How Do Document Parsers Break?
Auditing Structural Vulnerability in Document Intelligence

Anonymous ACL submission

Abstract

Document Layout Analysis (DLA) pipelines provide structured page representations for retrieval-augmented generation, long-document question answering, and other document intelligence systems, yet their robustness evaluation remains largely area-centric. We identify this *Footprint Bias* and propose **ProSA**, a lightweight output-level auditing framework that decouples controlled probing, policy-driven targeting, and structure-aware diagnosis. ProSA combines Block-level Structural Loss Rate (B-SLR), granularity-aware exposure descriptors, and pathway attribution to analyze where structural identity is lost, at what exposure granularity failures emerge, and how failures propagate. Across MinerU and PP-StructureV3 on 1,000 pages, affected area weakly tracks perturbation-induced OCR instability (R^2=0.384/0.110), whereas B-SLR aligns much more closely with it (R^2=0.727/0.916). Exposure descriptors further separate occlusion- and topology-dominant pathways, while matched-footprint structural probes cause much larger downstream QA/retrieval drops than area-matched erasure. These results shift DLA robustness evaluation from footprint-based measurement toward structure-aware vulnerability auditing.

1 Introduction

Document Layout Analysis (DLA) converts visually organized pages into structured representations of text, tables, figures, and other layout elements, supporting document intelligence systems such as retrieval-augmented generation over visually rich documents (Ueda et al., 2026), financial document reasoning (Zhao et al., 2024a), and medical record understanding (Cottet et al., 2026). Errors at this stage—including boundary corruption, spurious block merges, and misplaced outputs—can propagate as corrupted context, misaligned evidence, and unreliable downstream predictions.

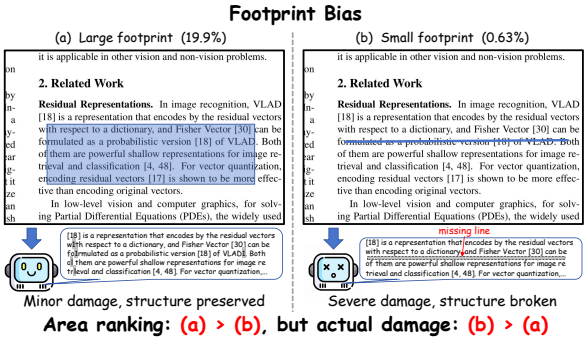

Figure 1: **Footprint Bias**: a large perturbation may preserve structure, while a small structural probe can drop a text line and cause greater parsing damage.

Despite progress on clean benchmarks and realistic evaluation settings, DLA robustness evaluation remains largely area-centric. Existing protocols often parameterize perturbation severity by global corruption magnitude or affected pixel footprint, and report aggregate degradation through metrics such as Character Error Rate (CER) or detection accuracy (Hendrycks and Dietterich, 2019; Michaelis et al., 2020; Chen et al., 2024; Du et al., 2025; Ouyang et al., 2025). Such evaluations tell us *whether* performance declines, but reveal little about *how* or *why* parsing fails structurally, especially when lightweight auditing is needed on unlabeled document collections.

A central limitation is what we term the **Footprint Bias**: the tendency to infer perturbation severity from pixel footprint alone. As Figure 1 shows, a broad erased region may corrupt local text while leaving the main paragraph structure largely intact. By contrast, a thin perturbation with a much smaller footprint can intersect a sensitive line or layout boundary, causing a missing-line failure in which the parser skips a line and connects non-adjacent text fragments as if they were continuous. The smaller perturbation can therefore be more damaging: it removes document evidence and disrupts the local reading flow despite occupying far fewer pixels.

Such mismatches can propagate downstream: a dropped line may remove answer-bearing evidence or corrupt retrieval/QA context, while a larger visible perturbation may leave block identity and reading order intact. They also expose why area-centric evaluation can mis-rank structural vulnerability. The key challenge is therefore not simply to measure degradation, but to explain structural vulnerability through three questions: (Q_1) where structural identity is lost, (Q_2) at what granularity perturbation exposure becomes predictive of failure, and (Q_3) which pathway—direct physical occlusion or topology-level disruption—drives the degradation.

To address this diagnostic problem, we propose **ProSA** (**Pro**be-guided **S**tructure-aware **A**uditing), a lightweight output-level framework for auditing structural vulnerability in DLA pipelines. ProSA instantiates the audit as a tripartite decomposition $\mathcal{F} = (\mathcal{S}, \Pi, \mathcal{D})$, where $\mathcal{S}$ defines a shared space of controlled visual probes, Π selects probes under different targeting policies, and $\mathcal{D}$ diagnoses the resulting clean–perturbed parsing divergence. By decoupling the probe space, selection logic, and diagnostic criteria, ProSA compares different targeting policies under a common action space without modifying the audited parser.

The diagnostic component is centered on B-SLR, an output-level signal that measures structural loss by comparing clean and perturbed parser outputs. Unlike footprint-based severity or terminal-only degradation scores, B-SLR checks whether each clean block preserves a valid geometry–text counterpart, capturing both missing geometry and lost textual identity under a unified structural criterion. Exposure descriptors and pathway attribution then explain at which exposure granularity these losses emerge and whether they arise from direct occlusion or topology-dominant disruption. Because the core diagnostics use the clean parse as structural reference, ProSA supports lightweight auditing on unlabeled documents while still incorporating benchmark annotations when available. Experiments show that these structural diagnostics better track OCR instability and downstream QA/retrieval drops than footprint-based severity.

In summary, our contributions are threefold:

(1) We identify the **Footprint Bias** in DLA robustness evaluation, showing that pixel footprint is an unreliable proxy for structural damage and may mis-rank parser vulnerability.

(2) We propose **ProSA**, a lightweight output-level auditing framework that couples controlled probes, policy-guided selection, and structure-aware diagnosis.

(3) We instantiate ProSA with B-SLR, exposure descriptors, and pathway attribution, enabling annotation-light analysis that better tracks OCR and QA/retrieval drops.

2 Related Work

Document parsing robustness and downstream document intelligence. Robustness evaluation has moved from generic corruption suites and detection benchmarks (Hendrycks and Dietterich, 2019; Michaelis et al., 2020) to document-specific parsing settings such as RoDLA, DocPTBench, and OmniDocBench (Chen et al., 2024; Du et al., 2025; Ouyang et al., 2025). Recent document intelligence studies further show that OCR and layout quality can affect retrieval-augmented generation, financial reasoning, and medical document understanding (Ueda et al., 2026; Zhao et al., 2024a; Cottet et al., 2026; Zhang et al., 2025). However, these evaluations mainly report corruption-level, benchmark-level, or downstream degradation. They provide limited diagnosis of which parser-level structural changes, such as block loss, text-geometry mismatch, or topology disruption, actually drive the observed failures. Our work complements this line by auditing structural failure patterns directly in DLA outputs and relating them to downstream QA/retrieval behavior.

