine-tuned without augmentation) and S4 (fine-tuned with structure-preserving augmentation). Rows list configuration items and their values; shared settings are reported once, and slash-separated run-specific values are ordered S3/S4. GPU time and memory are reported in hours (h) and gigabytes (GB), respectively. Token loss is diagnostic only; final checkpoints are selected by validation Latent-Border Reconstruction Score (LBRS; higher is better).

Configuration	Value
<i>Shared optimisation settings</i>	
Optimizer	8-bit AdamW
Learning rate	1×10^{-4}
Weight decay	0.01
Learning-rate scheduler	Cosine
Maximum gradient norm	1.0
Per-device batch size	1
Gradient accumulation	8
Effective batch size	8
GPUs	1
Evaluation / saving interval	Every 25 optimizer steps
Logging interval	Every 5 optimizer steps
Early-stopping patience	40 evaluations
Early-stopping threshold	1×10^{-4}
Warm-up ratio	0.05
<i>Observed run records (S3 / S4)</i>	
Steps per epoch	95 / 189
Resolved warm-up steps	475 / 945
Run endpoint	1,475 / 1,700 optimizer steps
Endpoint epoch	15.528 / 8.995
Termination	Early stopping / training endpoint
Approx. training-record presentations	11,755 / 13,592
<i>Compute (S3 / S4)</i>	
GPU time	6.18 h / 7.15 h
Peak allocated GPU memory	36.02 GB / 36.03 GB
Peak reserved GPU memory	67.06 GB / 64.13 GB
<i>Software environment</i>	
Python	3.12.13
PyTorch / CUDA	2.8.0 / 12.8
Transformers / Transformers	5.3.0 / 0.22.2 /
Reinforcement Learning (TRL) / Parameter-Efficient Fine-Tuning (PEFT) packages	0.19.1

E S5 GLM-OCR-Assisted Geometry-Refinement Algorithm

Inputs and integrity constraints. The frozen implementation consumes the source image and

parsed JSON from validation-selected S4 checkpoint 1,425; it reads no reference borders, metrics, or external-test annotations. Within the predicted table box, candidate text-like components are validated by zai-org/GLM-OCR with `do_sample=False`. Border transformations are deterministic conditional on the accepted content geometry. The raw S4 JSONL remains unchanged, and the validated refinement is written separately for S5 evaluation.

Ordered refinement procedure. The executable notebook applies the following stages in exactly this order:

1. **Content geometry extraction.** Detect candidate text-like boxes within the predicted table bbox, validate them with the OCR model, refit accepted boxes to tight ink, and optionally retain sufficiently ink-dense unrecognised candidates as safety boxes.
2. **First-internal-column gap completion.** Complete missing pieces only on the first internal vertical plane and only between existing fragments; a proposed piece is blocked when validated content or tight ink crosses its corridor.
3. **First-internal-row dangling-column completion.** Repair the first internal horizontal plane within independent header blocks, extending to qualifying dangling verticals without crossing top-connected hard barriers.
4. **First-row vertical colspan trimming.** Use first-row content anchors to shorten or remove intervening verticals that would incorrectly split a spanning header cell.
5. **Horizontal alignment.** Snap planes to safe whitespace corridors between content bands without changing segment x-extents.
6. **Whole-segment horizontal pruning.** Remove unsupported, duplicate, or multiline-gap horizontal segments under conservative visible-rule and topology guards.
7. **Vertical alignment.** Move planes to text-safe content-gap midpoints without changing segment y-ranges.
8. **Horizontal-to-vertical junction repair.** Re-lock raw-topology horizontal connections to the aligned vertical plane.

9. **Reciprocal vertical endpoint row lock.** Re-lock raw-topology vertical connections to the retained final horizontal row.
10. **First-column rowspan trimming.** Remove only the first-column portion of selected horizontal separators when content occupancy indicates a descriptor cell spanning subordinate rows.
11. **First-column content-envelope refinement.** Use OCR-confirmed width plus actual ink to refine first-column separator extents and protect true content envelopes.
12. **First-column content-top completion.** Add or complete a first-column top separator when content and crossing guards support the missing boundary.
13. **Post-first-column auto-connect.** Canonicalise the first internal column, close compatible vertical gaps, and resynchronise nearby horizontal endpoints without crossing content.
14. **Second-column local trim/top completion.** Apply a conservative coverage-aware correction to the second column, including an optional missing top separator and immediate-separator trimming under long-empty-region guards.
15. **Outer-boundary enforcement.** Replace near-bbox fragments with the exact predicted-bbox edges and restore the normalised schema.

Pipeline pseudocode.

```
input: source image I, raw S4 JSON P=(B,H,V)
C, masks <- detect_text_like_candidates(I, B)
C <- OCR_validate_and_refit(C, I)
V <- complete_first_internal_column_gaps(H,V,B,C,masks)
H,V <- complete_first_internal_row_dangling(H,V,B)
V <- trim_first_row_verticals_by_content(H,V,B,C)
H <- snap_horizontal_to_content_gaps(H,B,C,masks)
H <- prune_unsupported_horizontal_segments(H,V,B,C,masks)
V <- align_vertical_to_content_gap_midpoints(V,B,C,masks,H)
H <- repair_horizontal_endpoints_to_aligned_verticals(H,V)
V <- lock_vertical_endpoints_to_final_horizontal_rows(V,H)
H <- trim_first_column_rowspan_separators(H,V,B,C)
H <- refine_first_column_by_content_envelopes(H,C,masks)
H <- complete_first_column_content_top_separators(H,C,masks)
H,V <- post_first_column_autoconnect(H,V,B,C,masks)
H <- second_column_local_trim_and_top_completion(H,V,B,C)
H,V <- enforce_bbox_as_outer_boundary(H,V,B)
output: separate refined JSONL; raw S4 JSONL remains unchanged
```

Frozen implementation parameters. All stages are enabled. The zai-org/GLM-OCR model uses 64 maximum new tokens, batch size 8, a maximum padded-batch budget of 2.4M pixels, minimum crop size 192×64 , and maximum crop side 640 px. Candidate boxes use 3 px recognition padding and 1 px tight geometry padding; candidates require at

Table 11: Fixed line-metric settings applied during validation-LBRS checkpoint selection and external-test evaluation. Rows report boundary-plane clustering and matching tolerances, snap-distance thresholds, and the metric implementation version; distances are in pixels (px).

Setting	Value
Boundary-plane clustering tolerance	3 px
Boundary-plane matching tolerance	30 px
Snap thresholds	5 / 10 / 20 / 30 px
Line-metric version	v1

least 4 ink pixels and 8 px^2 area, while unrecognised boxes are retained when ink ratio is at least 0.010. Horizontal and vertical plane grouping tolerances are 3 px. Outer-boundary matching/enforcement uses 6 px. Horizontal safe-corridor width is 3 px with 2 px content clearance; horizontal midpoint search uses 0.5 px steps. Horizontal pruning preserves visible rules at an overlap ratio of at least 0.50, uses a 2 px rule-search radius, a 0.15 minimum supported-x ratio, and 2–3 px duplicate plane/band tolerances. Vertical alignment uses 2 px content clearance, a 3 px minimum empty corridor, 0.5 px midpoint search, and a 3 px target-agreement tolerance. Junction and endpoint association tolerances are 8 px. First-column and second-column local rules use the exact constants in the frozen S5 parameter configuration distributed with the authoritative implementation.

Failure modes. A false OCR box may block a valid completion, whereas an OCR miss may expose an unsafe corridor. Pruning can remove a legitimate short separator, and local first/second-column rules can over-regularise unusual layouts.

F Complete LBRS Evaluation Procedure

Table 11 records the fixed line-metric settings used for both validation checkpoint selection and external-test evaluation.

This appendix specifies metric version `lbsr_interval_dice_no_bbox_v1`, which is the implementation used for checkpoint selection and the five-system evaluation. It defines the component formulae, algorithms, coordinate conventions, and edge cases underlying the compact score in Equation 2.

F.1 Validity Gate and Component Clipping

Let

$$\begin{aligned} P_i &= \mathbb{K}[\text{prediction } i \text{ is parsed as JSON}], \\ S_i &= \mathbb{K}[\text{prediction } i \text{ satisfies the schema}], \end{aligned} \quad (7)$$

so that $G_i = P_i S_i$. The schema requires exactly the top-level keys `table_bbox_norm` and `completed_borders`, whose border object contains the horizontal and vertical segment lists. Bbox and segment coordinates must be finite and in $[0, 1]$; bbox and segment endpoints must increase. Missing, non-numeric, or non-finite values are invalid, and any parse or schema failure sets $G_i = 0$.

The implementation applies

$$\text{clip}_{[0,1]}^*(x) = \begin{cases} 0, & x \text{ invalid,} \\ \min(1, \max(0, x)), & \text{otherwise,} \end{cases} \quad (8)$$

to every weighted component.

F.2 Coordinate Conversion

The table box is normalised relative to the complete image, whereas border-segment coordinates are normalised relative to that table box. Predicted and reference segments are each converted using their corresponding box. For an image of width W_i and height H_i , and a table box (l, t, r, b) , a table-relative coordinate (u, v) is mapped to pixels as

$$\begin{aligned} \pi_x(u) &= \text{round}[(l + u(r - l))(W_i - 1)], \\ \pi_y(v) &= \text{round}[(t + v(b - t))(H_i - 1)]. \end{aligned} \quad (9)$$

Thus, a horizontal segment $[u_1, u_2, v]$ becomes the pixel interval $[\pi_x(u_1), \pi_x(u_2)]$ on plane $\pi_y(v)$, while a vertical segment $[u, v_1, v_2]$ becomes $[\pi_y(v_1), \pi_y(v_2)]$ on plane $\pi_x(u)$. The four bbox edges are generated from (l, t, r, b) using the same $W_i - 1$ and $H_i - 1$ convention.

