How Deep Is Your Value? Table-Cell Operands Let Go of VLM outputs Earlier Than Answers

Anonymous authors
Paper under double-blind review

Abstract

Vision-language models (VLMs) answer questions about tables directly from document page images, yet final-answer accuracy does not reveal how the visual evidence of a cell is processed inside the model. This matters because a cell's value can serve two roles: in LOOKUP, it is read out as the answer; in COMPUTE, it is used as an operand with values from other cells, often on other pages. A model could process every cell to the same depth regardless of its role, or release an operand once it has been used. We construct paired LOOKUP and COMPUTE instances that share a table, a target cell, and a pixel edit to that cell. In eight VLMs, we first verify that the cell's visual tokens carry the value used to construct the answer, and then intervene on these tokens layer by layer. We find that operands let go of the output earlier than answers: replacing the cell's value redirects the output only up to an earlier layer in COMPUTE than in LOOKUP, in all eight models under a log-probability criterion and in six under full generation. The difference lies in the value, not in the cell's presence: the depth at which removing the cell stops mattering is set largely by the model and barely moves across question types or page structures. An operand's value remains linearly decodable at the cell after its influence ends, but later text positions carry the computed result rather than an operand the model can recompute from. Code used for the analysis is publicly available.[1]

1 Introduction

Tables organize numerical values and their relationships through a row-and-column structure, making them a common form of structured evidence. As such tables are usually available as page images in practice, table visual question answering in documents (TabVQADoc), which asks the model to answer from the rendered page directly, has recently emerged as an active research area (Huang et al., 2026; Kim et al., 2024; Mathew et al., 2021). TabVQADoc requires first identifying question-relevant cells in a document image, and then understanding their context through the structural relationships between rows and columns.

Vision-language models (VLMs) approach this by visually reading table structure and numerical content from the page image itself without a separate OCR pipeline (Indrehus et al., 2026; Chia et al., 2025; Souibgui et al., 2025; Zheng et al., 2024; Lee et al., 2023; Kim et al., 2022). Once a relevant cell is located, its value can serve one of two roles depending on the query. It can be the answer itself, as in a query that asks for a single entry; or it can be an operand, an intermediate quantity that must be combined with values from other cells, often on different pages, before the answer exists. Yet VLMs answer less accurately when a table is given as page images than when the same content is given as text, and the accuracy degrades further as table structures become more complex (Huang et al., 2026; Han et al., 2026; Yang et al., 2026). Final-answer accuracy, however, cannot tell the two roles apart. Because table evidence is packed into small cells with visually similar neighbors, a correct answer does not reveal whether the model carried the cell's value to the output or consumed it in a computation, nor how deep into the model either happens.

In this study, we ask whether seeing a value and using it engage the model differently. We say a value is *seen* when it is read out as the answer, as in LOOKUP, and *used* when the model is acting on it, combining it with other values to produce the answer, as in COMPUTE. A seen value is the

[1] https://anonymous.4open.science/r/vlm-table-cell-layer-analysis-67D8/

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

answer, and its representation must be carried through the remaining layers to the output. A used value is needed only until the computation consumes it; what must be carried forward is the result. The same cell, rendered by the same pixels, imposes two different demands, and a single model must accommodate both with the same set of weights. If the model meets these demands differently, seeing and using should differ in how deep into the decoder the cell's value continues to influence the prediction.

Three lines of recent work point in this direction. If the effective depth of visual tokens varies across tasks (Wang et al., 2026), it should also vary across roles within a single table image, where the evidence is held fixed and only the demand on it changes. If operands are read early and their information relocated to later token positions (Mamidanna et al., 2025; Stolfo et al., 2023), a table cell serving as an operand should stop mattering at its original site sooner than the same cell serving as an answer. And if information can remain linearly decodable at layers where it no longer drives the output (Balef et al., 2026; Skean et al., 2025), a cell's value may still be readable after its visual site has ceased to influence the prediction.

Whether these observations carry over to table cells, however, remains unclear. Causal tracing has been extended to VLMs to study object-level representations (Li et al., 2026), but table cells differ in an important way: each cell is a small region containing a precise value. A model could simply process every cell to a fixed depth determined by its architecture, regardless of the query. If so, whether a value serves LOOKUP or COMPUTE would not affect how far its visual representation influences the decoder. Separating these possibilities requires isolating the role a value plays from the query it appears in. We therefore hold the table, the target cell, and the pixel edit fixed and vary only whether the value is the answer or an operand, using layer- and site-specific interventions on the visual tokens of ground-truth table cells in page images from PubTables-v2 (Smock et al., 2025). Our main contributions are as follows:

- **A paired protocol for seeing versus using a table cell**: We pair LOOKUP and COMPUTE questions that share the table, the target cell, and an edit to that cell, so that only whether the value is seen or used changes, and intervene on the cell's visual tokens layer by layer.
- **Used values stop steering the answer earlier**: Replacing the cell's value steers the answer up to an earlier layer when the value is used than when it is seen, in all eight VLMs under a log-probability criterion and in six under full generation.
- **The cell itself is needed to a model-specific depth**: Removing the evidence cells yields a depth that differs between models but barely across operations or page structures within one; only aggregation over many cells ends earlier, and the operations corresponding to seeing and using end at nearly the same depth. The role difference therefore lies in the value, not in the cell's presence.
- **Used values are consumed, not carried**: The value remains linearly decodable at the cell after its influence ends, but recomputation from it succeeds only at the cell site before that point; later text positions carry the result rather than a reusable operand.

2 RELATED WORKS

Multimodal table understanding. With the rapid advancement of MLLMs, multimodal table understanding, which takes table images directly as input to understand their structure and values, has been actively studied. Table-LLaVA/MMTab (Zheng et al., 2024), TableVQA-Bench (Kim et al., 2024), MMTableBench (Wu et al., 2026a), TableVista (Yang et al., 2026), and WildTableBench (Huang et al., 2026) propose benchmarks for evaluating visual table understanding and reasoning. In addition, methods such as TRivia (Zhang et al., 2026), Table-R1 (Wu et al., 2026b) improve TableQA performance by strengthening cell grounding or intermediate reasoning. For multi-page document understanding, MMLongBench-Doc (Ma et al., 2024) and MP-DocVQA (Tito et al., 2023) propose benchmarks that require leveraging evidence distributed across multiple pages. These works primarily focus on which evidence to locate and how to use it, but do not directly analyze how deep into the model the internal representation of a specific table cell continues to influence the prediction.

Mechanistic interpretability of VLMs. Mechanistic interpretability research analyzes the causal effect of internal representations on model predictions by directly intervening on them. Causal tracing and activation patching have been used to identify how specific layers and representation sites within language models affect predictions (Meng et al., 2022). More recently, these approaches have been extended to VLMs, for example to analyze visual-language interactions in BLIP (Palit et al.,

2023) or to examine the causal role of visual object representations and their relation to hallucination (Li et al., 2026). However, prior work has mainly targeted object-level visual representations, and how localized visual evidence corresponding to small table cells is used in TableQA reasoning remains underexplored.

Layer-wise representation and inference dynamics. As MLLMs become increasingly capable, understanding how their intermediate representations evolve across decoder layers has become an important research question. Some studies use layer-wise probing and representation analysis to identify the layers at which useful information emerges (Skean et al., 2025), while others analyze the inference dynamics of tabular foundation models through layer-wise probing and intervention (Balef et al., 2026). We extend this type of analysis to document-image TableQA, measuring the layer-wise causal influence of the visual representations corresponding to specific table cells. In particular, by comparing LOOKUP and COMPUTE questions that share the same table and the same target cell, we analyze how the intervention depth of identical visual evidence varies with whether its value is seen or used.

3 A FRAMEWORK FOR UNDERSTANDING TABLE-CELL REPRESENTATIONS

In this section, we describe the analysis framework underlying the key findings in Section 4. Rather than proposing a new model or training method, we aim to understand the inner workings of existing VLMs: where the visual evidence of a specific table cell resides inside the decoder, and up to which layer it remains causally involved in the prediction.

3.1 MODELS AND DATA

Models. In the experiments, we analyze eight open-source VLMs: Qwen2.5-VL-7B (Bai et al., 2025a), Qwen3-VL-8B and Qwen3-VL-32B (Bai et al., 2025b), Qwen3.5-9B (Qwen Team, 2026), Ministral-3-8B (Liu et al., 2026), LLaVA-OneVision-2-8B (An et al., 2026), and Gemma-4-12B and Gemma-4-31B (Gemma Team, 2026). Their language decoders contain 28, 36, 64, 32, 34, 36, 48, and 60 layers, respectively. The models span multiple families, decoder depths, and attention architectures. Architectural details are provided in Appendix A.1.

Dataset and tasks. We construct the data from the PubTables-v2 test split (Smock et al., 2025). Our primary setting is multi-page: a single table spans several pages and the two operands lie on different pages. The multi-page paired set contains 821 items from 83 tables in 78 documents. Each item pairs LOOKUP(B), which retrieves the value v_B of a target cell, with COMPUTE(A, B), which produces $f(A, B)$ by addition or subtraction. A single-page paired set of 739 items from 105 tables in 102 documents is built under the same rules and serves as a replication on largely the same documents.

Paired counterfactual construction. A and B are non-header cells with integer values, drawn from different rows of the same column. Each pair uses the same table and target cell B, which serves as the answer in LOOKUP and as an operand in COMPUTE. We construct one donor image by replacing the leading digit of B with B', while preserving the number of digits, font, alignment, cell size, and surrounding layout. We require $B' \neq B$ and $f(A, B') \neq f(A, B)$. Selection criteria and the rendering procedure are given in Appendix A.2.

Evaluation scope. Layer-wise patching uses items for which the model correctly answers LOOKUP(A), LOOKUP(B), and COMPUTE(A, B) on the base image, and LOOKUP(B') and COMPUTE(A, B') on the donor image. Endpoint comparisons additionally require a donor-directed effect in at least one layer of each task. This gate separates intervention effects from pre-existing task errors, and it is what makes the per-model item counts differ across analyses. Answers are graded by exact match after normalization (Appendix A.3).