Perturbation footprint, placement, and black-box diagnosis. Visual perturbations are often parameterized by magnitude, affected area, or patch budget. Cutout and Random Erasing use erased area as a key augmentation strength (DeVries and Taylor, 2017; Zhong et al., 2020), while document augmentation tools expose visual and spatial perturbation parameters for document images (Groleau et al., 2023). Patch-based robustness studies further show that location and composition matter: adversarial patches can remain effective under small spatial budgets (Brown et al., 2017), and recent analyses reveal placement hot-spots or jointly optimize patch shape, location, number, and content (Kimhi et al., 2025; Yang et al., 2026). Black-box explanation methods similarly perturb inputs to identify influential features or sensitive regions (Ribeiro et al., 2016; Lundberg and Lee, 2017; Petsiuk et al., 2018). Unlike these works, we audit structured DLA outputs rather than scalar predictions: each

parsed element couples geometry, category, and text. This requires element matching, geometry-text preservation analysis, and occlusion/topology pathway diagnosis, while still requiring no access to parser internals.

3 Problem Setting

Let $\mathcal{M}$ denote a document layout analysis (DLA) system that maps a document image I to a structured parsing output

$$E = \mathcal{M}(I) = \{e_i\}_{i=1}^{N}, \tag{1}$$

where each parsed element is represented as $e_i = (b_i, c_i, s_i)$, with bounding box b_i, canonical category label $c_i \in \mathcal{C}_5 = \{\text{text}, \text{title}, \text{table}, \text{figure}, \text{equation}\}$, and recognized text s_i.

Given a probe configuration P, we obtain a counterfactual page and its parser output:

$$I' = \mathrm{Perturb}(I; P), \qquad E_{\mathrm{adv}} = \mathcal{M}(I'). \tag{2}$$

We formulate auditing as comparing the clean and perturbed parsing outputs:

$$(\Delta, \mathcal{R}) = \mathrm{Audit}(E, E_{\mathrm{adv}}; P, \mathcal{Y}), \tag{3}$$

where $\mathcal{Y}$ denotes optional benchmark references when available. The terminal component Δ records externally visible degradation, including perturbation-induced OCR CER and detection-side $\Delta\mathrm{mAP}$, while the structural component $\mathcal{R}$ diagnoses where structural identity is lost, at what exposure granularity failure becomes predictive, and through which pathway it occurs. In our experiments, CER is computed against the clean parser output, and $\Delta\mathrm{mAP}$ is the per-image mAP@0.5 drop from clean to perturbed outputs against layout annotations.

This formulation extends conventional robustness evaluation beyond reporting Δ: by introducing $\mathcal{R}$, the audit shifts from aggregate degradation measurement to structural perturbation diagnosis, without parser retraining or access to parser internals.

4 Probe-guided Structure-aware Auditing

As illustrated in Figure 2, we propose **ProSA** (**Pro**be-guided **S**tructure-aware **A**uditing), a lightweight output-level framework for auditing structural vulnerability in DLA pipelines. ProSA organizes the audit into two coupled components: a *Policy-guided Prober*, which constructs controlled perturbations, and a *Structure-aware Auditor*, which diagnoses the resulting clean–perturbed output divergence. Formally, we instantiate ProSA as

$$\mathcal{F} = (\mathcal{S}, \Pi, \mathcal{D}), \tag{4}$$

where $\mathcal{S}$ is a shared probe space, Π is a policy space for selecting probes, and $\mathcal{D}$ is a diagnostic space for auditing the resulting parsing changes. This decomposition separates *what* perturbations can be applied, *how* they are selected, and *how* their effects are measured, enabling DLA parsers to be audited without modifying the parser itself.

4.1 Component I: Policy-guided Prober

The Policy-guided Prober defines the controlled perturbation side of ProSA. It contains two coupled parts: a shared probe space $\mathcal{S}$ that defines admissible visual interventions, and a policy space Π that selects probes under different targeting assumptions. Because all policies draw from the same $\mathcal{S}$, differences in observed damage can be attributed to selection logic rather than search-space changes.

Shared probe space. Each probe configuration $P \in \mathcal{S}$ is represented as

$$P = \langle \mathcal{H}, \mathcal{V}, \mathcal{B}, \mathcal{T} \rangle, \tag{5}$$

where $\mathcal{H}$ denotes geometry, $\mathcal{V}$ visual appearance, $\mathcal{B}$ composition behavior, and $\mathcal{T}$ placement strategy. This factorization keeps the perturbation space modular: geometry controls the spatial support, appearance controls the visible pattern, behavior controls how the probe is composed with the page, and placement controls where it interacts with layout structure. Figure 5 in Appendix A.1 visualizes representative perturbation examples, while detailed probe definitions and parameter ranges are provided there.

Policy space. Given the shared probe space, a policy $\pi \in \Pi$ maps available document context to a selected probe:

$$\pi(\text{context}) = \hat{P} \in \mathcal{S}. \tag{6}$$

Different policies use different levels of context, ranging from random and rule-based selection to LLM- and VLM-guided targeting. All policies emit the same probe schema, so their effects can be compared under a common action space. After a policy selects $\hat{P}$, the perturbation operator applies it to the clean page to obtain $I' = \mathrm{Perturb}(I; \hat{P})$, and the same parser is rerun to produce $E_{\mathrm{adv}} = \mathcal{M}(I')$. The resulting clean–perturbed pair is then passed

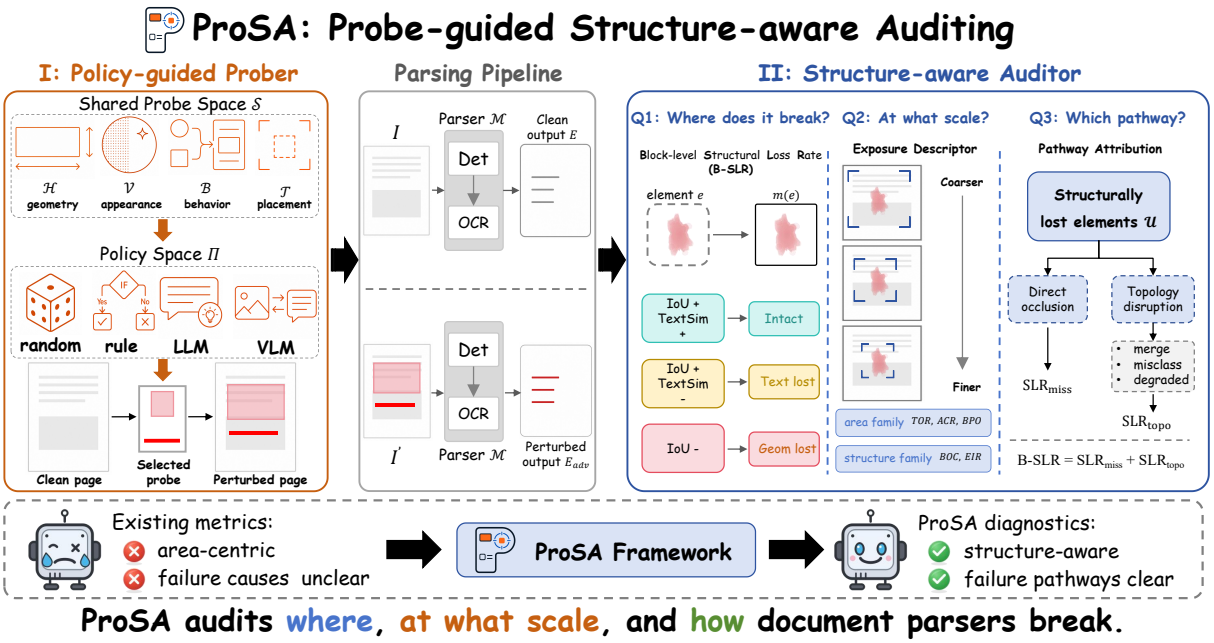

Figure 2: Overview of the proposed tripartite vulnerability auditing framework, linking controlled perturbation generation, policy-space probe selection, and diagnostic auditing.