For predicted and reference image-normalised boxes $\hat{B}_i$ and B_i^* , table localisation is continuous-coordinate intersection over union,

$$B_i = \frac{|\hat{B}_i \cap B_i^*|}{|\hat{B}_i \cup B_i^*|}, \quad (10)$$

where non-overlapping boxes have zero intersection. Schema-valid boxes have positive area, so the union denominator is non-zero; invalid boxes are handled by the validity gate.

F.3 Boundary-Plane Extraction and Clustering

Broken, duplicated, or fragmented pieces on approximately the same axis are collapsed into one structural plane. For horizontal segments, the plane coordinate is y ; for vertical segments, it is x . Let $z_1 \leq \dots \leq z_m$ be the sorted finite plane coordinates. Sequential clustering assigns z_j to the current cluster $\mathcal{C}$ when

$$|z_j - \text{median}(\mathcal{C})| \leq \delta_{\text{cluster}}, \quad (11)$$

$$\delta_{\text{cluster}} = 3 \text{ px};$$

otherwise, it begins a new cluster; each cluster is represented by its median coordinate.

For count and snap-position scoring, the top and bottom bbox edges are added to the horizontal set and the left and right bbox edges to the vertical set before clustering. For interval Dice, only explicitly predicted and annotated segments are clustered; bbox-derived edges are excluded.

F.4 Boundary-Derived Counts

Let $P_{H,i}$ and $E_{H,i}$ be the predicted and expected numbers of bbox-inclusive horizontal planes, and let $P_{V,i}$ and $E_{V,i}$ be the corresponding vertical-plane counts. The boundary-derived row and column counts are

$$\begin{aligned} \hat{r}_i &= \max(0, P_{H,i} - 1), \\ r_i &= \max(0, E_{H,i} - 1), \\ \hat{c}_i &= \max(0, P_{V,i} - 1), \\ c_i &= \max(0, E_{V,i} - 1). \end{aligned} \quad (12)$$

For predicted and reference counts $\hat{n}$ and n , respectively, the count component is

$$S_{\text{count}}(\hat{n}, n) = \text{clip}_{[0,1]} \left(1 - \frac{|\hat{n} - n|}{\max(1, n)} \right). \quad (13)$$

The score is one when both counts are zero. These are boundary-derived rather than semantic row and column counts.

F.5 Greedy One-to-One Plane Matching

For predicted plane positions $\hat{\mathcal{Z}} = \{\hat{z}_1, \dots, \hat{z}_P\}$ and expected positions $\mathcal{Z} = \{z_1, \dots, z_E\}$, candidate pairs are

$$\mathcal{A} = \{(j, k) : |\hat{z}_j - z_k| \leq \delta_{\text{match}}\}, \quad (14)$$

$$\delta_{\text{match}} = 30 \text{ px}.$$

Candidates are sorted by ascending absolute distance; equal-distance candidates retain their deterministic lexicographic (j, k) enumeration order.

A pair is accepted only if both planes remain unused. This greedy, non-Hungarian procedure is applied independently by axis; unmatched planes remain unpaired.

F.6 Snap-Position Score and MAE

For a matched plane error d , the snap-readiness quality function is

$$q(d) = \begin{cases} 1, & 0 \leq d \leq 5, \\ 1 - 0.03(d - 5), & 5 < d \leq 10, \\ 0.85 - 0.025(d - 10), & 10 < d \leq 20, \\ 0.60 - 0.03(d - 20), & 20 < d \leq 30, \\ 0, & d > 30. \end{cases} \quad (15)$$

For axis $A \in \{H, V\}$, let $\mathcal{M}_{A,i}$ be the bbox-inclusive match set, $M_{A,i} = |\mathcal{M}_{A,i}|$, and $d_{m,i}^A$ the absolute pixel error of match m . When $P_{A,i} = E_{A,i} = 0$, the snap-position score is defined as one. Otherwise,

$$S_{A,i}^{\text{snap}} = \frac{\sum_{m=1}^{M_{A,i}} q(d_{m,i}^A)}{\max(P_{A,i}, E_{A,i})}. \quad (16)$$

This is $R_{s,i}$ for $A = H$ and $C_{s,i}$ for $A = V$. Extra or missing planes contribute zero through the denominator. Diagnostic MAE is the arithmetic mean of matched-plane errors and is unavailable when $M_{A,i} = 0$; unmatched planes do not enter it.

F.7 Interval Unions and Interval Dice

Explicit horizontal and vertical pieces become x - and y -intervals, respectively. Invalid or zero-length pieces are removed; sorted overlapping or endpoint-touching intervals are merged, so duplicate coverage counts once. For union U , $\mu(U)$ is its summed non-overlapping length, and $\mu(\hat{U} \cap U)$ is computed by a two-pointer traversal of the merged lists.

Explicit predicted and expected planes are greedily matched by the same 30-pixel positional rule. For a matched predicted/reference pair with interval unions $\hat{U}$ and U , the per-plane similarity is

$$D(\hat{U}, U) = \frac{2\mu(\hat{U} \cap U)}{\mu(\hat{U}) + \mu(U)}. \quad (17)$$

If both interval-union lengths are zero, the implementation defensively returns one, although zero-length pieces are normally removed beforehand.

Let $P_{A,i}^D$ and $E_{A,i}^D$ be the numbers of predicted and expected explicit planes, and $\mathcal{M}_{A,i}^D$ their match

set. When $P_{A,i}^D = E_{A,i}^D = 0$, the axis interval-Dice score is defined as one. Otherwise,

$$S_{A,i}^D = \frac{\sum_{(j,k) \in \mathcal{M}_{A,i}^D} D(\hat{U}_{j,i}^A, U_{k,i}^A)}{\max(P_{A,i}^D, E_{A,i}^D)}. \quad (18)$$

This gives $R_{d,i} = S_{H,i}^D$ and $C_{d,i} = S_{V,i}^D$; one-sided or otherwise unmatched explicit planes contribute zero.

F.8 Dataset Aggregation

For N tables, the official score is Equation 3, the unweighted mean of the per-table LBRS values. Each table therefore has equal influence regardless of image dimensions or number of predicted segments. Parse success, schema validity, bbox IoU, count, snap, and interval-Dice metrics are also reported as macro means over per-table values. Horizontal and vertical MAE are averaged only over tables for which the corresponding MAE is available.

The implementation also reports a non-official compatibility diagnostic obtained by inserting macro-averaged components into Equation 2; it may differ because the official gate is applied per table before averaging. The official weights and interpretation are given in Section 5.4; Table 4 reports the Results-section robustness audit.

F.9 Paired Statistical Reporting

For each of the four ordered system contrasts, the implementation computes the later-minus-earlier LBRS difference for every external-test table. It reports the mean, median, and standard deviation of those differences, along with the numbers that are positive, zero, or negative. Paired bootstrap 95% confidence intervals for the mean difference use 20,000 resamples of the 100 aligned table pairs and random seed 42. These intervals quantify variation across the evaluated tables; they do not estimate variability across independently trained models.

G Validation Checkpoint Selection

All checkpoints saved every 25 optimiser steps are evaluated with the same inference configuration on the fixed 32-table validation subset: 59 checkpoints at steps 25–1,475 for S3 and 68 at steps 25–1,700 for S4. Figures 5 and 6 show the two validation trajectories. The unique maximum unrounded macro LBRS is selected independently for each condition; no external-test prediction or score enters either ranking. Table 12 reports the

selected and runner-up checkpoints: checkpoint 525 for S3 (96.737, 0.153 points above checkpoint 1,050) and checkpoint 1,425 for S4 (97.107, 0.167 points above checkpoint 1,575).

Table 13 gives the five highest-scoring checkpoints for each condition.

The minimum validation losses are 0.066548 for S3 and 0.066539 for S4. Their checkpoints differ from the task-metric selections: step 475 for S3 obtains 95.022 validation LBRS (1.715 below checkpoint 525), while step 750 for S4 obtains 95.641 (rank 21; 1.466 below checkpoint 1,425). Thus, token loss is diagnostic rather than the structural ranking criterion. The supplementary package includes the complete 59-row and 68-row validation leaderboards and top-10 extracts as machine-readable CSV files, together with the checkpoint-ranking provenance.

H Extended Results, Per-Table Reporting, and Qualitative Audit

H.1 Complete Quantitative Records

The supplementary package provides full component and per-table external-test records for S1–S5, together with paired summaries and per-table differences for each adjacent contrast. The machine-readable external-test records include sample and system provenance; validity and bbox IoU; horizontal and vertical count, snap, interval-Dice, and MAE metrics; LBRS; the selected checkpoint; and geometry-refinement status. Figure 11 provides a compact per-table view. Appendix G documents the separate validation leaderboards.

H.2 Extended Aggregate and Paired Diagnostics

The following records preserve the complete aggregate, conditional-validity, distributional, and component-level evidence underlying the concise RQ summaries in Section 5.