Question-type items. The question-type analysis of Section 4.3 uses a separate set of 918 items from the same split, which are not paired and carry no donor image. Each item is labeled by its operation, *Select*, *Join*, *Compute*, *Compare*, or *Aggregate*, and by whether its evidence lies on one page or across pages. Questions are generated from the annotated cells of the table, so each item carries its evidence cells with cell-level bounding boxes and a gold answer computed from them. Items labeled multi-page additionally passed a human judgment that one page does not suffice.

Why a LOOKUP–COMPUTE pair. Operation labels describe what a question asks. Our paired design focuses on whether a question sees or uses a particular cell's value, either as the answer or as

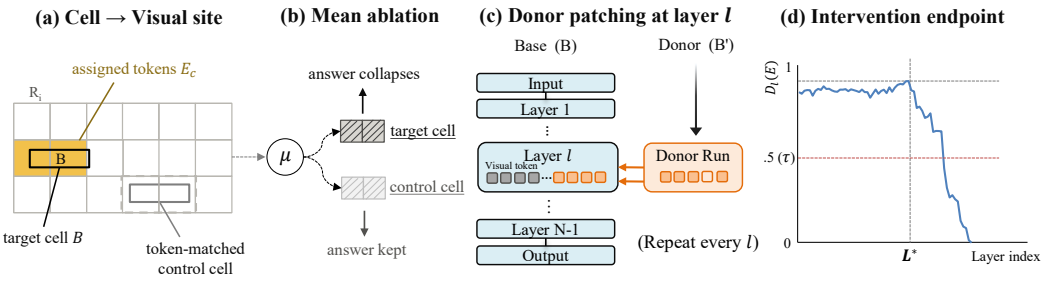

Figure 1: **Intervention settings.** (a) A cell is mapped to the visual tokens its bounding box overlaps, neighboring cells often share a token; (b) Mean ablation replaces those tokens with the image-level mean, with a matched control cell for comparison; (c) Donor patching copies the cell's states from a run on a page with a changed value into the original run at one layer; (d) The endpoint L^* is the last layer at which the intervention effect $D_l(E)$ exceeds τ. Amber marks the cell's visual site.

an operand. We focus on these two roles because they provide explicit counterfactual target answers, B' and $f(A, B')$, for the donor-directed endpoint. For other operations, changing a single cell does not necessarily produce a distinct, valid target answer. In COMPARE, the answer is a row label that may remain unchanged when B changes. In AGGREGATE, the answer depends on multiple cells and may remain unchanged after the edit. In JOIN, changing a key may break the lookup rather than steer it. Section 4.3 therefore examines all five operations using evidence removal. This approach requires no donor and measures sensitivity to removing evidence at the cell site rather than to replacing its value with a specific counterfactual. In that analysis, LOOKUP and COMPUTE correspond to SELECT and COMPUTE, respectively.

3.2 MAPPING TABLE CELLS TO VISUAL SITES

To measure up to which layer the visual evidence of a specific table cell remains involved, and whether this depends on its role, we must first locate that cell's visual tokens rather than the whole image. We expect the value to be initially carried by the visual tokens covering the cell and to become less dependent on them at deeper layers. We test the former in Section 4.1 and the latter in Section 4.2. Let B_c denote the bounding box of cell c and R_i the image region represented by visual token i. We align cell text with page-level word coordinates and scale the resulting box according to each model's image processor. The visual site of cell c is

$$E_c = \{i \mid R_i \cap B_c \neq \emptyset\}. \tag{1}$$

Mappings are generated separately for each model (Fig. 1a).

Cell boundaries frequently cross visual-token boundaries (Fig. A.1). In the single-page set, cells map to a median of two tokens, and 72% of evidence cells share at least one token with a neighboring cell. We therefore intervene on all positions in E_c and use their mean-pooled state for probing. Control cells are matched by token count and share no tokens with the target. An OCR check confirms that all inspected bounding boxes lie on the correct cell. Removing cells whose text lies close to a token boundary preserves the COMPUTE-first ordering in all eight models under both criteria, pattern a model-level sign test supports at $p = 0.008$, although the surviving samples are too small for about half the models to reach significance on their own (Appendix A.4). These checks concern the mapping itself; whether the site carries the evidence the model uses is tested in Section 4.1.

3.3 LAYER-WISE INTERVENTIONS AND ENDPOINTS

At each layer l, we replace the states at the positions in E with their counterparts from a second run (Fig. 1b–c): the donor run for patching and a mean-ablated run for evidence removal. The rest of the computation is left unchanged (Appendix A.5). Writing $D_l(E)$ for the resulting intervention effect and τ for a threshold, the intervention endpoint (Fig. 1d) is

$$L^*(E) = \max\{l \mid D_l(E) > \tau\}. \tag{2}$$

We read D_l under two criteria. Under full generation, the endpoint is the last layer at which patching produces the donor-consistent answer. Under the log-probability criterion, D_l is the normalized donor effect $D_p(l)$. Evidence-removal experiments use the analogous normalized effect $D(l)$ on the

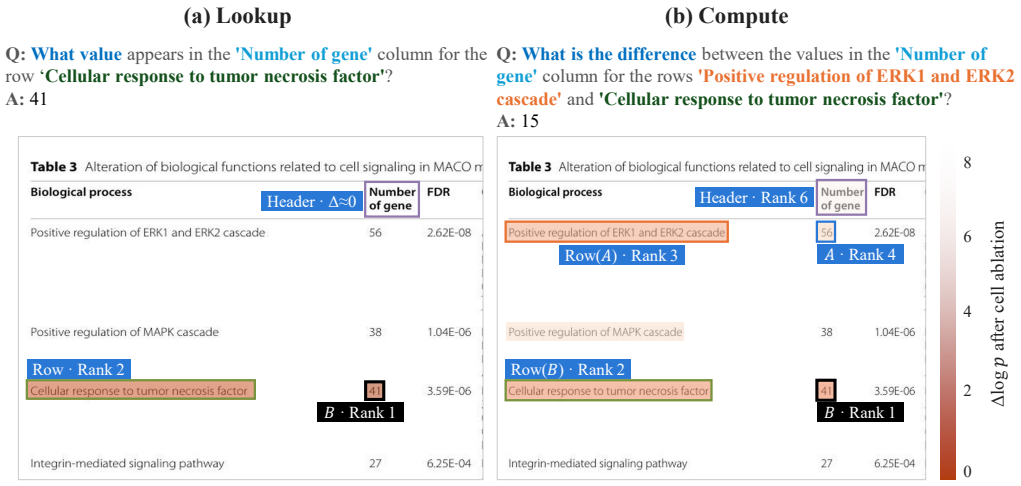

Figure 2: **Cell-wise ablation ranking of visual evidence.** (a) Cell-wise effects for paired LOOKUP; (b) Cell-wise effects for paired COMPUTE questions on the same table and target cell B. Black and blue outlines mark B and the other task-relevant cells. Ranks are computed among eligible cells after excluding cells that share visual tokens with B.

discriminative token. We use $\tau = 0.5$ and evaluate robustness at $\tau \in \{0.3, 0.5, 0.7\}$. Because decoder depth ranges from 28 to 64 layers, we report endpoints as raw layer indices for within-model comparisons and as relative depth, the layer index divided by the number of layers minus one, whenever models are compared with one another. Statistical procedures are described in Appendix A.5.

4 KEY ANALYSES

We examine how the cell's visual evidence is processed through four analyses: whether the cell site carries the evidence the model uses (Section 4.1), how deep its value steers the answer in each role (Section 4.2), whether the depth at which the cell stops being needed depends on the operation or the page structure (Section 4.3), and whether the value remains available and is reused after that depth (Section 4.4). The multi-page paired set is the default; each section states when and why it departs from it.

4.1 ANALYSIS 1: IS THE CELL-ALIGNED VISUAL SITE A VALID INTERVENTION TARGET?

Motivation. Every endpoint in this paper is measured by intervening on a cell's visual site. This measurement describes the cell's value only if the site is where the model reads that value, and three alternatives would break this. The model might recover the number through another path, so ablating the site should break correct answers far more often than ablating a matched control cell. The site might signal that a number is present without encoding which one, so replacing its state with the state for a different value should make the answer follow that value rather than merely become wrong. Or the site might matter only as image content, so B should rank near the top when every cell on the page is ablated in turn.

Experimental setup. All three tests run on all eight models with the same target: the cell B whose value is the answer to LOOKUP(B) and an operand of COMPUTE(A, B). All three are reported on both sets. Throughout, $\mathrm{abl}_c(x)$ denotes replacing the visual tokens of cell c with the mean visual token of the page.

Ablating the site collapses correct answers. Figure 3a shows that ablating the visual tokens of B turns 41–82% of correct LOOKUP answers and 19–63% of correct COMPUTE answers incorrect, while a token-matched control cell changes at most 4%. An exact McNemar test rejects equality in every model and task at $p \leq 2.2 \times 10^{-7}$ and the single-page set replicates the result.

Donor replacement steers the answer. As Figure 3b shows, replacing the layer-0 state of B with its donor counterpart steers 72–100% of LOOKUP answers to B' and 74–100% of COMPUTE answers to $f(A, B')$. The site therefore carries the particular value the answer is built from, not merely a signal that evidence is present.