to the Structure-aware Auditor. Implementation details are deferred to Appendix A.3.

4.2 Component II: Structure-aware Auditor

Diagnostic space. The Structure-aware Auditor instantiates the diagnostic space $\mathcal{D}$ by separating *judging* from *explaining*. The terminal component Δ records externally visible degradation signals, while the structural component $\mathcal{R}$ diagnoses how degradation arises inside the parsing process:

$$\begin{aligned} \Delta &= (\Delta_{\text{ocr}}, \Delta_{\text{det}}), \\ \mathcal{R} &= (\mathcal{R}_{\text{fail}}, \mathcal{R}_{\text{exposure}}, \mathcal{R}_{\text{path}}). \end{aligned} \tag{7}$$

We instantiate Δ_{ocr} with OCR CER and Δ_{det} with ΔmAP, the per-image mAP@0.5 drop against layout annotations between the clean and perturbed outputs, following the metric definitions in Appendix B.4. CER serves as the primary terminal signal, while ΔmAP provides a supplementary detection-side check. The three structural components correspond to the questions introduced in Section 1: $\mathcal{R}_{\text{fail}}$ measures whether structural identity is lost, $\mathcal{R}_{\text{exposure}}$ describes at what granularity perturbation exposure becomes predictive of failure, and $\mathcal{R}_{\text{path}}$ attributes failures to distinct propagation pathways.

Structural failure measurement. We instantiate $\mathcal{R}_{\text{fail}}$ by checking whether each clean parsed element preserves a valid geometry-text correspondence after perturbation. For a clean element $e = (b, c, s) \in E$, we assign its candidate perturbed counterpart by best-IoU lookup:

$$m(e) = \arg\max_{e' \in E_{\text{adv}}} \text{IoU}(e, e'), \tag{8}$$

where $\text{IoU}(e, e')$ denotes box overlap.

Geometric overlap alone is insufficient, because a perturbed prediction may still overlap the correct region while losing textual identity. We therefore define text consistency as

$$\text{TextSim}(e, e') = \frac{|\text{LCS}(s(e), s(e'))|}{\max(|s(e)|, |s(e')|)}, \tag{9}$$

where LCS denotes character-level longest common subsequence. We write $e \sim e'$ when both geometry and text are preserved:

$$\begin{aligned} e \sim e' := &\big(\text{IoU}(e, e') \geq \tau_{\text{iou}}\big) \\ &\wedge \big(\text{TextSim}(e, e') \geq \tau_{\text{text}}\big), \end{aligned} \tag{10}$$

with fixed diagnostic gates $\tau_{\text{iou}} = 0.1$ and $\tau_{\text{text}} = 0.5$, where IoU admits displaced counterparts and TextSim checks textual identity; see Appendix B.5.

Using this criterion, **B**lock-level **S**tructural **L**oss **R**ate (**B-SLR**) is defined as

$$\mathcal{R}_{\text{fail}} := \text{B-SLR} = \frac{|\{e \in E : \neg(e \sim m(e))\}|}{|E|}. \tag{11}$$

Thus, B-SLR counts an element as structurally lost not only when its box disappears, but also when a nearby prediction remains while its textual identity is no longer preserved. Because both E and E_{adv} are parser outputs, B-SLR does not require manual element annotations and can be computed

on unlabeled documents using the clean parse as the structural reference. Category corruption and topology-specific distortions are analyzed within $\mathcal{R}_{\text{path}}$.

Granularity-aware exposure descriptors. We instantiate $\mathcal{R}_{\text{exposure}}$ with five exposure descriptors that summarize how a probe interacts with page structure before terminal quality is evaluated. These descriptors span two practical settings: output-derived descriptors for reference-free auditing, and annotation-dependent descriptors for benchmark-side diagnosis when layout annotations are available. Let Ω denote the page domain, $A(P) \subseteq \Omega$ the perturbation support induced by probe P, Ω_L the union of annotated layout regions, $\partial\Omega_L$ the corresponding boundary support, L the set of annotated layout elements, and E the clean parser output.

For any spatial support $\Gamma \subseteq \Omega$ and any element collection $\mathcal{X}$, we define

$$\begin{aligned} \text{cover}(P, \Gamma) &= \frac{|A(P) \cap \Gamma|}{|\Gamma|}, \\ \text{hit}(P, \mathcal{X}) &= \frac{1}{|\mathcal{X}|} \sum_{e \in \mathcal{X}} \mathbf{1}\big[b(e) \cap A(P) \neq \emptyset\big]. \end{aligned} \tag{12}$$

These two operators induce the two diagnostic families:

$$\begin{aligned} \mathcal{G}_{\text{area}}(P) &= \big(\text{cover}(P, \Omega), \text{cover}(P, \Omega_L), \\ &\qquad \text{cover}(P, \partial\Omega_L)\big), \\ \mathcal{G}_{\text{struct}}(P) &= \big(\text{hit}(P, L), \text{hit}(P, E)\big). \end{aligned} \tag{13}$$

Together, these two families instantiate $\mathcal{R}_{\text{exposure}}$. Concretely, $\mathcal{G}_{\text{area}}$ corresponds to total occlusion ratio (TOR), area coverage ratio (ACR), and boundary pixel overlap (BPO), while $\mathcal{G}_{\text{struct}}$ corresponds to block overlap count ratio (BOC) and element interference ratio (EIR). TOR and EIR are output-derived, whereas ACR, BPO, and BOC are annotation-dependent diagnostics used when layout annotations are available. Exact constructions of Ω_L and $\partial\Omega_L$ are deferred to Appendix B.1. This separation allows the core exposure analysis to remain output-level through TOR and EIR, while ACR, BPO, and BOC provide additional benchmark-side probes when annotations are available.

Failure pathway attribution. We instantiate $\mathcal{R}_{\text{path}}$ by decomposing structural failure into distinct propagation pathways. The key distinction is whether an element fails because it is directly occluded by the perturbation, or because the perturbation disrupts surrounding layout structure without heavily covering the element itself.