Prompting diagnostics. Under S1, 45 outputs are unparsable and a further 21 fail the schema; the validity gate therefore assigns zero to 66 tables and yields a median LBRS of zero. Among the 34 valid predictions, mean bbox IoU is 97.544% and mean LBRS is 45.997, but horizontal-count recovery (20.853%) remains far below vertical-count recovery (74.428%). Thus, occasional strong geometry—including a best-table LBRS of 88.408—does not offset unreliable compliance and row-structure recovery.

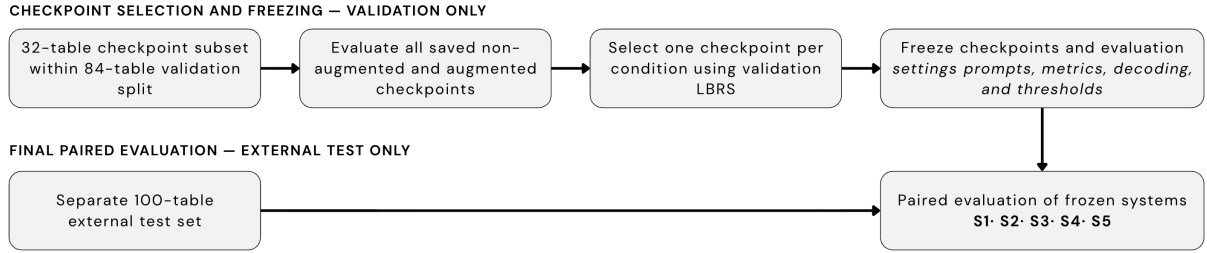


Figure 4: Validation-only checkpoint selection and external-test evaluation protocol. In the upper flow, all saved checkpoints from the non-augmented S3 and augmented S4 fine-tuning conditions are evaluated on a fixed 32-table subset of the 84-table validation split; one checkpoint per condition is selected using the Latent Border Reconstruction Score (LBRs), after which checkpoints, prompts, metrics, decoding settings, and thresholds are frozen. In the lower flow, the separate 100-table external test set is used only for paired evaluation of the frozen systems—prompting baselines S1–S2, fine-tuned systems S3–S4, and S5 refinement of S4. Arrows indicate the sequence and show that the frozen configurations and held-out test data meet only at final evaluation.

Table 12: Top-ranked and runner-up checkpoints for the S3 no-augmentation and S4 augmentation conditions, selected independently on the fixed 32-table validation subset using macro Latent Border Reconstruction Score (LBRs; 0–100, $\uparrow$ higher is better). Columns report the optimiser steps and LBRs values of the selected and runner-up checkpoints, the selected-minus-runner-up gap in LBRs points, and tie-break status. **Boldface** marks each selected step and score; both maxima are unique, so no tie-break was required.

Condition	Selected step	Val. LBRs $\uparrow$	Runner-up step	Runner-up LBRs	LBRs gap	Tie-break
S3: no augmentation	525	96.737	1,050	96.584	+0.153	Not required
S4: augmentation	1,425	97.107	1,575	96.940	+0.167	Not required

S2 reduces the gated-zero count to 26 (25 unparsable outputs and one additional schema failure) and raises median LBRs to 41.021. Among its 74 valid predictions, mean bbox IoU is 97.977% and mean LBRs is 52.098. The paired median gain over S1 is 22.832 points: 66 tables improve, 19 are unchanged, and 15 degrade. The validity transitions are 47 invalid-to-valid, 27 valid-to-valid, seven valid-to-invalid, and 19 invalid-to-invalid. Vertical interval Dice improves by only 2.467 points, so prompting recovers divisions more readily than their bounded extents. Conditional MAE is not directly comparable across systems with markedly different validity coverage. These diagnostics support the RQ1 prompting result in Section 5.

Fine-tuning diagnostics. Validation-selected checkpoint 525 (S3) has a median LBRs of 91.987, an interquartile range of 87.587–94.239, and a range of 55.242–99.356; 69 tables score at least 90, 20 at least 95, and nine below 80. Relative to S2, the median paired gain is 51.414 points: 98 tables improve and two degrade, and all 26 previously invalid outputs become valid while the other 74 remain valid. The complete component values are in Table 3 and the supplementary records; the distribution and validity transitions

substantiate the dominant fine-tuning effect reported for RQ1 in Section 5.

Augmentation diagnostics. The independently selected checkpoints are 525 for S3 and 1,425 for S4, using the same fixed validation protocol and matched documented configuration. Their training sets contain 757 original records and, for S4, 754 additional accepted text-randomised variants. Because the runs present approximately 11,755 and 13,592 records, respectively, this is an augmentation-with-additional-exposure comparison rather than an isolation of text diversity.

S4 has a median LBRs of 94.265, an interquartile range of 91.908–97.167, and a range of 63.486–99.787; 89 tables score at least 90, 43 at least 95, and four below 80. Relative to S3, the median gain is 1.793 points, with 69 improvements and 31 regressions. Vertical interval Dice supplies the largest component gain (+9.420 points). Horizontal MAE falls by 1.054 pixels, whereas vertical MAE rises by 2.371 pixels despite improved vertical snap and interval overlap; these metrics therefore capture complementary properties of matched boundaries. The aggregate RQ2 result and exposure qualification are given in Section 5.

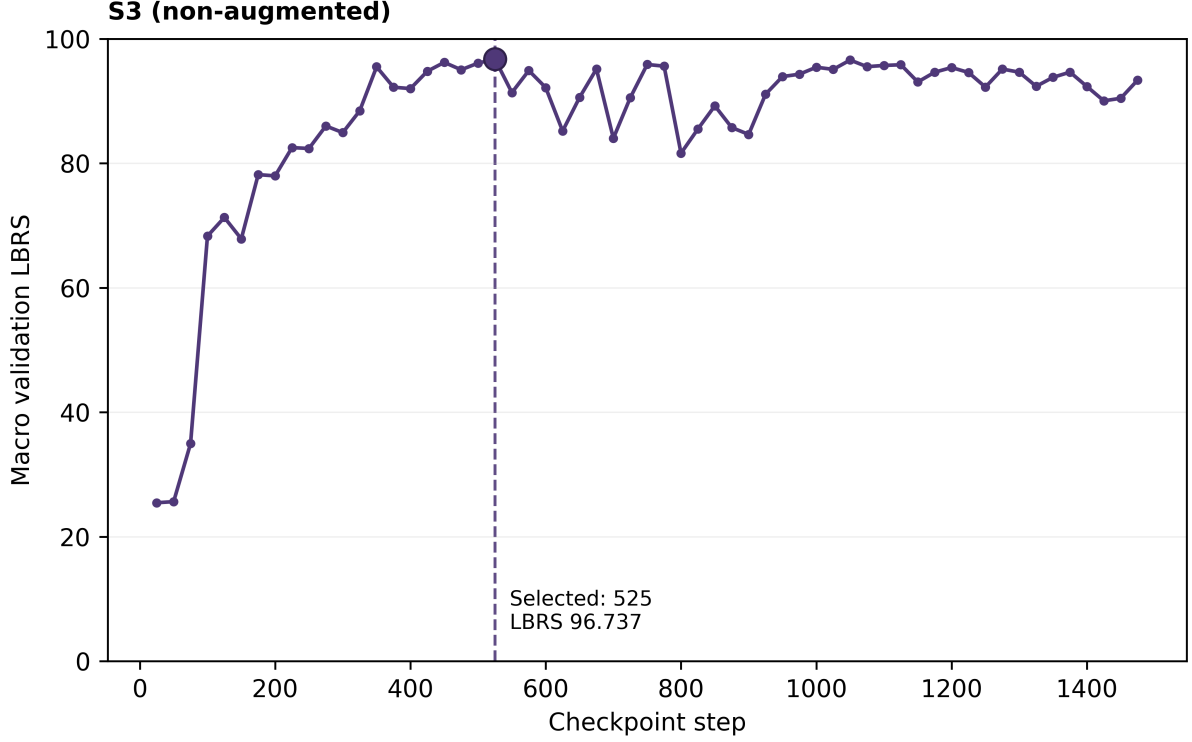


Figure 5: Validation checkpoint selection for S3, the non-augmented fine-tuned system, on the fixed 32-table validation subset. Points show macro Latent Border Reconstruction Score (LBRS; 0–100, higher is better) for 59 saved checkpoints at optimiser steps 25–1,475. The enlarged point and vertical dashed line mark checkpoint 525, selected as the unique maximum (LBRS 96.737); solid connecting lines guide visual tracing between discrete evaluations only.

Geometry-refinement diagnostics. The S5–S4 comparison is strictly paired because refinement operates on the exact raw S4 outputs. Although 90 tables improve and ten degrade, the paired median gain of 3.191 points contrasts with the mean gain of 0.467 reported in Section 5.

Refinement raises vertical snap-position from 49.374% to 82.512% and vertical interval Dice from 64.941% to 90.531%, while reducing vertical MAE from 13.886 to 7.770 pixels. The trade-off is a fall in vertical-count accuracy from 99.923% to 92.340%. Eight regressions exceed 30 LBRS points, each corresponding to a collapse in otherwise near-perfect S4 vertical-count recovery; Figure 10 shows the largest case. These diagnostics support the Section 5 interpretation of S4 as the primary predictor and S5 as a refinement analysis.