The target cell ranks far above other cells. In Figure 3c, ablating every cell in turn, B produces the largest drop in 36–76% of LOOKUP items and 18–50% of COMPUTE items against a chance rate of 0.8%, with mean rank percentiles of 0.91–1.00 and 0.80–0.98. Figure 2 shows one item: B

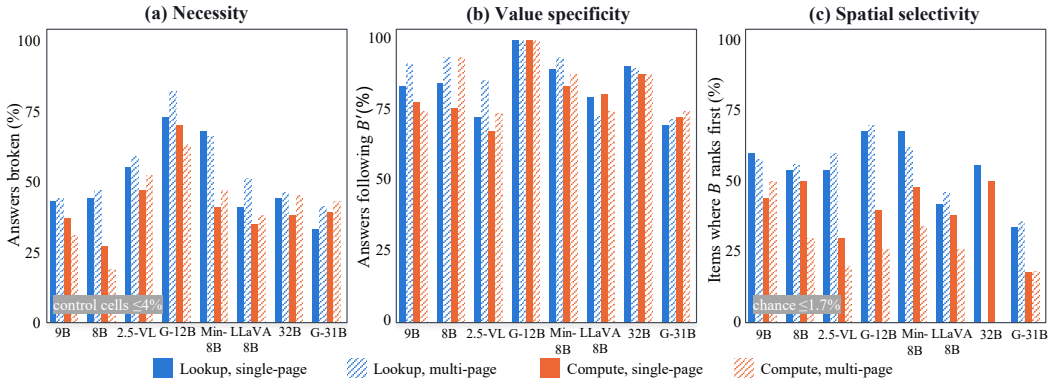

Figure 3: **Cell-aligned visual sites carry the evidence the model uses.** Percentage of items in which (a) ablating B breaks a correct answer, against a token-matched control cell; (b) replacing the layer-0 state of B with its donor state makes the answer follow B'; (c) B receives the largest log-probability drop when every cell is ablated in turn, against the rate expected by chance for Top-1 is 0.8% on average and at most 1.7% per

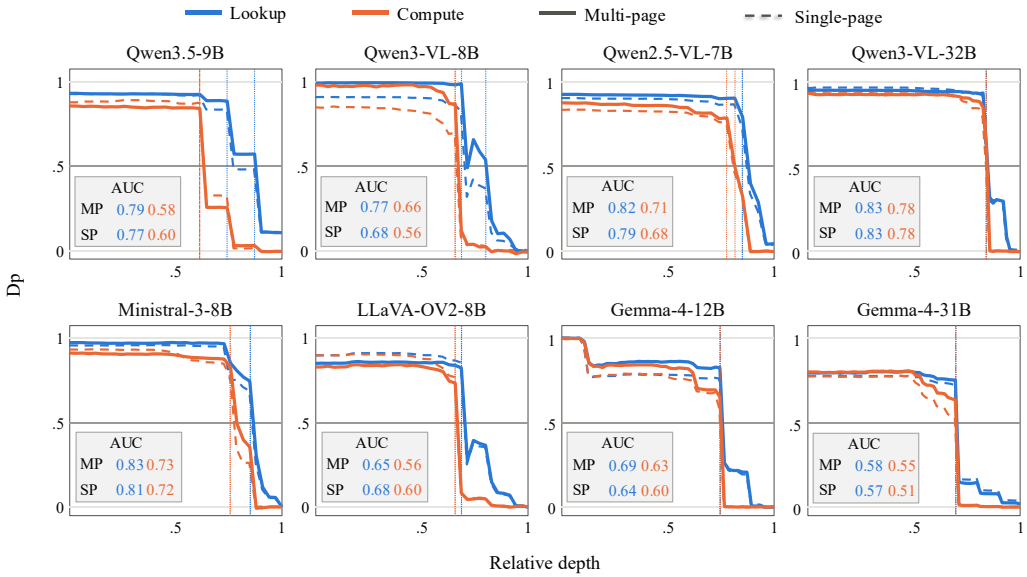

Figure 4: **Donor-effect curves for LOOKUP and COMPUTE.** Mean normalized donor effect $D_p(l)$ against relative depth; endpoint differences are in Table 1.

ranks first in both tasks, but under COMPUTE the drop also spreads to the operand A and the row labels that select the two operands. Where B is not ranked first, the top cell is usually one of these (Appendix A.6).

Takeaway 1. Interventions at answer-relevant cell sites selectively alter model predictions, supporting their use as targets for layer-wise intervention analysis.

4.2 ANALYSIS 2: DOES THE DEPTH OF A VALUE'S INFLUENCE DEPEND ON ITS ROLE?

Motivation. In LOOKUP, the value of B must be carried to the output, whereas in COMPUTE it is needed only until the operation consumes it. If the model reads the operand and then continues computing elsewhere, the value at the cell site should stop steering the COMPUTE answer earlier than the LOOKUP answer.

Experimental setup. We use the multi-page paired set: LOOKUP(B) and COMPUTE(A, B) share the same table and donor change $B \to B'$. At each decoder layer, we replace the target site states of the base run with the donor states and estimate endpoints under full generation and the normalized donor-effect curve $D_p(l)$, and summarize the paired difference $\Delta L^* = L^*_{\text{LOOKUP}} - L^*_{\text{COMPUTE}}$ (Appendix A.5). Thresholds are varied over $\tau \in \{0.3, 0.5, 0.7\}$, and the area under D_p gives a summary that uses no threshold.

Table 1: **Endpoint differences.** ΔL^* with a document-clustered 95% CI; n and p are in Appendix A.7.

Model	Full generation	Log-probability
Qwen3.5-9B	4.86 [3.11, 7.56]	5.49 [4.62, 6.18]
Qwen3-VL-8B	3.06 [1.62, 4.57]	4.00 [2.38, 5.08]
Qwen2.5-VL-7B	1.50 [0.34, 2.75]	1.66 [1.24, 2.01]
Gemma-4-12B	1.05 [−0.52, 2.55]	1.91 [0.96, 2.98]
Ministral-3-8B	1.92 [0.60, 3.17]	1.88 [1.34, 2.49]
LLaVA-OV2-8B	1.89 [1.43, 2.52]	3.19 [2.61, 3.67]
Qwen3-VL-32B	0.93 [−0.40, 2.48]	1.27 [0.50, 1.93]
Gemma-4-31B	1.78 [0.80, 2.87]	1.11 [0.52, 1.84]

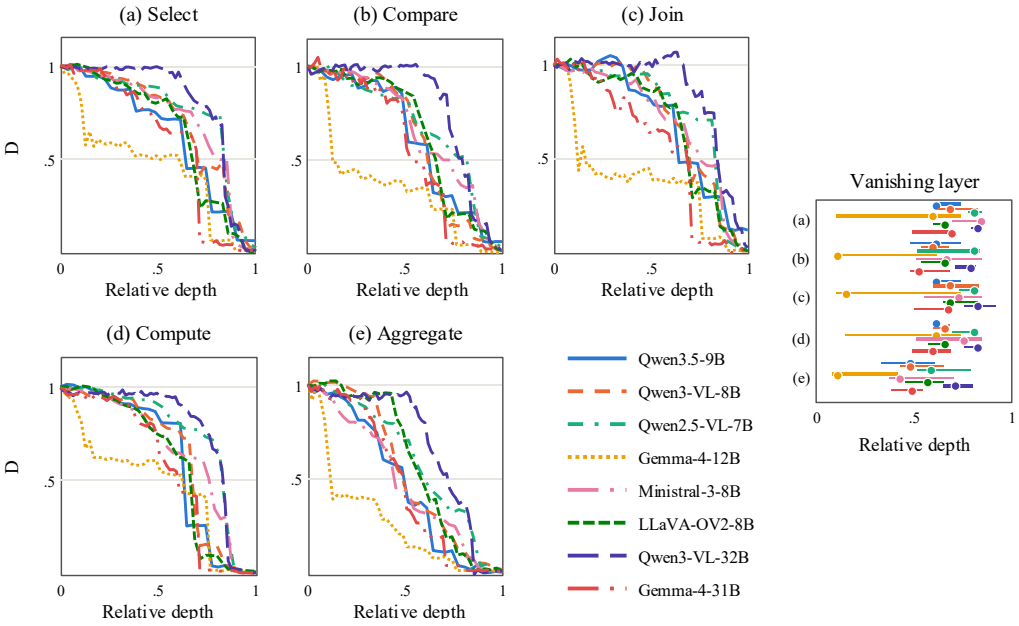

Figure 5: **Evidence-cell endpoints across question types.** (a)–(e) Mean normalized removal effect $D(l)$ against relative depth for SELECT, COMPARE, JOIN, COMPUTE and *Aggregate*; the gray line marks $\tau = 0.5$. Vanishing layer per model and operation: dots mark where the mean curve falls below τ.

COMPUTE ends earlier than LOOKUP. Under the log-probability criterion, COMPUTE ends earlier than LOOKUP in all eight models, by 1.11 to 5.49 layers, with every 95% confidence interval excluding zero (Tab. 1). Under full generation, six of eight models show a significantly earlier COMPUTE endpoint, by 1.50 to 4.86 layers, while Gemma-4-12B and Qwen3-VL-32B show no significant difference. The single-page set replicates the ordering in Appendix A.7.

Without a threshold, all eight models separate the roles. The area under D_p is larger for LOOKUP than for COMPUTE in all eight models, by 0.03 to 0.21 (Fig. 4; Tab. A.11), which a model-level sign test supports at $p = 0.008$. This includes Gemma-4-12B and Qwen3-VL-32B, whose full-generation endpoints are not separated: their areas still differ by 0.06 and 0.05, and their endpoints differ by about one layer, too little for a single threshold to resolve. Varying the threshold does not reverse the ordering either. On the single-page set, the LOOKUP endpoint of the mean curve is never earlier than the COMPUTE endpoint at $\tau = 0.3$, 0.5 or 0.7, and is strictly later at all three in five models (Tab. A.12).

Takeaway 2. The value at the same cell site stops steering the answer earlier when it is consumed as an operand (COMPUTE) than when it is reported as the answer (LOOKUP), and the difference holds under a threshold-free summary of the intervention curve.

4.3 ANALYSIS 3: DOES THE DEPTH AT WHICH A CELL IS NEEDED DEPEND ON THE OPERATION?

Motivation. Section 4.2 varies the role of one value. Document table questions also differ in what they do with their evidence: some select one cell, others join, compare or aggregate cells, often across pages. If more demanding operations kept their evidence in use longer, the depth at which removing a cell stops mattering would move with the operation. If the model instead reads every cell to a depth fixed by the model itself, that depth should stay put across operations and page structures. Evidence removal needs no donor answer, so it applies to every operation.

Experimental setup. We replace the evidence-cell state at each layer with its counterpart from a mean-ablated run and measure the vanishing layer across question types over single- and multi-page items in Figure 5. Because this analysis uses evidence removal rather than donor patching, its endpoints mark where removing the cell stops affecting the answer, not where its particular value stops steering it, and are not directly comparable to those of Section 4.2. Full numerical results are provided in Appendix A.8.