Let

$$\mathcal{U} = \big\{e \in E : \neg\big(e \sim m(e)\big)\big\} \tag{14}$$

be the set of structurally failed elements. To quantify direct physical occlusion, we define

$$\rho(e, P) = \frac{|b(e) \cap A(P)|}{|b(e)|}. \tag{15}$$

Given an occlusion threshold $\eta_{\text{occ}} = 0.3$, the failure set is partitioned as

$$\begin{cases} \mathcal{U}_{\text{miss}} = \{e \in \mathcal{U} : \rho(e, P) \geq \eta_{\text{occ}}\}, \\ \mathcal{U}_{\text{topo}} = \{e \in \mathcal{U} : \rho(e, P) < \eta_{\text{occ}}\}. \end{cases} \tag{16}$$

The corresponding normalized pathway losses are

$$\text{SLR}_{\text{miss}} = \frac{|\mathcal{U}_{\text{miss}}|}{|E|}, \quad \text{SLR}_{\text{topo}} = \frac{|\mathcal{U}_{\text{topo}}|}{|E|}. \tag{17}$$

By construction,

$$\text{B-SLR} = \text{SLR}_{\text{miss}} + \text{SLR}_{\text{topo}}.$$

We instantiate $\mathcal{R}_{\text{path}} = (\text{SLR}_{\text{miss}}, \text{SLR}_{\text{topo}})$, with the topology-dominant subset further resolved into merge, misclass, and degraded errors.

Within $\mathcal{U}_{\text{topo}}$, we further assign each failure to one of three layout-aware types: *merge*, where multiple clean elements collapse into the same perturbed prediction; *misclass*, where spatial correspondence is largely preserved but the canonical category changes; and *degraded*, which covers the remaining non-occlusion failures. We apply the priority order merge > misclass > degraded, yielding a disjoint and exhaustive partition of $\mathcal{U}_{\text{topo}}$. Full definitions are given in Appendix B.6.

4.3 Validation principle

The structural diagnostics in $\mathcal{R}$ are useful only if they remain grounded in terminal degradation signals. We validate this grounding from two perspectives. First, *faithfulness* measures whether a structural variable tracks perturbation-induced OCR instability:

$$\text{Faith}(X) = R^2\big(\overline{X}, \overline{\text{CER}}\big), \tag{18}$$

where overlines denote configuration-level means and R^2 denotes the coefficient of determination. Second, *ranking consistency* checks whether structural diagnostics preserve the vulnerability ordering suggested by terminal judges such as CER and ΔmAP. Together, these criteria ensure that $\mathcal{R}$

provides externally meaningful explanations rather than merely internally defined scores.

5 Experiments

5.1 Experimental Setup

We instantiate $\mathcal{F} = (\mathcal{S}, \Pi, \mathcal{D})$ on a shared pool of 1,000 validation pages from PubLayNet (Zhong et al., 2019) and DocLayNet (Pfitzmann et al., 2022), after filtering pages with fewer than five original spans; the simple/medium/complex split is 14.5%/46.4%/39.1%.

Audited pipelines. We audit two parsers that expose block-level boxes and recognized text: **MinerU v2.7.6** (Wang et al., 2024), based on DocLayout-YOLO (Zhao et al., 2024b) with separate recognition modules, and **PP-StructureV3** (Cui et al., 2025), a cascaded PaddleOCR pipeline using RT-DETR-H and PaddleOCR 3.x. Both are evaluated under the same B-SLR matching thresholds ($\tau_{\text{iou}} = 0.1$, $\tau_{\text{text}} = 0.5$). End-to-end VLM parsers are excluded because they do not expose the box-text outputs required by $\mathcal{D}$. Annotations are used only for sampling, detection metrics, and annotation-dependent exposure descriptors.

Auditing protocols. We use two complementary protocols. **Phase 1** performs fixed-configuration auditing by traversing 29 probe configurations in $\mathcal{S}$, yielding 29,000 records per pipeline. **Phase 2** performs policy-driven auditing by comparing five policy families—random, rule-based, LLM-biased, LLM-neutral, and VLM strategy-only—over the same page pool and shared probe space. Configuration identifiers and policy details are provided in Appendices C.1 and A.3.

Statistical protocol. The main text reports configuration-level analyses, with image fixed-effects regression, per-image Ordinary Least Squares (OLS) regression, Spearman analysis, dose-response binning, and within-configuration quantile checks reported in Appendix C.6.

5.2 Results and Analysis

We first quantify the Footprint Bias at the parser level and test whether the same mismatch propagates to downstream QA and evidence retrieval. We then diagnose the structural mechanisms behind it through B-SLR channels, pathway-specific exposure descriptors, probe-family sensitivity, and policy-driven targeting.

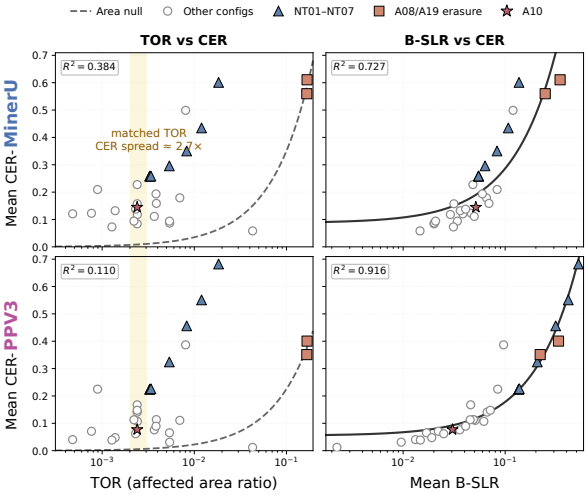

Figure 3: Configuration-level means (n=29) show that B-SLR aligns with perturbation-induced OCR instability substantially better than affected area in both parsers; higher CER and B-SLR indicate greater OCR instability and structural loss.

5.2.1 Footprint Bias Quantification

Figure 3 tests whether affected area is a reliable severity proxy for perturbation-induced OCR instability. It is not: TOR explains CER only weakly on MinerU (R^2=0.384) and almost not at all on PP-StructureV3 (R^2=0.110). Even within the matched-TOR region, configurations with comparable footprint exhibit a CER spread of roughly 2.7×, showing that footprint alone cannot determine whether a perturbation is structurally harmful.

By contrast, realized structural loss measured by B-SLR is much more aligned with the terminal OCR signal, with R^2=0.727 on MinerU and 0.916 on PP-StructureV3. Configurations that appear anomalous under a footprint-only view return to the fitted trend once structural loss is considered. Thus, the failure signal is not explained by how much area is perturbed alone, but by whether the perturbation breaks parser-visible structure.

5.2.2 Downstream Impact

Table 1: **Downstream propagation.** All values are percentages. EM: exact match; A@5: answer hit at top 5. AM/Str./LA: area-matched/structural/large-area. Bold marks the worse AM–Str. result.