H.3 Qualitative Selection

Table 14 fixes one typical case, the two paired refinement extremes, and one complex-layout case, avoiding post-hoc visual selection. Figures 7, 8, 9, and 10 show these four cases. Each figure uses the same eight panels: input, reference, S1–S5,

and an S4→S5 operation view in which retained, moved or trimmed, added, and removed borders are distinguished.

H.4 Unified Per-Table S1–S5 Heatmap

Figure 11 shows the paired progression for all 100 external-test tables. Rows are ordered by descending S4 LBRS, concentrating the hardest raw-model cases at the bottom; the principal pattern is the broad fine-tuning gain followed by smaller, non-uniform refinement changes, including conspicuous regressions.

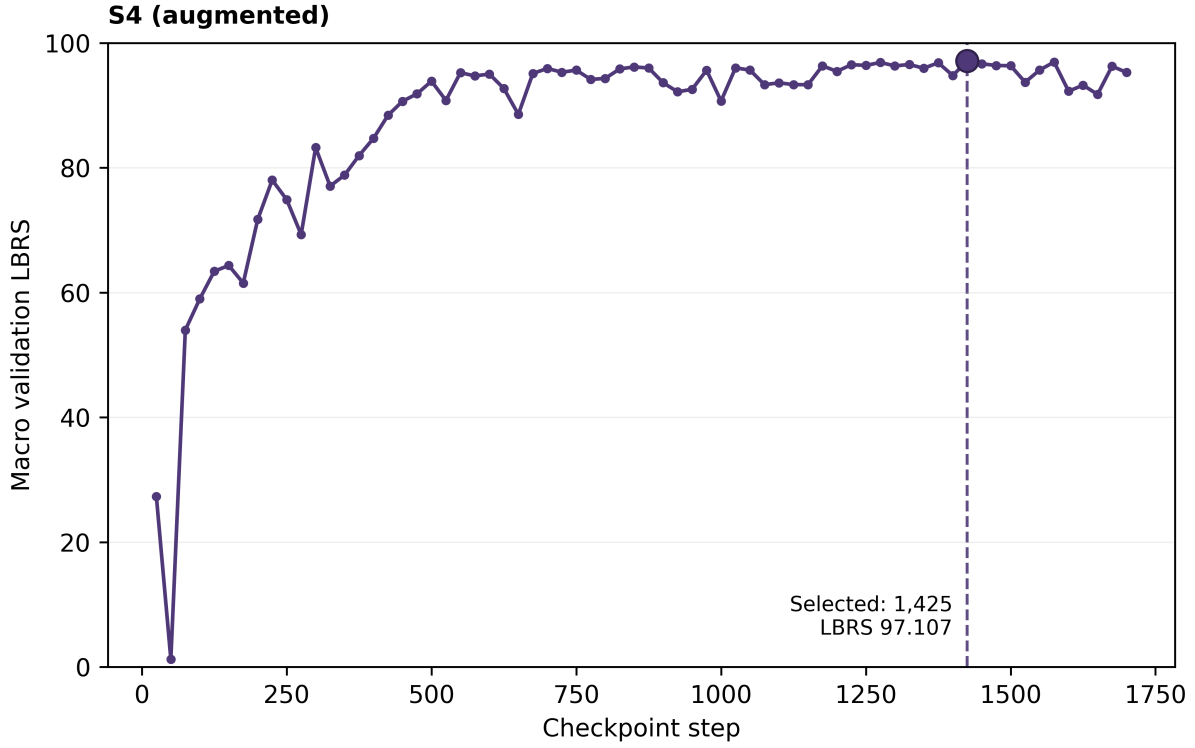


Figure 6: S4 checkpoint selection on the fixed 32-table validation subset. S4 is fine-tuned with structure-preserving augmentation; each point shows macro Latent Border Reconstruction Score (LBRS; 0–100, higher is better) for one of 68 checkpoints at steps 25–1,700. The enlarged point and vertical dashed line mark the unique maximum at step 1,425 (97.107); solid connecting lines are visual guides only.

Table 13: Five checkpoints with the highest macro Latent Border Reconstruction Score (LBRS; 0–100) for each fine-tuning condition—S3 (no augmentation) and S4 (augmentation)—on the fixed 32-table validation subset. Rows rank checkpoints independently within each condition; columns report the optimiser step, validation LBRS used for ranking, and macro diagnostic scores (%) for bounding-box intersection over union (BBox IoU) and boundary-derived row and column counts. $\uparrow$ indicates higher is better; **boldface** marks the selected checkpoint’s step and LBRS.

Rank	Step	Val. LBRS $\uparrow$	BBox IoU (%) $\uparrow$	Row count (%) $\uparrow$	Col. count (%) $\uparrow$
<i>S3: no augmentation</i>					
1	525	96.737	99.220	98.744	100.000
2	1,050	96.584	99.116	98.906	100.000
3	450	96.210	98.969	98.902	99.609
4	500	96.094	98.954	99.349	100.000
5	750	95.873	99.155	97.725	99.688
<i>S4: augmentation</i>					
1	1,425	97.107	98.855	99.349	100.000
2	1,575	96.940	99.001	99.141	100.000
3	1,275	96.873	98.890	99.010	100.000
4	1,375	96.828	98.859	98.718	100.000
5	1,450	96.658	98.910	98.828	99.760

Table 14: Selection criteria and analysis purposes for four qualitative audit cases from the fixed 100-table external test set. Rows give the case label, sample ID, selection criterion, and intended qualitative purpose, covering typical S5 performance, the largest positive and negative refinement changes, and a preselected complex layout. S4 is the unrefined augmented fine-tuned system, S5 applies GLM-OCR-assisted geometry refinement to S4, and LBRS denotes Latent-Border Reconstruction Score (0–100; higher is better).

Case	Sample	Selection criterion	Primary purpose
Q1	65	Closest to median S5 LBRS	Typical refined-system behaviour
Q2	89	Largest S5–S4 LBRS increase	Largest refinement improvement
Q3	63	Largest S5–S4 LBRS decrease	Largest refinement regression
Q4	6	Preselected complex-layout case	Grouped headers, spans, and local-panel structure