Harder operations do not keep the cell longer. If an operation required longer integration of evidence, its endpoint would move later. It does not: within a model, SELECT, JOIN, COMPUTE and

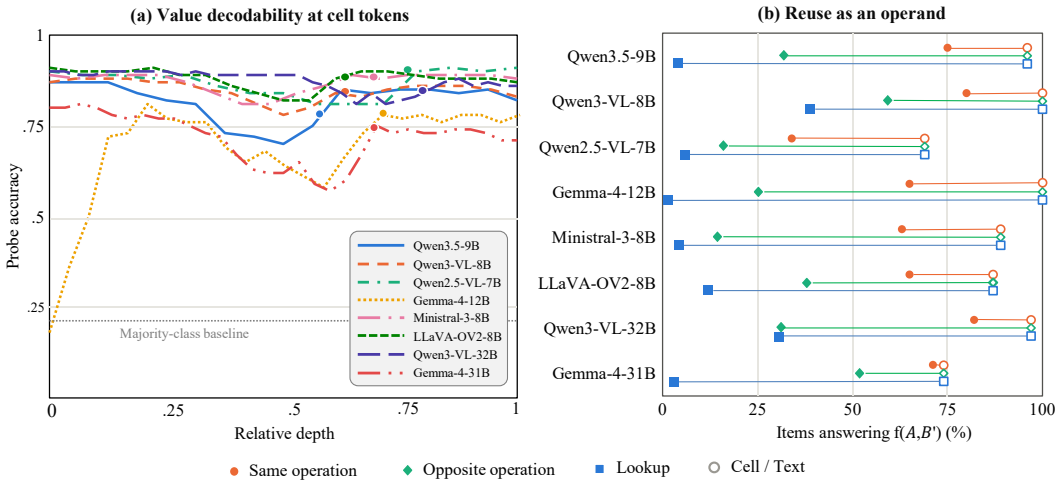

Figure 6: **Value decodability and reuse.** (a) Linear-probe accuracy for the first digit of the target cell's value, read from the hidden states at its visual tokens, against relative depth; markers give each model's COMPUTE endpoint L^* and the dashed line the majority-class baseline. (b) Items whose output becomes $f(A, B')$ at some layer when the $B \to B'$ activation difference is added at the cell site below L^* (open) or at the text positions above it (filled). The difference is taken under ● the same operation with another first operand A_2, ◆ the opposite operation, or ■ LOOKUP; the first carries the change in this operation's answer, the last only B'.

COMPARE end within one layer of L_{rep} in five models and within four in Ministral-3-8B, and in Gemma-4-31B only COMPUTE departs, ending five layers earlier. Moving SELECT between single- and multi-page items shifts its endpoint by at most two layers in seven of eight models. The depth differs between models, from 0.61 to 0.83 of the decoder, but within a model it is largely fixed. Gemma-4-12B, excluded from this is the one model without a stable endpoint. Its curve falls to near τ within the first few layers and stays there, so its vanishing layer moves by up to 30 layers across thresholds (Appendix Tab. A.14). Only AGGREGATE, whose answer depends on many cells, ends earlier in every model at $\tau = 0.5$ and 0.7; at $\tau = 0.3$ the gap shrinks to one layer or less in six of seven models, because its curve declines gradually.

Removal and replacement measure different things. On the question-type items, the operations that correspond to the two roles of Section 4.2, SELECT and COMPUTE, end at the same layer in four models and within one layer in two more (Appendix Tab. A.13), whereas donor patching on the paired items separates the roles in every model. The two measurements use different items, so they do not show that a given cell is needed equally long in both roles. They do show that the depth at which evidence stops being needed is set by the model, whereas the role effect appears in how deep a particular value keeps steering the answer.

Takeaway 3. The depth at which a cell is needed is set largely by the model. Neither the operation nor the page structure moves it, except aggregation over many cells, which ends earlier. The role dependence of Section 4.2, therefore concerns how deep the value steers the answer, and appears only when the intervention targets the value itself rather than the cell's presence.

4.4 ANALYSIS 4: DOES DECODABLE INFORMATION REMAIN FUNCTIONALLY USABLE?

Motivation. The endpoint marks where intervening at the cell site stops changing the answer, not where the value disappears. After it, the value may be gone from the cell, or still present there but no longer read because what the answer needs has already been passed to later positions. A probe at the cell tokens separates these two cases. Whether what is passed on can serve as an operand for recomputation is a further question, which the endpoint alone cannot answer.

Experimental setup. This section uses the single-page paired set, since the recomputation control requires eight runs per item to be correct and leaves too few multi-page items (Tab. A.3. We probe the first digit of v_B from the mean-pooled hidden states at the cell's visual tokens at every layer (Appendix A.9). Separately, we add the $B \to B'$ activation difference to the COMPUTE(A, B) run, at the cell site below the COMPUTE endpoint L^* or at the text positions above it, and count items whose output becomes $f(A, B')$. Adding a difference rather than the donor state keeps the donor's own answer from being copied. The difference is taken between runs with B and B' under the same operation with another first operand A_2, under the opposite operation, or under LOOKUP.

The value remains decodable after the endpoint. Probe accuracy for the first digit of v_B does not fall after L^*: beyond the endpoint it stays between 0.71 and 0.91 in all eight models (Fig. 6a), against a majority-class baseline of 0.20 and a label-shuffled control of 0.19 to 0.22. Qwen3.5-9B and both Gemma models dip in the middle layers but recover by L^*. The value is thus still present at the cell after the answer stops depending on it there: what ends at L^* is that dependence, not the value's presence. As a check, a probe for the first digit of the result stays near its baseline, at 0.22 to 0.24 against 0.18, as expected since the cell tokens precede the question (Appendix A.9).

Only the cell site below the endpoint supports recomputation. Injected at the cell site below L^*, the $B \to B'$ difference yields $f(A, B')$ in 69–100% of items, at the same rate, including the LOOKUP source, which carries B' but no computed answer (Fig. 6b): below the endpoint, the model recomputes from whatever value the cell holds. Above L^* the cell site no longer affects the output, so we inject at the text positions instead. There, success depends on whether the difference already encodes the change in this operation's answer. In all models, it is 34–82% for the same-operation source, 16–53% for the opposite-operation source, and 2–13% for the LOOKUP source, which reaches 32% only in Qwen3-VL-32B. Once the value has left the cell site, the model carries forward the answer rather than an operand it can recompute from.

Takeaway 4. The cell's value is still decodable well past L^*, but recomputation from it is confined to the cell site below L^*. What reaches the text positions afterwards is the answer itself, not a reusable operand.

5 DISCUSSION AND CONCLUSION

Our findings show that the depth over which a counterfactual change at a cell's visual site can steer the answer is associated with the role of the cell's value. This influence generally ends earlier when the value serves as an operand than when it is retrieved as the answer. The ordering is consistent across the tested models under the log-probability criterion and the threshold-free curve summary, while full-generation endpoints show model-dependent exceptions.

Evidence removal reveals a different pattern. Removal endpoints stay largely fixed within each model across operations, ending earlier mainly for aggregation over many cells. For selection, moving from single-page to multi-page items shifts the removal depth by at most two layers in seven of eight models, so evidence on a different page is not read more deeply. The operations corresponding to LOOKUP and COMPUTE end at similar removal depths, although their value-replacement endpoints differ. This suggests that the model sets the depth at which a cell stops being needed, while the role sets how deep its value steers the answer.

Value-related information remains linearly decodable at the cell site beyond the COMPUTE endpoint, even as cell-site edits lose their effectiveness in steering the computation. At later text positions, operation-matched activation differences are more effective than differences derived from LOOKUP. This pattern is consistent with transfer of information about the answer change, but does not establish that the transferred representation is a reusable operand. Decodability, therefore, indicates that information remains recoverable, without establishing its use in generating the answer.

Our paired analysis uses integer-valued cells, with COMPUTE limited to addition and subtraction. COMPUTE also requires reading a second cell and performing arithmetic, so the paired comparison does not isolate the role of B from these additional task demands. Removal and replacement endpoints are measured on different items, and recomputation is evaluated only on single-page items that pass its correctness gate. More generally, the correctness-gated analyses describe successful cases and do not diagnose model errors. Whether the role-dependent depth contributes to the lower accuracy of VLMs on page images therefore remains open; testing it requires extending these interventions to items the model answers incorrectly.

In conclusion, our study separates properties of a cell's evidence that final-answer accuracy does not distinguish. Evidence removal shows whether the cell is still needed, value replacement shows how deep its value steers the answer, and probing shows whether the value remains represented. Future work can develop value-specific interventions that do not require a donor-directed answer. Such interventions would extend the operand analysis beyond COMPUTE to COMPARE and AGGREGATE, where a changed operand need not change the answer in a known direction. It can also test whether interventions at the identified layers and positions improve use of operands in multi-page table questions, and whether the role difference persists when the same tables are given as text.

AI USE STATEMENT

In this work, we did not use generative AI tools for any tasks with required disclosure. Additionally, we used generative AI tools to draft parts of the research paper and to edit the paper to improve readability. We reviewed all AI-assisted work and revised it as necessary. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Xiang An, Yin Xie, Feilong Tang, Yunyao Yan, Huajie Tan, Didi Zhu, Changrui Chen, Xiuwei Zhao, Bin Qin, Kaicheng Yang, et al. LLaVA-OneVision-2: Towards next-generation perceptual intelligence. *arXiv preprint arXiv:2605.25979*, 2026. doi: 10.48550/arXiv.2605.25979.

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

Shuai Bai, An Yang, Keqin Chen, Xuejing Liu, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025b. https://github.com/QwenLM/Qwen3-VL.

Amir Rezaei Balef, Mykhailo Koshil, and Katharina Eggensperger. Is one layer enough? understanding inference dynamics in tabular foundation models. In *International Conference on Machine Learning (ICML)*, 2026. Accepted at ICML 2026.

Yew Ken Chia, Liying Cheng, Hou Pong Chan, Maojia Song, Chaoqun Liu, Mahani Aljunied, Soujanya Poria, and Lidong Bing. M-LongDoc: A benchmark for multimodal super-long document understanding and a retrieval-aware tuning framework. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2025. URL https://aclanthology.org/2025.emnlp-main.469/.

Gemma Team. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026. doi: 10.48550/arXiv.2607.02770.