Parser	Cond.	TOR	EM↑	Drop↓	Miss↓	BM25 A@5↑	Dense A@5↑
MinerU	Clean	0.00	63.5	–	0.0	94.6	93.6
	AM	0.10	63.1	0.4	1.1	93.8	92.8
	Str.	0.11	56.2	**7.3**	**13.7**	**81.0**	**80.4**
	LA	16.61	38.8	24.7	36.8	58.8	59.4
PP-V3	Clean	0.00	63.6	–	0.0	94.7	95.1
	AM	0.10	63.1	0.5	2.3	92.9	93.4
	Str.	0.11	58.1	**5.5**	**12.2**	**83.1**	**83.5**
	LA	16.61	38.4	25.2	39.6	57.9	58.7

Table 1 tests whether footprint-matched structural damage propagates to downstream document use. Rather than introducing a separate document QA benchmark, we reuse the same 975 clean-derived QA pairs under four counterfactual parser outputs, so that only the perturbation condition changes. For retrieval, the corpus is restricted to chunks from the same page and uses the same block-aware chunking across all conditions; this isolates whether answer-bearing evidence remains available after parsing. The key contrast controls for visual footprint: area-matched erasure (AM) and the structural probe (Str.) perturb nearly the same page area, with TOR around 0.10–0.11%.

Under this matched footprint, AM remains close to the clean condition, whereas Str. consistently damages both answering and evidence retrieval. Across the two parsers, Str. increases EM drop to 5.5–7.3 points, raises Miss to 12–14%, and lowers BM25/Dense A@5 from above 92% to roughly 80–84%. LA causes the largest absolute degradation, but with about 150× larger footprint. Thus, downstream failure follows structural interaction rather than footprint alone: a small structure-targeted probe can remove or disorganize answer-bearing evidence even when an area-matched erasure barely affects the task.

5.2.3 B-SLR Channels and Faithfulness

(a) Channel partition

	B-SLR	IoU	text
MinerU	.071	.028	**.043**
PP-V3	.111	.026	**.085**

(b) Cross-channel coupling

	ρ_s	R^2_{IoU}	R^2_{text}
MinerU	.891	**.927**	.893
PP-V3	.380	.186	**.900**

Table 2: Configuration-level B-SLR channel decomposition (n=29): mean channel losses and cross-channel coupling (ρ_s for IoU–text rank correlation; R^2 columns are univariate OLS channel–CER fits across configuration means).

Returning to the parser-level diagnostic, Table 2 decomposes composite B-SLR into an IoU-fail channel, where no valid geometric match survives, and a text-fail-only channel, where geometry is retained but textual identity is lost. Text-side failure is non-redundant and dominant in both pipelines: it exceeds the IoU channel on MinerU (.043 vs. .028) and more strongly on PP-V3 (.085 vs. .026), showing that many structural failures preserve coarse geometry while losing recognized content.

The channel coupling differs sharply across architectures. MinerU shows coherent geometry-text variation ($\rho_s = .891$), whereas PP-V3 is driven mainly by the text channel ($R^2_{\text{text}} = .900$ vs. $R^2_{\text{IoU}} = .186$). Thus, B-SLR does not collapse to a single text-only score: it captures architecture-specific geometry–text failure coupling while remaining aligned with OCR instability, reaching $R^2 = .727/.916$ and Spearman $\rho = .911/.973$ on MinerU/PP-V3.

5.2.4 Exposure Descriptors Predict Failure Pathways

		$\mathcal{G}_{\text{area}}$			$\mathcal{G}_{\text{struct}}$	
Pipeline	Pathway	TOR	ACR	BPO	BOC	EIR
MinerU	SLR_{miss}	.927	.907	**.964**	.071	.044
	SLR_{topo}	.004	.005	.003	.610	**.623**
PP-V3	SLR_{miss}	.916	.895	**.960**	.072	.073
	SLR_{topo}	.009	.008	.008	**.708**	.702

Table 3: Configuration-level R^2 (n=29) from univariate OLS fits between each diagnostic descriptor and pathway loss. Bold marks the strongest predictor per row.

We next ask which exposure granularity predicts each failure pathway. Table 3 regresses the two pathway losses, SLR_{miss} and SLR_{topo}, on the five diagnostic descriptors, grouped into the area family $\mathcal{G}_{\text{area}} = (\text{TOR}, \text{ACR}, \text{BPO})$ and the structure family $\mathcal{G}_{\text{struct}} = (\text{BOC}, \text{EIR})$.

The specialization is consistent across both pipelines. For SLR_{miss}, BPO is the strongest predictor (R^2=0.964/0.960 on MinerU/PP-V3), with TOR and ACR also high but weaker. For SLR_{topo}, the pattern reverses: area-family descriptors drop near zero, whereas BOC/EIR dominate (R^2=0.610/0.623 on MinerU and 0.708/0.702 on PP-V3). Thus, the two descriptor families are not interchangeable severity measures: boundary-sensitive area exposure predicts direct occlusion, while element-level structural interference predicts topology-dominant loss.

This addresses Q_2: footprint becomes informative only after being refined by structural contact. As exposure descriptors computed before terminal scoring, BPO and EIR indicate different risks: boundary overlap signals direct occlusion, while element interference signals topology-dominant disruption. The NT sweep in Section 5.2.5 exemplifies this regime, increasing structural interference without a matched increase in TOR. A ranking-consistency check against CER yields the same family-level ordering, indicating that the specialization is not an artifact of B-SLR alone.

Parser	Policy	Exposure / Budget				Damage			Failure Pathway			Efficiency	
		TOR	BPO	BOC	EIR	CER↑	ΔmAP↑	B-SLR↑	Miss	Topo	TopoShare	Eff-B↑	Eff-C↑
MinerU	Random	.035	.050	.331	.357	.226	.043	.088	.018	.070	.82	10.9	24.5
	Rule-based	.005	.015	.292	.352	.245	.048	.077	.000	.077	1.00	23.8	61.4
	LLM-biased	.006	.026	.157	.182	.177	.025	.068	.003	.065	.94	16.0	35.7
	LLM-neutral	.008	.011	.125	.145	.209	.034	.058	.003	.055	.91	12.2	40.2
	VLM	.003	.007	.099	.124	.176	.014	.056	.000	.056	1.00	**25.3**	**73.3**
PP-V3	Random	.035	.050	.331	.312	.227	.026	.077	.013	.064	.80	9.1	22.1
	Rule-based	.005	.015	.292	.312	.230	.037	.077	.001	.076	.98	**17.6**	**33.3**
	LLM-biased	.007	.027	.155	.125	.123	.012	.049	.003	.046	.88	9.4	18.1
	LLM-neutral	.008	.012	.127	.129	.119	.024	.048	.004	.044	.83	10.7	20.5
	VLM	.003	.007	.104	.092	.081	.005	.034	.000	.033	.98	15.9	30.5

Table 4: **Phase 2 policy audit.** Under the shared probe space and page pool, targeted policies reveal the expected efficiency pattern: structural loss is dominated by topology-sensitive failures and is achieved with much smaller footprint budgets than random sampling. Metric details are provided in Appendix B.7.