(a) Original borderless table									
Season	Treatment	N ₂ O emission (kg N ₂ O-N ha ⁻¹)		NO emission (kg NO-N ha ⁻¹)		N ₂ O emission factor (%)		NO emission factor (%)	
		Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation
The first season	Control	3.14 ± 0.18a	2.87 ± 0.05a	0.02 ± 0.01b	0.06 ± 0.01bc	0.25 ± 0.01b	4.24 ± 0.23a	3.56 ± 0.11 ± 0.01b	0.12 ± 0.01ab
	CF	11.20 ± 0.44a	11.34 ± 1.12a	0.23 ± 0.03b	0.25 ± 0.01b	0.01 ± 0.01c			
The second season	Control	0.25 ± 0.06c	0.15 ± 0.02c	0.03 ± 0.02b	0.01 ± 0.01c				
	CF	4.17 ± 0.40c	3.13 ± 0.01c	0.36 ± 0.04b	0.21 ± 0.01b	1.93 ± 0.09b	1.35 ± 0.01c	0.15 ± 0.02b	0.09 ± 0.01b
The third season	Control	1.03 ± 0.37b	0.80 ± 0.11b	0.16 ± 0.01a	0.15 ± 0.04a				
	CF	5.75 ± 0.13b	5.05 ± 0.09a	0.94 ± 0.23a	0.66 ± 0.05a	2.15 ± 0.06b	1.84 ± 0.21b	0.45 ± 0.05a	0.23 ± 0.02a
The fourth season	Control	0.83 ± 0.17bc	0.76 ± 0.10b	0.15 ± 0.02a	0.19 ± 0.01ab				
	CF	2.95 ± 0.27d	2.42 ± 0.40c	0.47 ± 0.13b	0.40 ± 0.14b	0.96 ± 0.12c	0.76 ± 0.13d	0.19 ± 0.04b	0.14 ± 0.02ab
(b) Ground-truth completed latent borders									
Season	Treatment	N ₂ O emission (kg N ₂ O-N ha ⁻¹)		NO emission (kg NO-N ha ⁻¹)		N ₂ O emission factor (%)		NO emission factor (%)	
		Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation
The first season	Control	3.14 ± 0.18a	2.87 ± 0.05a	0.02 ± 0.01b	0.06 ± 0.01bc	0.25 ± 0.01b	4.24 ± 0.23a	3.56 ± 0.11 ± 0.01b	0.12 ± 0.01ab
	CF	11.20 ± 0.44a	11.34 ± 1.12a	0.23 ± 0.03b	0.25 ± 0.01b	0.01 ± 0.01c			
The second season	Control	0.25 ± 0.06c	0.15 ± 0.02c	0.03 ± 0.02b	0.01 ± 0.01c				
	CF	4.17 ± 0.40c	3.13 ± 0.01c	0.36 ± 0.04b	0.21 ± 0.01b	1.93 ± 0.09b	1.35 ± 0.01c	0.15 ± 0.02b	0.09 ± 0.01b
The third season	Control	1.03 ± 0.37b	0.80 ± 0.11b	0.16 ± 0.01a	0.15 ± 0.04a				
	CF	5.75 ± 0.13b	5.05 ± 0.09a	0.94 ± 0.23a	0.66 ± 0.05a	2.15 ± 0.06b	1.84 ± 0.21b	0.45 ± 0.05a	0.23 ± 0.02a
The fourth season	Control	0.83 ± 0.17bc	0.76 ± 0.10b	0.15 ± 0.02a	0.19 ± 0.01ab				
	CF	2.95 ± 0.27d	2.42 ± 0.40c	0.47 ± 0.13b	0.40 ± 0.14b	0.96 ± 0.12c	0.76 ± 0.13d	0.19 ± 0.04b	0.14 ± 0.02ab
(c) S1: Base VLM, minimal prompt									
Parse unsuccessful / schema invalid No valid latent-border rendering available									
(d) S2: Base VLM, full prompt									
Season	Treatment	N ₂ O emission (kg N ₂ O-N ha ⁻¹)		NO emission (kg NO-N ha ⁻¹)		N ₂ O emission factor (%)		NO emission factor (%)	
		Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation
The first season	Control	3.14 ± 0.18a	2.87 ± 0.05a	0.02 ± 0.01b	0.06 ± 0.01bc	0.25 ± 0.01b	4.24 ± 0.23a	3.56 ± 0.11 ± 0.01b	0.12 ± 0.01ab
	CF	11.20 ± 0.44a	11.34 ± 1.12a	0.23 ± 0.03b	0.25 ± 0.01b	0.01 ± 0.01c			
The second season	Control	0.25 ± 0.06c	0.15 ± 0.02c	0.03 ± 0.02b	0.01 ± 0.01c				
	CF	4.17 ± 0.40c	3.13 ± 0.01c	0.36 ± 0.04b	0.21 ± 0.01b	1.93 ± 0.09b	1.35 ± 0.01c	0.15 ± 0.02b	0.09 ± 0.01b
The third season	Control	1.03 ± 0.37b	0.80 ± 0.11b	0.16 ± 0.01a	0.15 ± 0.04a				
	CF	5.75 ± 0.13b	5.05 ± 0.09a	0.94 ± 0.23a	0.66 ± 0.05a	2.15 ± 0.06b	1.84 ± 0.21b	0.45 ± 0.05a	0.23 ± 0.02a
The fourth season	Control	0.83 ± 0.17bc	0.76 ± 0.10b	0.15 ± 0.02a	0.19 ± 0.01ab				
	CF	2.95 ± 0.27d	2.42 ± 0.40c	0.47 ± 0.13b	0.40 ± 0.14b	0.96 ± 0.12c	0.76 ± 0.13d	0.19 ± 0.04b	0.14 ± 0.02ab
(e) S3: Selected non-augmented model									
Season	Treatment	N ₂ O emission (kg N ₂ O-N ha ⁻¹)		NO emission (kg NO-N ha ⁻¹)		N ₂ O emission factor (%)		NO emission factor (%)	
		Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation
The first season	Control	3.14 ± 0.18a	2.87 ± 0.05a	0.02 ± 0.01b	0.06 ± 0.01bc	0.25 ± 0.01b	4.24 ± 0.23a	3.56 ± 0.11 ± 0.01b	0.12 ± 0.01ab
	CF	11.20 ± 0.44a	11.34 ± 1.12a	0.23 ± 0.03b	0.25 ± 0.01b	0.01 ± 0.01c			
The second season	Control	0.25 ± 0.06c	0.15 ± 0.02c	0.03 ± 0.02b	0.01 ± 0.01c				
	CF	4.17 ± 0.40c	3.13 ± 0.01c	0.36 ± 0.04b	0.21 ± 0.01b	1.93 ± 0.09b	1.35 ± 0.01c	0.15 ± 0.02b	0.09 ± 0.01b
The third season	Control	1.03 ± 0.37b	0.80 ± 0.11b	0.16 ± 0.01a	0.15 ± 0.04a				
	CF	5.75 ± 0.13b	5.05 ± 0.09a	0.94 ± 0.23a	0.66 ± 0.05a	2.15 ± 0.06b	1.84 ± 0.21b	0.45 ± 0.05a	0.23 ± 0.02a
The fourth season	Control	0.83 ± 0.17bc	0.76 ± 0.10b	0.15 ± 0.02a	0.19 ± 0.01ab				
	CF	2.95 ± 0.27d	2.42 ± 0.40c	0.47 ± 0.13b	0.40 ± 0.14b	0.96 ± 0.12c	0.76 ± 0.13d	0.19 ± 0.04b	0.14 ± 0.02ab
(f) S4: Selected augmented model, raw									
Season	Treatment	N ₂ O emission (kg N ₂ O-N ha ⁻¹)		NO emission (kg NO-N ha ⁻¹)		N ₂ O emission factor (%)		NO emission factor (%)	
		Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation
The first season	Control	3.14 ± 0.18a	2.87 ± 0.05a	0.02 ± 0.01b	0.06 ± 0.01bc	0.25 ± 0.01b	4.24 ± 0.23a	3.56 ± 0.11 ± 0.01b	0.12 ± 0.01ab
	CF	11.20 ± 0.44a	11.34 ± 1.12a	0.23 ± 0.03b	0.25 ± 0.01b	0.01 ± 0.01c			
The second season	Control	0.25 ± 0.06c	0.15 ± 0.02c	0.03 ± 0.02b	0.01 ± 0.01c				
	CF	4.17 ± 0.40c	3.13 ± 0.01c	0.36 ± 0.04b	0.21 ± 0.01b	1.93 ± 0.09b	1.35 ± 0.01c	0.15 ± 0.02b	0.09 ± 0.01b
The third season	Control	1.03 ± 0.37b	0.80 ± 0.11b	0.16 ± 0.01a	0.15 ± 0.04a				
	CF	5.75 ± 0.13b	5.05 ± 0.09a	0.94 ± 0.23a	0.66 ± 0.05a	2.15 ± 0.06b	1.84 ± 0.21b	0.45 ± 0.05a	0.23 ± 0.02a
The fourth season	Control	0.83 ± 0.17bc	0.76 ± 0.10b	0.15 ± 0.02a	0.19 ± 0.01ab				
	CF	2.95 ± 0.27d	2.42 ± 0.40c	0.47 ± 0.13b	0.40 ± 0.14b	0.96 ± 0.12c	0.76 ± 0.13d	0.19 ± 0.04b	0.14 ± 0.02ab
(g) S5: S4 with geometry refinement									
Season	Treatment	N ₂ O emission (kg N ₂ O-N ha ⁻¹)		NO emission (kg NO-N ha ⁻¹)		N ₂ O emission factor (%)		NO emission factor (%)	
		Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation
The first season	Control	3.14 ± 0.18a	2.87 ± 0.05a	0.02 ± 0.01b	0.06 ± 0.01bc	0.25 ± 0.01b	4.24 ± 0.23a	3.56 ± 0.11 ± 0.01b	0.12 ± 0.01ab
	CF	11.20 ± 0.44a	11.34 ± 1.12a	0.23 ± 0.03b	0.25 ± 0.01b	0.01 ± 0.01c			
The second season	Control	0.25 ± 0.06c	0.15 ± 0.02c	0.03 ± 0.02b	0.01 ± 0.01c				
	CF	4.17 ± 0.40c	3.13 ± 0.01c	0.36 ± 0.04b	0.21 ± 0.01b	1.93 ± 0.09b	1.35 ± 0.01c	0.15 ± 0.02b	0.09 ± 0.01b
The third season	Control	1.03 ± 0.37b	0.80 ± 0.11b	0.16 ± 0.01a	0.15 ± 0.04a				
	CF	5.75 ± 0.13b	5.05 ± 0.09a	0.94 ± 0.23a	0.66 ± 0.05a	2.15 ± 0.06b	1.84 ± 0.21b	0.45 ± 0.05a	0.23 ± 0.02a
The fourth season	Control	0.83 ± 0.17bc	0.76 ± 0.10b	0.15 ± 0.02a	0.19 ± 0.01ab				
	CF	2.95 ± 0.27d	2.42 ± 0.40c	0.47 ± 0.13b	0.40 ± 0.14b	0.96 ± 0.12c	0.76 ± 0.13d	0.19 ± 0.04b	0.14 ± 0.02ab
(h) S4–S5 geometry-refinement operations									
Season	Treatment	N ₂ O emission (kg N ₂ O-N ha ⁻¹)		NO emission (kg NO-N ha ⁻¹)		N ₂ O emission factor (%)		NO emission factor (%)	
		Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation	Continuous cropping	Crop rotation
The first season	Control	3.14 ± 0.18a	2.87 ± 0.05a	0.02 ± 0.01b	0.06 ± 0.01bc	0.25 ± 0.01b	4.24 ± 0.23a	3.56 ± 0.11 ± 0.01b	0.12 ± 0.01ab
	CF	11.20 ± 0.44a	11.34 ± 1.12a	0.23 ± 0.03b	0.25 ± 0.01b	0.01 ± 0.01c			
The second season	Control	0.25 ± 0.06c	0.15 ± 0.02c	0.03 ± 0.02b	0.01 ± 0.01c				
	CF	4.17 ± 0.40c	3.13 ± 0.01c	0.36 ± 0.04b	0.21 ± 0.01b	1.93 ± 0.09b	1.35 ± 0.01c	0.15 ± 0.02b	0.09 ± 0.01b
The third season	Control	1.03 ± 0.37b	0.80 ± 0.11b	0.16 ± 0.01a	0.15 ± 0.04a				
	CF	5.75 ± 0.13b	5.05 ± 0.09a	0.94 ± 0.23a	0.66 ± 0.05a	2.15 ± 0.06b	1.84 ± 0.21b	0.45 ± 0.05a	0.23 ± 0.02a
The fourth season	Control	0.83 ± 0.17bc	0.76 ± 0.10b	0.15 ± 0.02a	0.19 ± 0.01ab				
	CF	2.95 ± 0.27d	2.42 ± 0.40c	0.47 ± 0.13b	0.40 ± 0.14b	0.96 ± 0.12c	0.76 ± 0.13d	0.19 ± 0.04b	0.14 ± 0.02ab

Figure 7: Qualitative latent-border reconstruction for Q4 (sample 6), the preselected complex-layout external-test case with grouped headers, row spans, and local partial boundaries. (a) Original borderless table; (b) ground-truth completed borders; and (c–g) outputs from S1–S5, respectively: base vision–language model (VLM) with minimal and full prompts, non-augmented and augmented fine-tuned models, and geometry-refined S4. Panel (c) has no rendering because the S1 output is unparsable or schema-invalid. In (d–f), red and blue denote horizontal and vertical predicted borders; in (g), green and cyan denote the refined horizontal and vertical borders. (h) S4-to-S5 operations: gray retained, orange moved or trimmed, green added, and red dashed removed. S4 recovers the principal topology, while S5 primarily adjusts geometry and preserves the spanning structure.