Jiin Han, Suyoung Bae, Yerim Choi, Sangyun Lee, Kyutae Kim, RyunHo Kim, Jee-Hyong Lee, Changkyu Choi, and YunSeok Choi. Pubtables-qa: A benchmark toward cross-page table reasoning in table visual question answering, 2026. First International Workshop on Responsible Multimodal Foundation Models for Knowledge Discovery, co-located with ACM SIGKDD 2026.

Junzhe Huang, Xiaoxiao Sun, Yan Yang, Yuxuan Hou, Ruotian Zhang, Sirui Li, Hehe Fan, Serena Yeung-Levy, and Xin Yu. WildTableBench: Benchmarking multimodal foundation models on table understanding in the wild. In *Proceedings of the Third Conference on Language Modeling (COLM)*, 2026.

Kjetil Indrehus, Adrian Duric, Changkyu Choi, and Ali Ramezani-Kebrya. Towards self-explainable document visual question answering with chain-of-explanation predictions. *arXiv preprint arXiv:2605.06058*, 2026.

Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. OCR-free document understanding transformer. In *European Conference on Computer Vision*, pp. 498–517. Springer, 2022.

Yoonsik Kim, Moonbin Yim, and Ka Yeon Song. TableVQA-Bench: A visual question answering benchmark on multiple table domains, 2024.

Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexiang Hu, Fangyu Liu, Julian Martin Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2Struct: Screenshot parsing as pretraining for visual language understanding. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 18893–18912. PMLR, 23–29 Jul 2023.

Qiming Li, Zekai Ye, Xiaocheng Feng, Weihong Zhong, Weitao Ma, and Xiachong Feng. Causal tracing of object representations in large vision language models: Mechanistic interpretability and hallucination mitigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 31645–31653, 2026. doi: 10.1609/aaai.v40i37.40431.

Alexander H. Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, et al. Ministral 3. *arXiv preprint arXiv:2601.08584*, 2026. doi: 10.48550/arXiv.2601.08584.

Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, Pan Zhang, Liangming Pan, Yu-Gang Jiang, Jiaqi Wang, Yixin Cao, and Aixin Sun. MMLONGBENCH-DOC: Benchmarking long-context document understanding with visualizations. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-3041. Datasets and Benchmarks Track.

Siddarth Mamidanna, Daking Rai, Ziyu Yao, and Yilun Zhou. All for one: LLMs solve mental math at the last token with information transferred from other tokens. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 30747–30760, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.1565.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. DocVQA: A dataset for VQA on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2200–2209, 2021.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, pp. 17359–17372, 2022. doi: 10.52202/068431-1262.

Vedant Palit, Rohan Pandey, Aryaman Arora, and Paul Pu Liang. Towards vision-language mechanistic interpretability: A causal tracing tool for BLIP. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pp. 2856–2861, 2023.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Oscar Skean, Md Rifat Arefin, Dan Zhao, Niket Nikul Patel, Jalal Naghiyev, Yann LeCun, and Ravid Shwartz-Ziv. Layer by layer: Uncovering hidden representations in language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 55854–55875. PMLR, 2025.

Ray Smith. An overview of the Tesseract OCR engine. In *Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR)*, volume 2, pp. 629–633, 2007. doi: 10.1109/ICDAR.2007.4376991.

Brandon Smock, Valerie Faucon-Morin, Max Sokolov, Libin Liang, Tayyibah Khanam, Amrit Ramesh, and Maury Courtland. Pubtables-v2: A new large-scale dataset for full-page and multi-page table extraction. *arXiv preprint arXiv:2512.10888*, 2025.

Mohamed Ali Souibgui, Changkyu Choi, Andrey Barsky, Kangsoo Jung, Ernest Valveny, and Dimosthenis Karatzas. DocVXQA: Context-aware visual explanations for document question answering. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 56549–56569. PMLR, 2025.

Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7035–7052, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.435.

Rubèn Tito, Dimosthenis Karatzas, and Ernest Valveny. Hierarchical multimodal transformers for multi-page docvqa, 2023. URL https://arxiv.org/abs/2212.05935.

Yahong Wang, Juncheng Wu, Zhangkai Ni, Longzhen Yang, Yihang Liu, Chengmei Yang, Ying Wen, Lianghua He, Xianfeng Tang, Hui Liu, and Yuyin Zhou. When token pruning is worse than random: Understanding visual token information in VLLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31910–31919, 2026.

Xianjie Wu, Xiaohang Xu, Tingyu Jiang, Jian Yang, Di Liang, Xianfu Cheng, Zhenhe Wu, Linzheng Chai, Wei Zhang, Jiaheng Liu, Ge Zhang, Bob Simons, Tongliang Li, and Zhoujun Li. MMTableBench: A multi-level multimodal benchmark for reasoning and layout complexity in table qa. In *Proceedings of the ACM Web Conference 2026*, pp. 3881–3892. Association for Computing Machinery, 2026a. doi: 10.1145/3774904.3792367.

Zhenhe Wu, Jian Yang, Zhongjiang He, Changzai Pan, Jiaheng Liu, Xianjie Wu, Yu Zhao, Shuangyong Song, Yongxiang Li, Zhoujun Li, and Xuelong Li. Table-R1: Region-based reinforcement learning for table understanding. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 27374–27391. Association for Computational Linguistics, 2026b. doi: 10.18653/v1/2026.findings-acl.1364.

Zheyuan Yang, Liqiang Shang, Junjie Chen, Xun Yang, Chenglong Xu, Bo Yuan, Chenyuan Jiao, Yaoru Sun, and Yilun Zhao. TableVista: Benchmarking multimodal table reasoning under visual and structural complexity. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 34967–34985. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.1745.

Junyuan Zhang, Bin Wang, Qintong Zhang, Fan Wu, Zichen Wen, Jialin Lu, Junjie Shan, Ziqi Zhao, Shuya Yang, Ziling Wang, Ziyang Miao, Huaping Zhong, Yuhang Zang, Xiaoyi Dong, Ka-Ho Chow, and Conghui He. TRivia: Self-supervised fine-tuning of vision-language models for table recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 33196–33206, 2026.

Mingyu Zheng, Xinwei Feng, Qingyi Si, Qiaoqiao She, Zheng Lin, Wenbin Jiang, and Weiping Wang. Multimodal table understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9102–9124. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.493.

A Appendix

A.1 Model Architectures

Table A.1 summarizes the decoder and vision encoder configurations of the eight models we analyze. Layers are numbered from 0, and an intervention at layer l is applied to the input hidden state of decoder layer l. Relative depth is defined as the layer index divided by the number of decoder layers minus one.

Table A.1: **Model architectures.** Decoder depth, hidden size, attention heads, attention type, and vision encoder for each model. Heads are given as query / key-value heads. *Global* denotes full-attention layers, with their layer indices in parentheses. Gemma-4-12B is the only model with a unified, encoder-free architecture that ingests raw image patches instead of using a separate vision encoder (Gemma Team, 2026).

Model	Layers	Hidden	Heads	Decoder attention	Vision encoder
Qwen2.5-VL-7B	28	3,584	28 / 4	full attention in all layers	ViT, 32 layers
Qwen3-VL-8B	36	4,096	32 / 8	full attention in all layers	ViT, 27 layers
Qwen3.5-9B	32	4,096	16 / 4	24 linear-attention + 8 global (3, 7, . . . , 31)	ViT, 27 layers
Ministral-3-8B	34	4,096	32 / 8	full attention in all layers	Pixtral, 24 layers
LLaVA-OneVision-2-8B	36	4,096	32 / 8	full attention in all layers	OneVision encoder, 24 layers
Gemma-4-12B	48	3,840	16 / 8	40 sliding-window (1,024) + 8 global (5, 11, . . . , 47)	unified, encoder-free
Qwen3-VL-32B	64	5,120	64 / 8	full attention in all layers	ViT, 27 layers
Gemma-4-31B	60	5,376	32 / 16	50 sliding-window (1,024) + 10 global (5, 11, . . . , 59)	Gemma 4 vision encoder, 27 layers

A.2 Data Construction

Evaluation sets. Table A.2 lists the documents the sets share. The two paired sets share 73 documents. The question-type set contains 887 unique questions, 31 questions appear twice because two of its source pools overlap.

Table A.2: **Evaluation sets.** Shared columns give the number of documents, counted by PMC identifier, that a set has in common with each other set.

			Shared documents			
Set	Items	Docs	MP	SP	QT	Used in
Multi-page paired (MP)	821	78	78	73	44	Sec. 4.1, Sec. 4.2, Sec. 4.4
Single-page paired (SP)	739	102	73	102	57	Sec. 4.1, Sec. 4.2, Sec. 4.4
Question-type (QT)	918	275	44	57	275	Sec. 4.3

Table A.3: **Items passing the correctness gate.** Pass = items correct in all five runs, with the pass rate in parentheses; Pairs = passing items with a defined endpoint in both tasks. Recomputation = single-page items passing the six-run and eight-run gates.

	Multi-page		Single-page		Recomputation	
Model	Pass (%)	Pairs	Pass (%)	Pairs	6-run	8-run
Qwen3.5-9B	53 (6)	43	145 (20)	114	47	24
Qwen3-VL-8B	18 (2)	17	78 (11)	58	13	5
Qwen2.5-VL-7B	95 (12)	68	154 (21)	103	51	29
Gemma-4-12B	158 (19)	158	252 (34)	252	196	180
Ministral-3-8B	68 (8)	60	96 (13)	80	29	19
LLaVA-OV2-8B	52 (6)	37	115 (16)	89	36	23
Qwen3-VL-32B	112 (14)	95	168 (23)	148	76	38
Gemma-4-31B	341 (42)	254	365 (49)	265	301	272

Correctness gate. Of the items passing the five-run gate, those with a defined endpoint in both tasks enter Table 1. The recomputation control of Section 4.4 adds the same-operation source runs,

a six-run gate, and the opposite-operation source runs, an eight-run gate on which all three sources are scored. The multi-page set passes the five-run gate at 2 to 42% of items against 11 to 49% for the single-page set (Tab. A.3), which is why the recomputation control is run on the single-page set. Items flagged as `result_in_table` are kept in the gate.

Question Types. Examples are grouped by their operation and page structure. Questions and answers are reproduced exactly as presented to the models.