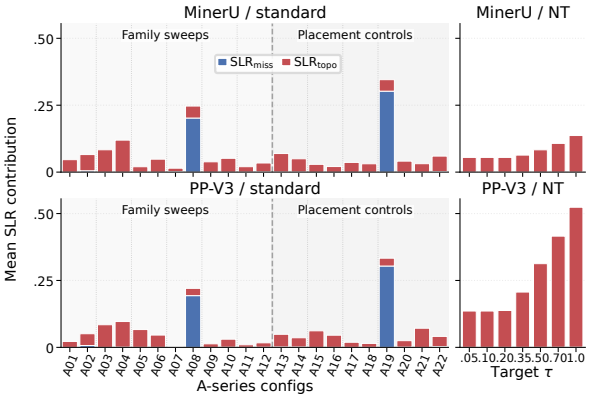

Figure 4: **Phase 1 pathway decomposition.** Bars decompose B-SLR into $\mathrm{SLR}_{\mathrm{miss}}$ and $\mathrm{SLR}_{\mathrm{topo}}$; higher bars indicate greater structural loss, and configuration identifiers are decoded in Appendix C.1.

5.2.5 Sensitivity Across Probe Families

Figure 4 reports the Phase 1 fixed-configuration audit, where the targeting rule is fixed and variation mainly reflects probe-level effects. Damage is highly concentrated: the largest A-series spikes appear at A08 and A19 for both pipelines, indicating that severe erasure-style probes are disproportionately destructive, while most other configurations remain much weaker. The dominant pathway also varies by configuration: extreme erasure is largely miss-driven, whereas many non-extreme configurations are dominated by $\mathrm{SLR}_{\mathrm{topo}}$, indicating topology-level disruption rather than direct element removal.

The NT sweep further reveals parser-dependent sensitivity to sustained structural interference. MinerU increases gradually as the target τ increases, whereas PP-StructureV3 shows a steeper rise in topology-dominant loss at higher target levels. Overall, Phase 1 shows that the probe space contains qualitatively different failure regimes: severe erasure produces miss-heavy spikes, while structure-targeted interference increasingly induces topology-dominant failure.

5.2.6 Targeting Efficiency Across Policies

Table 4 reports the Phase 2 policy audit, where all policies share the same probe space and page pool, so differences reflect targeting logic rather than attack strength. Targeted policies consistently trade footprint for structural contact: Rule-based matches Random on PP-V3 B-SLR with about one seventh of the TOR, and VLM/Rule-based achieve the highest per-footprint efficiencies on MinerU/PP-V3. Their high TopoShare and near-zero Miss indicate that this efficiency mainly comes from topology-sensitive disruption rather than direct physical occlusion. Thus, under a shared $\mathcal{S}$, vulnerability depends not only on how much area a policy perturbs, but on whether it places probes where layout structure is sensitive; Appendix D.6 shows the same rank-consistency trend.

6 Conclusion

We studied how document parsers break under controlled visual perturbations and identified *Footprint Bias*: pixel footprint alone can mis-rank structural vulnerability. We proposed **ProSA**, a lightweight output-level framework that decouples controlled probing, policy-guided selection, and structure-aware diagnosis. Across MinerU and PP-StructureV3, B-SLR tracks OCR instability more faithfully than affected area, exposure descriptors separate occlusion- and topology-dominant pathways, and targeted probes reveal disproportionate QA/retrieval degradation under small area budgets. Overall, DLA robustness should be audited through parser-visible structural loss and failure pathways, not footprint alone.

Limitations

This work focuses on diagnostic auditing rather than universal robustness certification. First, our experiments instantiate the framework on representative DLA pipelines and layout benchmarks; extending the audit to broader parser families and document domains is a natural direction for future work. Second, our current implementation uses explicit layout elements and recognized text as the shared output interface; adapting the framework to less structured document understanding systems may require additional output alignment modules.

Future work. Future work can extend the audit to more parser families, develop adapters for end-to-end or generative document parsers, and connect structural diagnostics with downstream tasks such as retrieval-augmented document question answering.

Ethics Statement

This work studies vulnerability auditing for defensive robustness evaluation. We report attack-like procedures solely to improve safety testing and mitigation in document intelligence systems.

Reproducibility Statement

We specify probe families, the matching protocol, and evaluation metrics in equations and tables; figure paths in the source repository correspond to the plots reported here.

References

Youngmin Baek, Daehyun Nam, Sungrae Park, Junyeop Lee, Seung Shin, Jeonghun Baek, Chae Young Lee, and Hwalsuk Lee. 2020. Cleval: Character-level evaluation for text detection and recognition tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. 2017. Adversarial patch. *arXiv preprint arXiv:1712.09665*.

Yufan Chen, Jiaming Zhang, Kunyu Peng, Junwei Zheng, Ruiping Liu, Philip Torr, and Rainer Stiefelhagen. 2024. Rodla: Benchmarking the robustness of document layout analysis models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15556–15566.

Jonathan Pattin Cottet, Véronique Eglin, and Alex Aussem. 2026. Lightweight domain-specific language model for real-time structuring of medical prescriptions. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 915–926, Rabat, Morocco. Association for Computational Linguistics.

Cheng Cui, Ting Sun, Manhui Lin, Tingquan Gao, Yubo Zhang, Jiaxuan Liu, Xueqing Wang, Zelun Zhang, Changda Zhou, Hongen Liu, Yue Zhang, Wenyu Lv, Kui Huang, Yichao Zhang, Jing Zhang, Jun Zhang, Yi Liu, Dianhai Yu, and Yanjun Ma. 2025. Paddleocr 3.0 technical report. *Preprint*, arXiv:2507.05595.

Terrance DeVries and Graham W. Taylor. 2017. Improved regularization of convolutional neural networks with cutout. *Preprint*, arXiv:1708.04552.

Yongkun Du, Pinxuan Chen, Xuye Ying, and Zhineng Chen. 2025. Docptbench: Benchmarking end-to-end photographed document parsing and translation. *Preprint*, arXiv:2511.18434.

Alexander Groleau, Kok Wei Chee, Stefan Larson, Samay Maini, and Jonathan Boarman. 2023. Augraphy: A data augmentation library for document images. In *Document Analysis and Recognition - ICDAR 2023*, pages 384–401, Cham. Springer Nature Switzerland.

Dan Hendrycks and Thomas Dietterich. 2019. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*.

Shai Kimhi, Moshe Kimhi, and Avi Mendelson. 2025. Benchmarking adversarial patch selection and location. *Mathematics*, 14(1):103.

Chae Young Lee, Youngmin Baek, and Hwalsuk Lee. 2019. Tedeval: A fair evaluation metric for scene text detectors. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 7, pages 14–17.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Claudio Michaelis, Benjamin Mitzkus, Robert Geirhos, Evgenia Rusak, Oliver Bringmann, Alexander S. Ecker, Matthias Bethge, and Wieland Brendel. 2020. Benchmarking robustness in object detection: Autonomous driving when winter is coming.