(a) Original borderless table					
Fruit type	Global average	China			
		N rate	Sample size	Survey area	Reference
Apple	126	702	n = 3366	Hebei, Shandong, Beijing, Shanxi, Shaanxi, Henan	Lu (2009)
Citrus	125	780	n = 483	Sichuan, Guangxi, Hubei	Lu (2009)
Grape	70	1142	n = 182	Hebei, Shandong, Beijing, Shanxi	Lu (2009)
Peach	155	598	n = 1477	Hebei, Shandong, Beijing, Shanxi, Anhui, Henan, Jiangsu, Shanghai, Sichuan	Lu (2009)
Pear	141	525	n = 414	Hebei, Shandong, Shanxi, Anhui	Lu (2009)
Watermelon	156	326	n = 295	Hebei, Henan, Gansu, Ningxia, Sichuan, Chongqing	Li (2013)
Banana	149	872		Fujian, Guangdong, Guangxi, Hainan, Yunnan	Zhao et al. (2020)
(c) S1: Base VLM, minimal prompt					
Parse unsuccessful / schema invalid No valid latent-border rendering available					
(e) S3: Selected non-augmented model					
Fruit type	Global average	China			
		N rate	Sample size	Survey area	Reference
Apple	126	702	n = 3366	Hebei, Shandong, Beijing, Shanxi, Shaanxi, Henan	Lu (2009)
Citrus	125	780	n = 483	Sichuan, Guangxi, Hubei	Lu (2009)
Grape	70	1142	n = 182	Hebei, Shandong, Beijing, Shanxi	Lu (2009)
Peach	155	598	n = 1477	Hebei, Shandong, Beijing, Shanxi, Anhui, Henan, Jiangsu, Shanghai, Sichuan	Lu (2009)
Pear	141	525	n = 414	Hebei, Shandong, Shanxi, Anhui	Lu (2009)
Watermelon	156	326	n = 295	Hebei, Henan, Gansu, Ningxia, Sichuan, Chongqing	Li (2013)
Banana	149	872		Fujian, Guangdong, Guangxi, Hainan, Yunnan	Zhao et al. (2020)
(g) S5: S4 with geometry refinement					
Final postprocessed horizontal and vertical borders (box outer boundary enforced) — green: horizontal; cyan: vertical					
Fruit type	Global average	China			
		N rate	Sample size	Survey area	Reference
Apple	126	702	n = 3366	Hebei, Shandong, Beijing, Shanxi, Shaanxi, Henan	Lu (2009)
Citrus	125	780	n = 483	Sichuan, Guangxi, Hubei	Lu (2009)
Grape	70	1142	n = 182	Hebei, Shandong, Beijing, Shanxi	Lu (2009)
Peach	155	598	n = 1477	Hebei, Shandong, Beijing, Shanxi, Anhui, Henan, Jiangsu, Shanghai, Sichuan	Lu (2009)
Pear	141	525	n = 414	Hebei, Shandong, Shanxi, Anhui	Lu (2009)
Watermelon	156	326	n = 295	Hebei, Henan, Gansu, Ningxia, Sichuan, Chongqing	Li (2013)
Banana	149	872		Fujian, Guangdong, Guangxi, Hainan, Yunnan	Zhao et al. (2020)
(b) Ground-truth completed latent borders					
Fruit type	Global average	China			
		N rate	Sample size	Survey area	Reference
Apple	126	702	n = 3366	Hebei, Shandong, Beijing, Shanxi, Shaanxi, Henan	Lu (2009)
Citrus	125	780	n = 483	Sichuan, Guangxi, Hubei	Lu (2009)
Grape	70	1142	n = 182	Hebei, Shandong, Beijing, Shanxi	Lu (2009)
Peach	155	598	n = 1477	Hebei, Shandong, Beijing, Shanxi, Anhui, Henan, Jiangsu, Shanghai, Sichuan	Lu (2009)
Pear	141	525	n = 414	Hebei, Shandong, Shanxi, Anhui	Lu (2009)
Watermelon	156	326	n = 295	Hebei, Henan, Gansu, Ningxia, Sichuan, Chongqing	Li (2013)
Banana	149	872		Fujian, Guangdong, Guangxi, Hainan, Yunnan	Zhao et al. (2020)
(d) S2: Base VLM, full prompt					
Fruit type	Global average	China			
		N rate	Sample size	Survey area	Reference
Apple	126	702	n = 3366	Hebei, Shandong, Beijing, Shanxi, Shaanxi, Henan	Lu (2009)
Citrus	125	780	n = 483	Sichuan, Guangxi, Hubei	Lu (2009)
Grape	70	1142	n = 182	Hebei, Shandong, Beijing, Shanxi	Lu (2009)
Peach	155	598	n = 1477	Hebei, Shandong, Beijing, Shanxi, Anhui, Henan, Jiangsu, Shanghai, Sichuan	Lu (2009)
Pear	141	525	n = 414	Hebei, Shandong, Shanxi, Anhui	Lu (2009)
Watermelon	156	326	n = 295	Hebei, Henan, Gansu, Ningxia, Sichuan, Chongqing	Li (2013)
Banana	149	872		Fujian, Guangdong, Guangxi, Hainan, Yunnan	Zhao et al. (2020)
(f) S4: Selected augmented model, raw					
Fruit type	Global average	China			
		N rate	Sample size	Survey area	Reference
Apple	126	702	n = 3366	Hebei, Shandong, Beijing, Shanxi, Shaanxi, Henan	Lu (2009)
Citrus	125	780	n = 483	Sichuan, Guangxi, Hubei	Lu (2009)
Grape	70	1142	n = 182	Hebei, Shandong, Beijing, Shanxi	Lu (2009)
Peach	155	598	n = 1477	Hebei, Shandong, Beijing, Shanxi, Anhui, Henan, Jiangsu, Shanghai, Sichuan	Lu (2009)
Pear	141	525	n = 414	Hebei, Shandong, Shanxi, Anhui	Lu (2009)
Watermelon	156	326	n = 295	Hebei, Henan, Gansu, Ningxia, Sichuan, Chongqing	Li (2013)
Banana	149	872		Fujian, Guangdong, Guangxi, Hainan, Yunnan	Zhao et al. (2020)
(h) S4–S5 geometry-refinement operations					
Fruit type	Global average	China			
		N rate	Sample size	Survey area	Reference
Apple	126	702	n = 3366	Hebei, Shandong, Beijing, Shanxi, Shaanxi, Henan	Lu (2009)
Citrus	125	780	n = 483	Sichuan, Guangxi, Hubei	Lu (2009)
Grape	70	1142	n = 182	Hebei, Shandong, Beijing, Shanxi	Lu (2009)
Peach	155	598	n = 1477	Hebei, Shandong, Beijing, Shanxi, Anhui, Henan, Jiangsu, Shanghai, Sichuan	Lu (2009)
Pear	141	525	n = 414	Hebei, Shandong, Shanxi, Anhui	Lu (2009)
Watermelon	156	326	n = 295	Hebei, Henan, Gansu, Ningxia, Sichuan, Chongqing	Li (2013)
Banana	149	872		Fujian, Guangdong, Guangxi, Hainan, Yunnan	Zhao et al. (2020)

Figure 8: Qualitative latent-border reconstruction for Q1 (external-test sample 65), selected as the case closest to the median S5 Latent Border Reconstruction Score (LBRS). (a) Original borderless table; (b) ground-truth completed borders; and (c–g) outputs from S1–S5, respectively: the base vision–language model (VLM) with minimal and full prompts, the non-augmented and augmented fine-tuned models, and geometry-refined S4. Panel (c) has no rendering because the S1 output is parse- or schema-invalid. In (d–f), red and blue denote predicted horizontal and vertical borders; in (g), green and cyan denote the refined horizontal and vertical borders. (h) S4-to-S5 refinement operations: gray retained, orange moved or trimmed, green added, and red dashed removed.