Question examples

Select, single-page

Which BRCA1 mutation is associated with 3 patients?
A: c.68_69delAG (p.GluValfs)

How many subjects does Brosan et al. deal with in table1?
A: 646

Aggregate, single-page

How many BRCA1 variants in Table 1 have patient counts greater than 1?
A: 5

What is the average frequency across the four SF-36-related entries?
A: 23.5

Select, multi-page

In the table captioned "Descriptive statistics for CSAMS-G.", what is the SD reported for x18?
A: 1.075

Across the multi-page Table 5, which row has the highest value in the P value column?
A: Type of institution

What is the Women % for the least common masturbation frequency category?
A: 0.5

Which compound name appears in both Table 3 and Table 4 with a base m/z of 191.15?
A: 2,4-Di-tert-butylphenol

Join, multi-page

Using Table 1, find the entry whose Location is 11q12.3; according to Table 2, what is its Resolution (Å)?
A: 2.00

In Table 2, for the metabolite with the largest Fold Change, what is the corresponding S.E. listed in Table 4?
A: 657.516

Based on Table 3, what is the RLS-positive count for the age group with the highest frequency in Table 1?
A: 58

Compute, multi-page

What is the average of the two highest Appreciation loadings in Table 3?
A: 0.638

In Table 1, what is the sum of the MW values for the rows AhDREB10 and AhDREB76?
A: 58182.10

Across this multi-page table, what is the sum of the Anemia ($n = 1434$) values for the rows "Not Metropolitan" and "No charge"?
A: 0

What is the absolute difference between the FC value for Pcdhga5 in Table 1 and the FC value for M0R4L3 in Table 2?
A: 1.01

Compare, multi-page

Which row shows the dominant pattern under OPJ5 in this multi-page table?
A: Aldehydes

What is the smallest P value when combining Tables 2 and 3?
A: 1.53E-14

Among BRCA2 variants, which has the most patients and how many?
A: c.9371A>T (3)

Aggregate, multi-page

Sum the number of studies using DPOAE across Tables 2 and 3.
A: 4

Across the multi-page Table 1, what is the combined sample size reported under the Methodology column for the studies by Wolf L., 2022 and Chalmers J., 2023?
A: 2145

Candidate selection. Candidate cells must also have a text match rate of at least 0.95 between the cell text and the word strings distributed with PubTables-v2. In the multi-page set, the table spans two pages in 660 items, three in 120 and four in 41; B has two digits in 390 items, three in 285 and four in 146; and the operation is a difference in 548 items and a sum in 273. In the single-page set,

B has two digits in 639 items, three in 69 and four in 31, and the operation is a difference in 482 items and a sum in 257. Candidates whose result $f(A, B)$ is itself printed in the table are flagged as `result_in_table` and kept.

Donor image. B' has the same number of digits as B and differs only in its leading digit, subject to $B' \neq B$ and $f(A, B') \neq f(A, B)$. The donor image is produced by covering the bounding box of the B cell in the original page with the surrounding background color and redrawing B' in a serif font matched to the original glyph height and color. Base and donor images are saved as lossless PNGs from the same JPEG decode, so the two images differ only inside the B cell. All pairs are saved as cropped comparison images to allow visual inspection.

A.3 Prompting, Decoding, and Answer Matching

For all models, the input consists of the page image(s) and the question, with the following instruction appended after the question:

```
Reply with the answer only. No explanation, no reasoning, no
bullet points.
```

All experiments use greedy decoding (`do_sample=False`). The maximum generation length is set per experiment above the longest expected answer (8–32 tokens) and truncates no answer in the paired analyses. Answers are graded by exact match after Unicode NFKC normalization, lowercasing, whitespace stripping and collapsing, removal of thousands separators, replacement of the Unicode minus (−) by a hyphen, and removal of a trailing period. The Lookup and Compute question formats are as follows.

> Lookup: In the table 5, what value appears in the ‘ISCO 2008 No’ column for the row ‘Protective service workers NEC’?
> Compute: In the table 5, what is the sum of the values in the ‘ISCO 2008 No’ column for the rows ‘Primary care physicians’ and ‘Protective service workers NEC’?

A.4 Cell-to-Token Mapping by Model

Cell bounding boxes are mapped into each model’s resized input image with the processor’s own resize rule (Fig. A.1, Table A.4), for multi-page inputs the mapping uses the page that carries the cell.

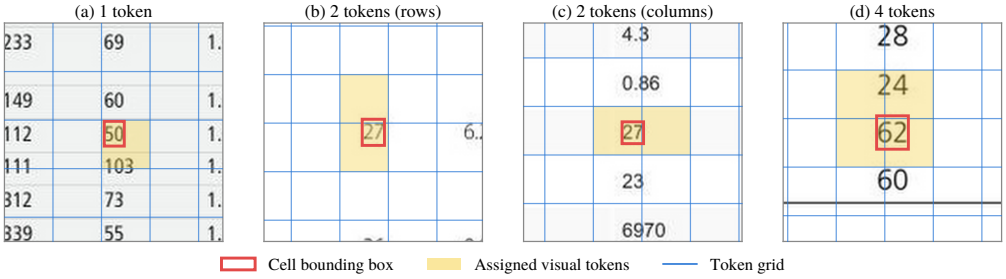

Figure A.1: **Mapping table cells to model-specific visual sites.** Red boxes denote cell boundaries, yellow regions the assigned visual tokens, and blue lines the visual-token grid.

Bounding-box check. Every visual site is derived from a cell’s bounding box, so we checked that each box lies on the intended cell. From each of the question-type set, the multi-page paired set and the single-page paired set, we sampled 300 evidence cells, cropped their bounding boxes, and read each crop as a single text line with Tesseract 5.3.4 OCR (Smith, 2007) and its English model. OCR matched the annotated value in 95.3% of cells, respectively. We inspected every mismatch in its page context and found the box on the correct cell in all cases; the mismatches came from OCR failures on rotated tables, two-line cells, small type, and symbols such as “-” and “M”.

Table A.4: **Visual-token geometry by model.** Pixel coverage of each merged visual token after model-specific image resizing.

Model	Token size (px)	Note
Qwen3.5-9B, Qwen3-VL-8B, Qwen3-VL-32B	32×32	patch 16, 2×2 spatial merge
Qwen2.5-VL-7B	28×28	patch 14, 2×2 spatial merge
Ministral-3-8B	28×28	Pixtral processor; an `[IMG_BREAK]` token after each token row
LLaVA-OneVision-2 8B	28×28	Qwen2-VL processor
Gemma-4-12B, Gemma-4-31B	48×48	resized to the soft-token budget (1,120)

Boundary proximity. When the text of B lies close to a token edge, patching its tokens may also patch part of a neighboring cell. We compute the margin between the B-cell text box and the nearest border of its visual-token rectangle, divided by the token side length at each model's input resolution, and repeat the paired comparison on multi-page items with margin > 0.1 and > 0.2 (Table A.5). Filtering costs sample size rather than effect size: as few as nine pairs remain in Qwen3-VL-8B. Counting a model as significant if either criterion reaches $p < 0.05$, seven of eight models are significant at margin > 0.1 and four at margin > 0.2.

Table A.5: **Robustness to digits near visual-token boundaries.** Multi-page set, pairs filtered by margin without re-running. FG = full-generation endpoint; LP = log-probability endpoint at $\tau = 0.5$. The two criteria keep different items valid, so pair counts differ. Order = COMPUTE first : tied : reversed; ΔL = mean [document-clustered bootstrap 95% CI]; p = document-level sign-flip permutation test.

		Margin > 0.1				Margin > 0.2			
Model		Pairs	Order	ΔL [95% CI]	p	Pairs	Order	ΔL [95% CI]	p
Qwen3.5-9B	FG	35	30 : 5 : 0	4.49 [3.51, 5.75]	0.0014	25	23 : 2 : 0	4.84 [3.63, 6.33]	0.0022
	LP	40	36 : 4 : 0	6.75 [4.80, 8.85]	0.0005	26	22 : 4 : 0	4.92 [3.50, 5.76]	0.004
Qwen3-VL-8B	FG	12	12 : 0 : 0	2.75 [1.89, 3.47]	0.03	9	9 : 0 : 0	2.89 [1.88, 3.89]	0.063
	LP	12	12 : 0 : 0	4.42 [3.44, 4.92]	0.03	9	9 : 0 : 0	4.44 [3.14, 5.00]	0.063
Qwen2.5-VL-7B	FG	58	39 : 10 : 9	1.57 [0.10, 3.11]	0.057	36	24 : 7 : 5	1.72 [0.51, 3.41]	0.01
	LP	73	54 : 18 : 1	2.52 [1.59, 3.58]	<5e-5	44	33 : 10 : 1	2.98 [1.62, 4.45]	5e-5
Gemma-4-12B	FG	111	35 : 66 : 10	1.09 [−0.85, 2.82]	0.28	64	17 : 43 : 4	1.19 [−0.02, 2.73]	0.089
	LP	111	42 : 67 : 2	1.81 [0.67, 2.96]	0.0059	64	23 : 40 : 1	1.33 [−0.28, 3.06]	0.14
Ministral-3-8B	FG	40	20 : 8 : 12	1.60 [−0.11, 3.13]	0.13	29	15 : 6 : 8	1.66 [−0.24, 3.34]	0.16
	LP	47	34 : 11 : 2	4.91 [2.25, 8.36]	0.0002	32	21 : 9 : 2	3.16 [1.16, 6.42]	0.0014
LLaVA-OV2-8B	FG	31	29 : 2 : 0	2.00 [1.46, 2.74]	0.00055	18	17 : 1 : 0	1.72 [1.18, 2.50]	0.0043
	LP	44	40 : 1 : 3	2.50 [−1.92, 5.65]	0.33	25	24 : 1 : 0	6.00 [3.21, 8.52]	0.0021
Qwen3-VL-32B	FG	73	17 : 26 : 30	0.40 [−0.79, 2.04]	0.63	47	12 : 17 : 18	1.28 [−0.09, 3.55]	0.16
	LP	85	35 : 25 : 25	−0.28 [−6.82, 4.50]	0.97	51	21 : 16 : 14	2.98 [0.23, 5.54]	0.1
Gemma-4-31B	FG	174	38 : 124 : 12	1.39 [0.41, 2.49]	0.0076	105	20 : 82 : 3	0.93 [−0.09, 2.06]	0.12
	LP	222	39 : 168 : 15	0.43 [−1.27, 2.24]	0.64	111	18 : 90 : 3	1.01 [−0.15, 2.49]	0.13

Token sharing. A target cell that shares a visual token with another cell is patched together with part of that neighbor. If the endpoint difference came from the neighbor, it should weaken among cells that own their tokens. It does not: on the multi-page set, ΔL is positive for exclusive cells in all eight models, which a model-level sign test rejects at $p = 0.008$, and reaches $p < 0.05$ in four (Table A.6).