Linke Ouyang, Yuan Qu, Hongbin Zhou, Jiawei Zhu, Rui Zhang, Qunshu Lin, Bin Wang, Zhiyuan Zhao, Man Jiang, Xiaomeng Zhao, Jin Shi, Fan Wu, Pei Chu, Minghao Liu, Zhenxiang Li, Chao Xu, Bo Zhang, Botian Shi, Zhongying Tu, and Conghui He. 2025. Omnidocbench: Benchmarking diverse pdf document parsing with comprehensive annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24838–24848.

Vitali Petsiuk, Abir Das, and Kate Saenko. 2018. Rise: Randomized input sampling for explanation of black-box models. *Preprint*, arXiv:1806.07421.

Birgit Pfitzmann, Christoph Auer, Michele Dolfi, Ahmed S. Nassar, and Peter Staar. 2022. Doclaynet: A large human-annotated dataset for document-layout segmentation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, pages 3743—3751, New York, NY, USA. Association for Computing Machinery.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1135–1144, New York, NY, USA. Association for Computing Machinery.

Nobuhiro Ueda, Yuyang Dong, Krisztián Boros, Daiki Ito, Takuya Sera, and Masafumi Oyamada. 2026. SCAN: Semantic document layout analysis for textual and visual retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 1618–1637, Rabat, Morocco. Association for Computational Linguistics.

Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, Bo Zhang, Liqun Wei, Zhihao Sui, Wei Li, Botian Shi, Yu Qiao, Dahua Lin, and Conghui He. 2024. Mineru: An open-source solution for precise document content extraction. *Preprint*, arXiv:2409.18839.

Zenghui Yang, Xingquan Zuo, Hai Huang, Gang Chen, Xinchao Zhao, and Tianle Zhang. 2026. IMPACT: Irregular multi-patch adversarial composition based on two-phase optimization. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Junyuan Zhang, Qintong Zhang, Bin Wang, Linke Ouyang, Zichen Wen, Ying Li, Ka-Ho Chow, Conghui He, and Wentao Zhang. 2025. Ocr hinders rag: Evaluating the cascading impact of ocr on retrieval-augmented generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 17443–17453.

Yilun Zhao, Yitao Long, Hongjun Liu, Ryo Kamoi, Linyong Nan, Lyuhao Chen, Yixin Liu, Xiangru Tang, Rui Zhang, and Arman Cohan. 2024a. DocMath-eval: Evaluating math reasoning capabilities of LLMs in understanding long and specialized documents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16103–16120, Bangkok, Thailand. Association for Computational Linguistics.

Zhiyuan Zhao, Hengrui Kang, Bin Wang, and Conghui He. 2024b. Doclayout-YOLO: Enhancing document layout analysis through diverse synthetic data and global-to-local adaptive perception.

Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. 2019. Publaynet: Largest dataset ever for document layout analysis. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1015–1022.

Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. 2020. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, pages 13001–13008.

A Implementation Details of Probe and Policy Spaces

A.1 Probe Space and Probe Catalog

Each probe configuration is specified by four dimensions:

$$P = \langle \mathcal{H}, \mathcal{V}, \mathcal{B}, \mathcal{T} \rangle, \tag{19}$$

where $\mathcal{H}$ denotes geometry, $\mathcal{V}$ visual appearance, $\mathcal{B}$ composition behavior, and $\mathcal{T}$ placement strategy. This factorization keeps the perturbation space modular: probe geometry, visual appearance, composition behavior, and placement can be controlled or varied while all policies draw from the same admissible space.

Table 5 summarizes the probe catalog used throughout the audit. The catalog covers line-like artifacts, area erasures, local overlays, point clusters, and irregular patches, so that the shared probe space contains both footprint-heavy and structure-sensitive perturbations. Figure 5 visualizes representative examples of these probe families.

A.2 Probe Generation and Placement

This subsection specifies how the probe catalog in Table 5 is rendered on a document page. We denote geometry masks by $\mathcal{H}_{(\cdot)}$ to distinguish probe geometry from the diagnostic exposure families in the main text. The geometry primitives are imple-

Probe	Geom.	Behavior	Key parameters	Placement $\mathcal{T}$
P1 (horiz. crease)	line	inject	$w \in [1, 10],\ l_r \in [0.5, 1.0]$	a/c/r
P2 (vert. crease)	line	inject	$w \in [1, 10],\ l_r \in [0.5, 1.0]$	a/c/r
P3 (circular overlay)	disk	blend	$r \in [30, 90],\ \alpha \in [0.2, 1.0]$	a/c/r
P4 (rect. modification)	rect	erase	$a_{\text{area}} \in [3\%, 25\%],\ \beta \in [0.2, 1.0]$	c/b
P5 (thin horiz. line)	line	inject	$w \in [1, 5],\ l_r \in [0.2, 0.8]$	b/c/r
P6 (gradient band)	line+grad	blend	$\alpha \in [0.05, 0.4],\ w \in [2, 10]$	a
P7 (dot cluster)	points	inject	$n \in [10, 100],\ r \in [1, 4],\ \sigma \in [10, 50]$	r
P8 (irregular patch)	blob	blend	$r_b \in [30, 80],\ \kappa \in [0.1, 0.5],\ \alpha \in [0.3, 0.7]$	a
P9 (diag. crease)	line	inject	$\theta \in [20^\circ, 70^\circ],\ w \in [1, 6]$	a

Table 5: Probe catalog and parameter ranges. Placement codes a/c/r/b denote anchor/content/random/bridge.

Figure 5: Representative visual examples of the probe families used in the controlled perturbation space.

mented as binary masks:

$$\mathcal{H}_{\text{line}}(\theta, l, w) = \{(x, y) : |x \sin\theta - y \cos\theta| < \tfrac{w}{2},\ 0 \le x\cos\theta + y\sin\theta \le l\}, \quad (20)$$

$$\mathcal{H}_{\text{disk}}(r) = \{(x, y) : x^2 + y^2 \le r^2\}, \quad (21)$$

$$\mathcal{H}_{\text{rect}}(w_r, h_r) = \{(x, y) : |x| \le w_r/2,\ |y| \le h_r/2\}, \quad (22)$$

$$\mathcal{H}_{\text{blob}}(r_b, \kappa) = \{(r, \phi) : r \le r_b\big(1 + \kappa \cdot \text{Perlin}(\phi)\big)\}, \quad (23)$$

$$\mathcal{H}_{\text{point}}(n, r, \sigma) = \bigcup_{i=1}^{n} \mathcal{H}_{\text{disk}}(r; c_i),\quad c_i \sim \mathcal{N}(\mu, \sigma^2 I). \quad (24)$$

Appearance families include solid color, gradient, ring, and procedural texture. All probe actions are unified by alpha blending:

$$I'(x, y) = \big(1 - \alpha(x, y)\big) I(x, y) + \alpha(x, y) q(x, y), \quad (25)$$

where $q(x, y)$ is the probe texture or background target. The three composition behaviors in Table 5 are instantiated as *inject* ($\alpha = 1$), *blend* ($\alpha \in (0, 1)$), and *erase* (blending toward the local background color).