(a) Original borderless table				(b) Ground-truth completed latent borders			
Factor	Groups	R	P-value	Factor	Groups	R	P-value
Management	All (AS + ASD)	0.50	0.001	Management	All (AS + ASD)	0.50	0.001
Management	AS	0.33	0.001	Management	AS	0.33	0.001
Management	ASD	0.77	0.001	Management	ASD	0.77	0.001
Treatment	All (CT + NT)	0.07	0.024	Treatment	All (CT + NT)	0.07	0.024
Treatment	CT	−0.01	0.548	Treatment	CT	−0.01	0.548
Treatment	NT	0.28	0.001	Treatment	NT	0.28	0.001

(c) S1: Base VLM, minimal prompt				(d) S2: Base VLM, full prompt			
Parse unsuccessful / schemal invalid No valid latent-border rendering available				Factor	Groups	R	P-value
				Management	All (AS + ASD)	0.50	0.001
				Management	AS	0.33	0.001
				Management	ASD	0.77	0.001
				Treatment	All (CT + NT)	0.07	0.024
				Treatment	CT	−0.01	0.548
				Treatment	NT	0.28	0.001

(e) S3: Selected non-augmented model				(f) S4: Selected augmented model, raw			
Factor	Groups	R	P-value	Factor	Groups	R	P-value
Management	All (AS + ASD)	0.50	0.001	Management	All (AS + ASD)	0.50	0.001
Management	AS	0.33	0.001	Management	AS	0.33	0.001
Management	ASD	0.77	0.001	Management	ASD	0.77	0.001
Treatment	All (CT + NT)	0.07	0.024	Treatment	All (CT + NT)	0.07	0.024
Treatment	CT	−0.01	0.548	Treatment	CT	−0.01	0.548
Treatment	NT	0.28	0.001	Treatment	NT	0.28	0.001

(g) S5: S4 with geometry refinement				(h) S4-S5 geometry-refinement operations			
Factor	Groups	R	P-value	Factor	Groups	R	P-value
Management	All (AS + ASD)	0.50	0.001	Management	All (AS + ASD)	0.50	0.001
Management	AS	0.33	0.001	Management	AS	0.33	0.001
Management	ASD	0.77	0.001	Management	ASD	0.77	0.001
Treatment	All (CT + NT)	0.07	0.024	Treatment	All (CT + NT)	0.07	0.024
Treatment	CT	−0.01	0.548	Treatment	CT	−0.01	0.548
Treatment	NT	0.28	0.001	Treatment	NT	0.28	0.001

Factor	Groups	R	P-value	Factor	Groups	R	P-value
Management	All (AS + ASD)	0.50	0.001	Management	All (AS + ASD)	0.50	0.001
Management	AS	0.33	0.001	Management	AS	0.33	0.001
Management	ASD	0.77	0.001	Management	ASD	0.77	0.001
Treatment	All (CT + NT)	0.07	0.024	Treatment	All (CT + NT)	0.07	0.024
Treatment	CT	−0.01	0.548	Treatment	CT	−0.01	0.548
Treatment	NT	0.28	0.001	Treatment	NT	0.28	0.001

Figure 9: Qualitative latent-border reconstruction for case Q2 (external-test sample 89), selected for the largest S5–S4 improvement in Latent Border Reconstruction Score (LBRS; +12.406 points, higher is better). (a) Original borderless table; (b) ground-truth completed borders; and (c–g) outputs from S1–S5, respectively: the base vision-language model (VLM) with minimal and full prompts, the non-augmented and augmented fine-tuned models, and the geometry-refined S4 output. Panel (c) has no rendering because the S1 output is parse- or schema-invalid. In (d–f), red and blue denote predicted horizontal and vertical borders; in (g), green and cyan denote their refined counterparts. (h) S4-to-S5 operations: gray retained, orange moved or trimmed, green added, and red dashed removed.

(a) Original borderless table									
Treatments	GWP (kg CO ₂ -eq ha ⁻¹)			Yield GWP (kg CO ₂ -eq kg ⁻¹ yield)			NECB (kg C ha ⁻¹)		
	2018	2019	Mean	2018	2019	Mean	2018	2019	Mean
SC-N0	125.10e	125.08f	125.09f	0.13e	0.10e	0.11e	-873.69g	-322.74de	-598.22g
SC-N1	183.93d	198.88e	191.40e	0.17d	0.14d	0.16d	-721.74f	-396.76e	-559.25g
SC-N2	305.58b	332.81c	319.19c	0.28b	0.23b	0.25b	-694.71f	-215.75d	-455.23f
SC-N3	343.38a	401.94b	372.66b	0.30ab	0.26ab	0.28ab	-391.74e	-98.73cd	-245.24e
FO-N0	161.70d	152.07ef	156.88ef	0.16de	0.11de	0.13de	-38.67d	56.25c	8.79d
FO-N1	251.42c	279.63d	265.53d	0.22c	0.19c	0.20c	82.29c	72.29c	77.29c
FO-N2	270.06c	295.51cd	282.79cd	0.23c	0.19c	0.21c	195.37b	380.27b	287.82b
FO-N3	374.90a	458.17a	416.54a	0.32a	0.30a	0.31a	836.29a	897.27a	866.78a
Source of variance									
C	**	*		ns	ns	ns	**	**	**
N	**	**	**	**	**	**	**	ns	**
C×N	**	*	**	*	*	*	**	*	**

(b) Ground-truth completed latent borders									
Treatments	GWP (kg CO ₂ -eq ha ⁻¹)			Yield GWP (kg CO ₂ -eq kg ⁻¹ yield)			NECB (kg C ha ⁻¹)		
	2018	2019	Mean	2018	2019	Mean	2018	2019	Mean
SC-N0	125.10e	125.08f	125.09f	0.13e	0.10e	0.11e	-873.69g	-322.74de	-598.22g
SC-N1	183.93d	198.88e	191.40e	0.17d	0.14d	0.16d	-721.74f	-396.76e	-559.25g
SC-N2	305.58b	332.81c	319.19c	0.28b	0.23b	0.25b	-694.71f	-215.75d	-455.23f
SC-N3	343.38a	401.94b	372.66b	0.30ab	0.26ab	0.28ab	-391.74e	-98.73cd	-245.24e
FO-N0	161.70d	152.07ef	156.88ef	0.16de	0.11de	0.13de	-38.67d	56.25c	8.79d
FO-N1	251.42c	279.63d	265.53d	0.22c	0.19c	0.20c	82.29c	72.29c	77.29c
FO-N2	270.06c	295.51cd	282.79cd	0.23c	0.19c	0.21c	195.37b	380.27b	287.82b
FO-N3	374.90a	458.17a	416.54a	0.32a	0.30a	0.31a	836.29a	897.27a	866.78a
Source of variance									
C	**	*		ns	ns	ns	**	**	**
N	**	**	**	**	**	**	**	ns	**
C×N	**	*	**	*	*	*	**	*	**

(c) S1: Base VLM, minimal prompt									
Treatments	GWP (kg CO ₂ -eq ha ⁻¹)			Yield GWP (kg CO ₂ -eq kg ⁻¹ yield)			NECB (kg C ha ⁻¹)		
	2018	2019	Mean	2018	2019	Mean	2018	2019	Mean
SC-N0	125.10e	125.08f	125.09f	0.13e	0.10e	0.11e	-873.69g	-322.74de	-598.22g
SC-N1	183.93d	198.88e	191.40e	0.17d	0.14d	0.16d	-721.74f	-396.76e	-559.25g
SC-N2	305.58b	332.81c	319.19c	0.28b	0.23b	0.25b	-694.71f	-215.75d	-455.23f
SC-N3	343.38a	401.94b	372.66b	0.30ab	0.26ab	0.28ab	-391.74e	-98.73cd	-245.24e
FO-N0	161.70d	152.07ef	156.88ef	0.16de	0.11de	0.13de	-38.67d	56.25c	8.79d
FO-N1	251.42c	279.63d	265.53d	0.22c	0.19c	0.20c	82.29c	72.29c	77.29c
FO-N2	270.06c	295.51cd	282.79cd	0.23c	0.19c	0.21c	195.37b	380.27b	287.82b
FO-N3	374.90a	458.17a	416.54a	0.32a	0.30a	0.31a	836.29a	897.27a	866.78a
Source of variance									
C	**	*		ns	ns	ns	**	**	**
N	**	**	**	**	**	**	**	ns	**
C×N	**	*	**	*	*	*	**	*	**

(d) S2: Base VLM, full prompt									
Parse unsuccessful / schemal invalid No valid latent-border rendering available									

(e) S3: Selected non-augmented model									
Treatments	GWP (kg CO ₂ -eq ha ⁻¹)			Yield GWP (kg CO ₂ -eq kg ⁻¹ yield)			NECB (kg C ha ⁻¹)		
	2018	2019	Mean	2018	2019	Mean	2018	2019	Mean
SC-N0	125.10e	125.08f	125.09f	0.13e	0.10e	0.11e	-873.69g	-322.74de	-598.22g
SC-N1	183.93d	198.88e	191.40e	0.17d	0.14d	0.16d	-721.74f	-396.76e	-559.25g
SC-N2	305.58b	332.81c	319.19c	0.28b	0.23b	0.25b	-694.71f	-215.75d	-455.23f
SC-N3	343.38a	401.94b	372.66b	0.30ab	0.26ab	0.28ab	-391.74e	-98.73cd	-245.24e
FO-N0	161.70d	152.07ef	156.88ef	0.16de	0.11de	0.13de	-38.67d	56.25c	8.79d
FO-N1	251.42c	279.63d	265.53d	0.22c	0.19c	0.20c	82.29c	72.29c	77.29c
FO-N2	270.06c	295.51cd	282.79cd	0.23c	0.19c	0.21c	195.37b	380.27b	287.82b
FO-N3	374.90a	458.17a	416.54a	0.32a	0.30a	0.31a	836.29a	897.27a	866.78a
Source of variance									
C	**	*		ns	ns	ns	**	**	**
N	**	**	**	**	**	**	**	ns	**
C×N	**	*	**	*	*	*	**	*	**