A.5 INTERVENTION DEFINITIONS

Mean ablation. At the vision-encoder output, we replace the targeted merged visual tokens immediately before the language model with the mean vector over all visual tokens from the same image. For multi-page inputs, we compute the mean separately for each page. We apply the same procedure to target and control cells. We select the control from non-header cells that lie in a different row and column from the target, contain a value different from both the target value and the gold answer, and share no visual tokens with the target. Among eligible cells, we choose the one with the closest token count to the target.

Table A.6: **Endpoint difference by token sharing.** Multi-page set, full generation. Exclusive = the B cell shares no visual token with another cell; shared = it shares at least one. ΔL = mean [document-clustered bootstrap 95% CI]; p = document-level sign-flip permutation test. All pairs are in Table 1.

	Exclusive			Shared		
Model	Pairs	ΔL [95% CI]	p	Pairs	ΔL [95% CI]	p
Qwen3.5-9B	29	5.17 [3.26, 8.61]	0.0022	14	4.21 [2.11, 6.87]	0.032
Qwen3-VL-8B	11	2.82 [1.89, 3.64]	0.015	6	3.50 [1.00, 6.57]	0.12
Qwen2.5-VL-7B	39	1.33 [0.28, 2.42]	0.05	29	1.72 [−0.23, 3.75]	0.17
Gemma-4-12B	78	1.38 [−0.87, 3.22]	0.23	80	0.72 [−1.17, 2.81]	0.46
Ministral-3-8B	38	2.74 [1.29, 4.21]	0.011	22	0.50 [−0.93, 2.00]	0.59
LLaVA-OV2-8B	30	2.00 [1.48, 2.66]	0.001	7	1.43 [0.33, 2.57]	0.25
Qwen3-VL-32B	60	0.23 [−1.33, 2.32]	0.93	35	2.11 [0.29, 4.33]	0.051
Gemma-4-31B	128	1.62 [0.04, 3.52]	0.06	126	1.94 [0.78, 3.06]	0.0054

Evidence removal. Each item involves three runs: a clean run on the original image, a corrupt run in which the evidence-cell tokens are mean-ablated, and a patched run. At layer l, the evidence-cell hidden states of the corrupt run overwrite the clean run at the same positions, at that one layer only, and the gold answer is teacher-forced to measure the log-probability of its discriminative token,

$$D(l) = \frac{\log p_{\text{clean}} - \log p_{\text{patched@}l}}{\log p_{\text{clean}} - \log p_{\text{corrupt}}}. \tag{A.1}$$

$D(l) \approx 1$ means that corrupting only this cell's state at layer l damages the answer as much as corrupting the whole input, and $D(l) \approx 0$ means it leaves the answer unaffected. The discriminative token is the position in the gold answer whose log-probability drops most when the evidence is removed; it avoids answers such as "0." or "1" whose first token does not depend on the evidence. Only items with $\log p_{\text{clean}} - \log p_{\text{corrupt}} > 0.5$ nat are usable.

Donor patching. Under the full-generation criterion, the endpoint is the last layer at which the greedily generated answer equals the donor answer, B' for LOOKUP and $f(A, B')$ for COMPUTE. Under the log-probability criterion, the log-probability of the donor answer is normalized between the base run and the full-donor run,

$$D_p(l) = \frac{\log p^{\text{donor}}_{\text{patched@}l} - \log p^{\text{donor}}_{\text{base}}}{\log p^{\text{donor}}_{\text{donor}} - \log p^{\text{donor}}_{\text{base}}}, \tag{A.2}$$

Only items whose denominator exceeds 0.5 nat are usable. The full-generation criterion is defined only on items that pass the correctness gate, whereas the log-probability criterion requires only the usability condition.

Statistics. Target and control collapse rates are compared with an exact McNemar test on the same items. The paired endpoint difference ΔL is reported as a mean with a document-clustered bootstrap 95% confidence interval over 10,000 resamples, and tested with a document-level sign-flip permutation test over 20,000 permutations, whose smallest attainable p is reported as <5e-5. Consistency across models is tested with a sign test over the eight models, for which eight of eight in one direction gives $p = 0.008$. All resampling uses seed 0. Models are compared by testing the difference on the same item pairs, never by setting one model's significant result beside another's non-significant one. No correction for multiple comparisons is applied across models or groups, so per-model p values should be read as descriptive.

A.6 VALIDITY OF THE CELL SITE

Necessity. For items the model answers correctly, the visual tokens of the target cell B or of the control cell are mean-ablated before the first decoder layer, and we count answers that become incorrect. Target and control outcomes are compared with an exact McNemar test on the same items (Table A.7).

Value specificity. For items passing the five-run gate, the layer-0 state of the B cell is replaced by its state from the donor image, and we count answers that take the donor value, B' for LOOKUP and $f(A, B')$ for COMPUTE (Table A.8). Patching at the last decoder layer leaves the original answer unchanged in every item, model and set, so the effect is not an artifact of the patching itself.

Spatial selectivity. For 50 multi-page items per model, drawn from 29 documents, every cell of every page the question uses is mean-ablated in turn, 225 to 270 cells per item on average, and B is ranked among them by the drop in the log-probability of the answer (Table A.9). Where B is not first, the top cell is usually the operand A or the row label that selects B's row.

Table A.7: **Necessity of the target cell.** Collapse (%) of correctly answered items when the target cell B or the control cell is mean-ablated, shown as B / control. McNemar $p \leq 2.2 \times 10^{-7}$ in every cell.

	Multi-page				Single-page			
	LOOKUP		COMPUTE		LOOKUP		COMPUTE	
Model	n	B / ctrl	n	B / ctrl	n	B / ctrl	n	B / ctrl
Qwen3.5-9B	721	44 / 0	213	31 / 3	613	43 / 0	419	37 / 1
Qwen3-VL-8B	705	47 / 0	158	19 / 3	632	44 / 1	339	27 / 1
Qwen2.5-VL-7B	671	59 / 1	227	52 / 1	578	55 / 1	375	47 / 1
Gemma-4-12B	678	82 / 1	376	63 / 4	602	73 / 0	472	70 / 3
Ministral-3-8B	625	66 / 1	250	47 / 1	503	68 / 0	360	41 / 2
LLaVA-OV2-8B	643	51 / 1	205	38 / 2	548	41 / 1	348	35 / 1
Qwen3-VL-32B	706	46 / 0	303	45 / 1	643	44 / 0	440	38 / 1
Gemma-4-31B	699	41 / 0	536	43 / 1	628	33 / 0	604	39 / 0

Table A.8: **Value specificity of the target cell.** Answers (%) that take the donor value after the layer-0 state of B is replaced by its donor state; n = items passing the five-run gate.

	Multi-page			Single-page		
Model	n	LOOKUP	COMPUTE	n	LOOKUP	COMPUTE
Qwen3.5-9B	53	92	75	142	84	78
Qwen3-VL-8B	18	94	94	78	85	76
Qwen2.5-VL-7B	95	86	74	154	73	68
Gemma-4-12B	158	100	100	252	100	100
Ministral-3-8B	68	94	88	96	90	84
LLaVA-OV2-8B	52	73	75	115	80	81
Qwen3-VL-32B	112	90	88	168	91	88
Gemma-4-31B	341	72	75	365	70	73

Table A.9: **Spatial selectivity of the target cell.** Multi-page set, 50 items per model. Top-1 = items (%) in which B produces the largest drop; Pct. = mean rank percentile of B. Brackets give document-clustered bootstrap 95% confidence intervals; chance for Top-1 is 0.8%.

	LOOKUP		COMPUTE	
Model	Top-1 (%)	Pct.	Top-1 (%)	Pct.
Qwen3.5-9B	58 [44, 72]	0.91 [0.83, 0.97]	50 [33, 67]	0.83 [0.72, 0.93]
Qwen3-VL-8B	56 [43, 68]	0.94 [0.88, 1.00]	30 [20, 40]	0.88 [0.81, 0.94]
Qwen2.5-VL-7B	60 [42, 76]	0.98 [0.94, 1.00]	20 [9, 32]	0.81 [0.66, 0.94]
Gemma-4-12B	70 [55, 85]	1.00 [1.00, 1.00]	26 [14, 41]	0.98 [0.96, 0.99]
Ministral-3-8B	62 [45, 78]	0.91 [0.83, 0.98]	34 [18, 51]	0.86 [0.77, 0.93]
LLaVA-OV2-8B	46 [29, 64]	0.93 [0.86, 0.99]	26 [12, 42]	0.80 [0.67, 0.91]
Qwen3-VL-32B	76 [62, 88]	0.97 [0.93, 1.00]	44 [31, 59]	0.92 [0.85, 0.98]
Gemma-4-31B	36 [22, 52]	0.98 [0.93, 1.00]	18 [8, 30]	0.94 [0.88, 0.98]

Table A.10: **Endpoint differences with item counts and p values.** $\Delta L = L^*_{\text{LOOKUP}} - L^*_{\text{COMPUTE}}$, mean [document-clustered bootstrap 95% CI]; p = document-level sign-flip permutation test. (a) Multi-page set, both criteria scored on the same items, as in Table 1. (b) Single-page set; the log-probability criterion needs no correctness gate and so has more items.