(f) S4: Selected augmented model, raw									
Treatments	GWP (kg CO ₂ -eq ha ⁻¹)			Yield GWP (kg CO ₂ -eq kg ⁻¹ yield)			NECB (kg C ha ⁻¹)		
	2018	2019	Mean	2018	2019	Mean	2018	2019	Mean
SC-N0	125.10e	125.08f	125.09f	0.13e	0.10e	0.11e	-873.69g	-322.74de	-598.22g
SC-N1	183.93d	198.88e	191.40e	0.17d	0.14d	0.16d	-721.74f	-396.76e	-559.25g
SC-N2	305.58b	332.81c	319.19c	0.28b	0.23b	0.25b	-694.71f	-215.75d	-455.23f
SC-N3	343.38a	401.94b	372.66b	0.30ab	0.26ab	0.28ab	-391.74e	-98.73cd	-245.24e
FO-N0	161.70d	152.07ef	156.88ef	0.16de	0.11de	0.13de	-38.67d	56.25c	8.79d
FO-N1	251.42c	279.63d	265.53d	0.22c	0.19c	0.20c	82.29c	72.29c	77.29c
FO-N2	270.06c	295.51cd	282.79cd	0.23c	0.19c	0.21c	195.37b	380.27b	287.82b
FO-N3	374.90a	458.17a	416.54a	0.32a	0.30a	0.31a	836.29a	897.27a	866.78a
Source of variance									
C	**	*		ns	ns	ns	**	**	**
N	**	**	**	**	**	**	**	ns	**
C×N	**	*	**	*	*	*	**	*	**

(g) S5: S4 with geometry refinement									
Treatments	GWP (kg CO ₂ -eq ha ⁻¹)			Yield GWP (kg CO ₂ -eq kg ⁻¹ yield)			NECB (kg C ha ⁻¹)		
	2018	2019	Mean	2018	2019	Mean	2018	2019	Mean
SC-N0	125.10e	125.08f	125.09f	0.13e	0.10e	0.11e	-873.69g	-322.74de	-598.22g
SC-N1	183.93d	198.88e	191.40e	0.17d	0.14d	0.16d	-721.74f	-396.76e	-559.25g
SC-N2	305.58b	332.81c	319.19c	0.28b	0.23b	0.25b	-694.71f	-215.75d	-455.23f
SC-N3	343.38a	401.94b	372.66b	0.30ab	0.26ab	0.28ab	-391.74e	-98.73cd	-245.24e
FO-N0	161.70d	152.07ef	156.88ef	0.16de	0.11de	0.13de	-38.67d	56.25c	8.79d
FO-N1	251.42c	279.63d	265.53d	0.22c	0.19c	0.20c	82.29c	72.29c	77.29c
FO-N2	270.06c	295.51cd	282.79cd	0.23c	0.19c	0.21c	195.37b	380.27b	287.82b
FO-N3	374.90a	458.17a	416.54a	0.32a	0.30a	0.31a	836.29a	897.27a	866.78a
Source of variance									
C	**	*		ns	ns	ns	**	**	**
N	**	**	**	**	**	**	**	ns	**
C×N	**	*	**	*	*	*	**	*	**

(h) S4-S5 geometry-refinement operations									
Treatments	GWP (kg CO ₂ -eq ha ⁻¹)			Yield GWP (kg CO ₂ -eq kg ⁻¹ yield)			NECB (kg C ha ⁻¹)		
	2018	2019	Mean	2018	2019	Mean	2018	2019	Mean
SC-N0	125.10e	125.08f	125.09f	0.13e	0.10e	0.11e	-873.69g	-322.74de	-598.22g
SC-N1	183.93d	198.88e	191.40e	0.17d	0.14d	0.16d	-721.74f	-396.76e	-559.25g
SC-N2	305.58b	332.81c	319.19c	0.28b	0.23b	0.25b	-694.71f	-215.75d	-455.23f
SC-N3	343.38a	401.94b	372.66b	0.30ab	0.26ab	0.28ab	-391.74e	-98.73cd	-245.24e
FO-N0	161.70d	152.07ef	156.88ef	0.16de	0.11de	0.13de	-38.67d	56.25c	8.79d
FO-N1	251.42c	279.63d	265.53d	0.22c	0.19c	0.20c	82.29c	72.29c	77.29c
FO-N2	270.06c	295.51cd	282.79cd	0.23c	0.19c	0.21c	195.37b	380.27b	287.82b
FO-N3	374.90a	458.17a	416.54a	0.32a	0.30a	0.31a	836.29a	897.27a	866.78a
Source of variance									
C	**	*		ns	ns	ns	**	**	**
N	**	**	**	**	**	**	**	ns	**
C×N	**	*	**	*	*	*	**	*	**

Figure 10: Qualitative latent-border reconstruction for Q3 (external-test sample 63), selected for the largest S5–S4 decrease in Latent Border Reconstruction Score (LBRS; −41.802 points on a 0–100 scale, higher is better). (a) Original borderless table; (b) ground-truth completed borders; and (c–g) outputs from S1–S5, respectively: the base vision–language model (VLM) with minimal and full prompts, the non-augmented and augmented fine-tuned models, and the geometry-refined S4 output. Panel (d) has no rendering because the S2 output is unparsable or schema-invalid. In (c), (e), and (f), red and blue denote predicted horizontal and vertical borders; in (g), green and cyan denote their refined counterparts. (h) S4-to-S5 refinement operations: gray retained, orange moved or trimmed, green added, and red dashed removed.

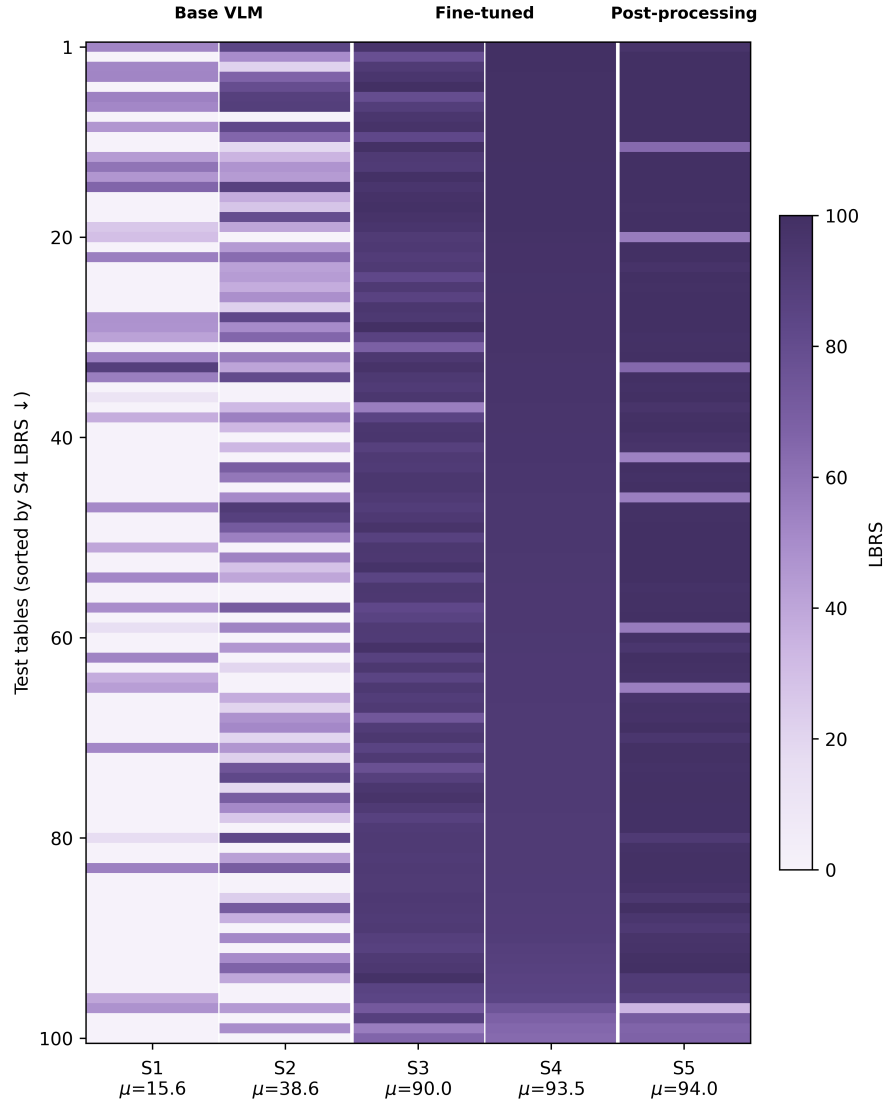


Figure 11: Per-table Latent Border Reconstruction Score (LBRS; 0–100, higher is better) for five systems on the 100-table external test set. Columns show prompting-only base vision–language model (VLM) baselines S1–S2, fine-tuned systems S3–S4, and S5 post-processing of S4; each cell is one table’s score, with darker purple indicating higher LBRS. Rows align the same table across systems and are sorted from top to bottom by decreasing S4 score. μ denotes each system’s mean over all 100 tables, and paired S4–S5 cells show the effect of refinement on each S4 output.