	Full generation			Log-probability		
Model	n	ΔL [95% CI]	p	n	ΔL [95% CI]	p
(a) Multi-page						
Qwen3.5-9B	43	4.86 [3.11, 7.56]	0.0007	43	5.49 [4.62, 6.18]	<5e-5
Qwen3-VL-8B	17	3.06 [1.62, 4.57]	0.015	17	4.00 [2.38, 5.08]	0.015
Qwen2.5-VL-7B	68	1.50 [0.34, 2.75]	0.030	68	1.66 [1.24, 2.01]	<5e-5
Gemma-4-12B	158	1.05 [−0.52, 2.55]	0.20	158	1.91 [0.96, 2.98]	0.00035
Ministral-3-8B	60	1.92 [0.60, 3.17]	0.020	60	1.88 [1.34, 2.49]	<5e-5
LLaVA-OV2-8B	37	1.89 [1.43, 2.52]	0.0006	37	3.19 [2.61, 3.67]	0.0006
Qwen3-VL-32B	95	0.93 [−0.40, 2.48]	0.27	95	1.27 [0.50, 1.93]	0.0069
Gemma-4-31B	254	1.78 [0.80, 2.87]	0.0011	254	1.11 [0.52, 1.84]	0.0002
(b) Single-page						
Qwen3.5-9B	114	2.74 [1.96, 3.49]	<5e-5	142	5.20 [4.24, 6.21]	<5e-5
Qwen3-VL-8B	58	2.69 [1.60, 3.61]	0.0003	78	4.19 [2.39, 6.39]	<5e-5
Qwen2.5-VL-7B	103	1.50 [1.06, 1.95]	<5e-5	154	2.81 [1.67, 4.17]	<5e-5
Gemma-4-12B	252	0.51 [−1.05, 1.82]	0.54	252	1.16 [0.42, 1.91]	0.0014
Ministral-3-8B	80	2.90 [2.04, 3.85]	<5e-5	96	2.51 [1.66, 3.53]	<5e-5
LLaVA-OV2-8B	89	1.67 [0.85, 2.51]	0.0014	115	2.04 [0.78, 3.12]	0.0037
Qwen3-VL-32B	148	−0.28 [−1.54, 0.73]	0.66	168	2.29 [1.47, 3.26]	<5e-5
Gemma-4-31B	265	2.78 [2.00, 3.56]	<5e-5	365	1.89 [0.82, 2.89]	0.00085

Table A.11: **Area under D_p.** Relative depth on $[0, 1]$. Multi-page uses the items of Table 1; single-page is the replication set. LOOKUP exceeds COMPUTE in all eight models in both sets, model-level sign test $p = 0.008$ each.

	Multi-page			Single-page		
Model	LOOKUP	COMPUTE	Diff.	LOOKUP	COMPUTE	Diff.
Qwen3.5-9B	0.79	0.58	0.21	0.77	0.60	0.17
Qwen3-VL-8B	0.77	0.66	0.11	0.68	0.56	0.12
Qwen2.5-VL-7B	0.82	0.71	0.11	0.79	0.68	0.11
Gemma-4-12B	0.69	0.63	0.06	0.64	0.60	0.04
Ministral-3-8B	0.83	0.73	0.10	0.81	0.72	0.09
LLaVA-OV2-8B	0.65	0.56	0.09	0.68	0.60	0.08
Qwen3-VL-32B	0.83	0.78	0.05	0.83	0.78	0.05
Gemma-4-31B	0.58	0.55	0.03	0.57	0.51	0.06

A.7 ENDPOINTS UNDER TWO CRITERIA

Table A.10 gives the item counts and p values behind Table 1, together with the single-page replication. The two models without a full-generation difference, Gemma-4-12B and Qwen3-VL-32B, are the same on both sets.

Table A.11 gives the area under D_p; on the single-page set LOOKUP exceeds COMPUTE by 0.04 to 0.17. The area is computed on the same items as the endpoint difference, so the mean of the item-level differences equals the difference of the areas under the mean curves.

Table A.12 gives the endpoints of the mean curve at the three thresholds.

Table A.12: **Threshold robustness of the paired endpoints.** Single-page set, vanishing layer of the mean D_p curve at $\tau = 0.3 / 0.5 / 0.7$.

Model	LOOKUP	COMPUTE
Qwen3.5-9B	27 / 23 / 23	23 / 19 / 19
Qwen3-VL-8B	28 / 24 / 24	23 / 23 / 21
Qwen2.5-VL-7B	24 / 23 / 23	23 / 21 / 21
Gemma-4-12B	35 / 35 / 35	35 / 35 / 29
Ministral-3-8B	28 / 28 / 27	26 / 25 / 25
LLaVA-OV2-8B	28 / 24 / 24	23 / 23 / 23
Qwen3-VL-32B	54 / 53 / 52	53 / 53 / 52
Gemma-4-31B	41 / 41 / 41	41 / 41 / 31

Table A.13: **Vanishing layer by operation and page structure.** Evidence removal at $\tau = 0.5$. L_{rep} is the vanishing layer of each model's curve over all question-type items. Bold marks endpoints three or more layers from L_{rep}.

			Single-page		Multi-page				
Model	Layers	L_{rep}	Select	Aggr.	Select	Join	Compute	Compare	Aggr.
Items (n)			61	22	229	110	302	52	75
Qwen3.5-9B	32	19	19	**11**	19	19	19	19	**15**
Qwen3-VL-8B	36	23	23	**13**	24	24	23	24	**17**
Qwen2.5-VL-7B	28	22	22	**15**	22	22	22	22	**16**
Gemma-4-12B	48	15	9	7	7	7	27	23	5
Ministral-3-8B	34	24	23	**13**	24	**28**	25	22	**16**
LLaVA-OV2-8B	36	23	23	21	23	23	23	23	**20**
Qwen3-VL-32B	64	52	52	**42**	52	52	52	52	**42**
Gemma-4-31B	60	40	**29**	**28**	41	41	**35**	39	**29**

A.8 QUESTION-TYPE ENDPOINTS AND THRESHOLD ROBUSTNESS

Table A.13 gives the vanishing layer at $\tau = 0.5$ underlying Figure 5, and Table A.14 repeats it at $\tau = 0.3$ and 0.7. Besides Gemma-4-12B, single-page *Select* in Gemma-4-31B also moves with the threshold, from layer 40 to 21.

A.9 PROBING AND RECOMPUTATION

Probing. For each item we store the layer-wise hidden states of the LOOKUP and COMPUTE prompts at the B-cell tokens, mean-pooled over the tokens. The classifier is multinomial logistic regression on standardized features with L2 regularization and $C = 0.05$. Evaluation uses document-level 5-fold cross-validation, so that all items of a document and both the base and donor variants fall in the same fold, and every prediction comes from a probe trained on other documents. The targets are the first digit of B and the first digit of the result R. The label-shuffled control is the same

Table A.14: **Threshold robustness of the question-type endpoints.** Vanishing layer at $\tau = 0.3 / 0.5 / 0.7$. Gemma-4-12B, in parentheses, moves by up to 30 layers across thresholds and is not interpreted.

Group	n	Qwen3.5-9B	Qwen3-VL-8B	Qwen2.5-VL-7B	Gemma-4-12B	Ministral-3-8B	LLaVA-OV2-8B	Qwen3-VL-32B	Gemma-4-31B
Select, SP	61	23/19/19	24/23/22	22/22/19	(34/9/4)	28/23/17	24/23/22	52/52/45	40/29/21
Aggregate, SP	22	14/11/7	15/13/12	21/15/7	(23/7/5)	24/13/7	23/21/16	51/42/37	29/28/22
Select, MP	229	23/19/19	24/24/23	22/22/22	(35/7/5)	28/24/18	24/23/22	52/52/52	41/41/35
Join, MP	110	23/19/19	29/24/23	23/22/22	(35/7/5)	28/28/24	24/23/22	53/52/52	41/41/41
Compute, MP	302	19/19/19	24/23/23	22/22/21	(35/27/5)	26/25/17	23/23/21	53/52/50	41/35/29
Compare, MP	52	27/19/15	30/24/22	23/22/22	(29/23/7)	28/22/17	29/23/20	52/52/48	41/39/30
Aggregate, MP	75	19/15/11	23/17/15	22/16/14	(20/5/4)	25/16/13	24/20/17	52/42/38	35/29/22

Table A.15: **Value decodability and reuse.** Single-page set. Probe B = first-digit accuracy of B at its cell tokens after L^*; probe R = best-layer accuracy for the first digit of the result at the same tokens, against a baseline of 0.18. Reuse = items (%) whose answer becomes $f(A, B')$, scored on the eight-run items; at the cell site the three sources give identical rates.

		Probe			Cell, below L^*	Text, above L^*		
Model	L^*	B	R	n	all sources	same	opposite	LOOKUP
Qwen3.5-9B	19	0.82–0.85	0.22	24	96	75	33	4
Qwen3-VL-8B	23	0.83–0.86	0.22	5	100	80	60	40
Qwen2.5-VL-7B	22	0.90–0.91	0.24	29	69	34	17	7
Gemma-4-12B	35	0.76–0.78	0.24	180	100	65	26	2
Ministral-3-8B	24	0.88–0.89	0.24	19	89	63	16	5
LLaVA-OV2-8B	23	0.87–0.90	0.23	23	87	65	39	13
Qwen3-VL-32B	52	0.86–0.88	0.23	38	97	82	32	32
Gemma-4-31B	41	0.71–0.75	0.22	272	74	72	53	4

pipeline with labels shuffled within the training folds; on the multi-page set it reaches 0.21 to 0.23. Per-model values are in Table A.15.

Recomputation control. The receiver is the COMPUTE(A, B) run on the base image. Injecting the donor state wholesale would allow the donor's own answer to be copied, so we instead add to the receiver's layer-l input an activation difference that carries only the $B \to B'$ change. Each difference is taken between two runs that differ only in B, and the three sources differ only in which run it comes from: (1) the same operation with a different operand A_2, COMPUTE$(A_2, B')-$COMPUTE(A_2, B); (2) the opposite operation, COMPUTE$_g(A_2, B')-$COMPUTE$_g(A_2, B)$; and (3) LOOKUP$(B')-$LOOKUP(B). Copying any source's answer cannot produce the receiver's recomputed result $f(A, B')$. The difference is injected at the B-cell tokens below L^* or at all text positions above it, one layer at a time, and answers are judged by greedy full generation. Here L^* is the endpoint of the mean D_p curve for COMPUTE at $\tau = 0.5$ on the single-page set (Table A.12). All three sources are scored on the items passing the eight-run gate (Table A.3); results are in Table A.15.