Placement strategies are defined over content and boundary masks derived from the clean-layout boxes. Let $\{b_i\}$ denote clean-layout boxes and ∂b_i their boundaries. The anchor region is

$$M_{\text{anchor}} = \text{Dil}\left(\bigcup_i \partial b_i, \delta\right) \setminus M_{\text{content}}^{\text{erode}(\delta)}, \quad (26)$$

where $\delta = 5$ px (Appendix B.5). We use four placement strategies: **anchor** (a, on M_{anchor}), **content** (c, inside M_{content}), **random** (r, uniformly over the page), and **bridge** (b, at the midpoint between two neighboring blocks). Across experiments, one to three probes are placed per page under a global area budget.

A.3 Policy Families

The policy space Π maps page context to a probe configuration in the shared probe space of Table 5. Formally, each policy is written as

$$\pi(\text{context}) = \hat{P} = \langle \hat{\mathcal{H}}, \hat{\mathcal{V}}, \hat{\mathcal{B}}, \hat{\mathcal{T}} \rangle, \quad (27)$$

where the context features are extracted from the current page, and $\hat{P}$ follows the same geometry–appearance–behavior–placement schema as Section A.1.

We use five non-iterative policy families under the identical probe space:

- **Random** (π_{rand}): uniformly samples the probe type, parameters, and placement strategy.
- **Rule-based** (π_{rule}): selects probes using

boundary density and inter-block gap heuristics.

- **LLM-biased** ($\pi_{\text{LLM}}^{\text{biased}}$): uses a language-guided prompt with explicit structural hints.

- **LLM-neutral** ($\pi_{\text{LLM}}^{\text{neutral}}$): uses a language-guided prompt without explicit structural hints.

- **VLM strategy-only** (π_{VLM}): predicts probe type and placement strategy from the page image without direct coordinate output.

All policies emit the same schema—probe type, parameters, and placement strategy—so differences in observed damage can be attributed to targeting logic rather than search-space mismatch.

A.4 Controlled Prompt Design

The biased and neutral LLM prompts are controlled variants designed to test whether structural targeting is driven by explicit hints or by page-level context. Table 6 summarizes their differences in role framing, objective wording, strategy labels, context rendering, and hint strength. If the neutral prompt still favors boundary-oriented strategies, we treat this as evidence that the preference is not solely induced by explicit structural hints.

B Diagnostic Metrics, Matching, and Terminal Evaluation

B.1 Exposure Descriptor Definitions

This subsection gives the operational definitions of the five exposure descriptors used in Section 4.2. Let $A(P)$ denote the perturbation mask induced by probe configuration P, and let (W, H) be the page size. For annotated layout elements L, we define the annotated layout support and boundary support as

$$\begin{aligned} \Omega_L &= \bigcup_{\ell \in L} b(\ell), \\ M_{\partial L} &= \mathrm{Dil}(\cup_{\ell \in L} \partial b(\ell), \delta) \setminus \Omega_L^{\mathrm{erode}(\delta)}. \end{aligned} \tag{28}$$

The five descriptors are computed as

$$\begin{aligned} \mathrm{TOR}(P) &= \frac{|A(P)|}{W \times H}, \\ \mathrm{ACR}(P, L) &= \frac{|A(P) \cap \Omega_L|}{|\Omega_L|}, \\ \mathrm{BPO}(P, L) &= \frac{|A(P) \cap M_{\partial L}|}{|M_{\partial L}|}, \\ \mathrm{BOC}(P, L) &= \frac{|\{\ell \in L : b(\ell) \cap A(P) \neq \emptyset\}|}{|L|}, \\ \mathrm{EIR}(P, E) &= \frac{|\{e \in E : b(e) \cap A(P) \neq \emptyset\}|}{|E|}. \end{aligned} \tag{29}$$

TOR and EIR are output-derived, whereas ACR, BPO, and BOC are annotation-dependent benchmark diagnostics. For BOC and EIR, the intersection predicate is satisfied when at least one pixel of the bounding box overlaps the probe mask (`overlap_px=1`).

B.2 Output Matching and TextSim

Each clean element $e \in E$ is assigned a candidate counterpart in the perturbed output by maximum-IoU lookup:

$$m(e) = \arg\max_{e' \in E_{\text{adv}}} \mathrm{IoU}(e, e'). \tag{30}$$

Ties are broken by adversarial-element index, and no one-to-one Hungarian assignment is enforced. This allows multiple clean elements to map to the same perturbed element, which is necessary for detecting merge-type topology failures.

A match is accepted only when both the geometry and text gates are satisfied:

$$\begin{aligned} \mathrm{IoU}(e, m(e)) &\geq \tau_{\text{iou}}, \\ \mathrm{TextSim}(e, m(e)) &\geq \tau_{\text{text}}. \end{aligned} \tag{31}$$

We use $\tau_{\text{iou}} = 0.1$ and $\tau_{\text{text}} = 0.5$ in all experiments; their rationale and sensitivity are discussed in Appendix B.5.

Before computing text consistency, both strings are stripped of leading/trailing whitespace and case-folded to lowercase; internal whitespace is preserved, and no punctuation removal or tokenization is applied. Let $\tilde{a}$ and $\tilde{b}$ denote the normalized strings. We compute text consistency by character-level longest common subsequence:

$$\mathrm{TextSim}(a, b) = \begin{cases} 1, & |\tilde{a}| = |\tilde{b}| = 0, \\ \frac{|\mathrm{LCS}(\tilde{a}, \tilde{b})|}{\max(|\tilde{a}|, |\tilde{b}|)}, & \text{otherwise.} \end{cases} \tag{32}$$

TextSim is used only as a matching criterion for B-SLR, not as the CER computation itself.

Dimension	LLM-biased	LLM-neutral
System role	Expert adversarial tester for DLA	QA analyst for robustness
Objective	Max parsing disruption (merge/split/miss)	Test model response to degradation
Strategy labels	bridge/anchor/content/random	between/edge/inside/anywhere
Context source	`to_text_description()` with gap info	`to_neutral_description()` coords only
Hint strength	Explicit spatial guidance	Minimal; no structural cues

Table 6: Controlled prompt-pair design for testing structural targeting preference.

B.3 Canonical Label Mapping

Table 7 reports the label normalization used for detection evaluation and category-level diagnostics. Raw dataset and parser labels are mapped to the five canonical layout categories $\mathcal{C}_5 = \{\text{text}, \text{title}, \text{table}, \text{figure}, \text{equation}\}$; non-content labels are ignored for mAP.

B.4 Terminal Metric Definitions

We instantiate the terminal component Δ through OCR and detection degradation:

$$\Delta = \big(\Delta_{\text{ocr}}, \Delta_{\text{det}}\big), \tag{33}$$

where Δ_{ocr} is measured by CER and Δ_{det} by the per-image mAP@0.5 drop.

For the OCR channel, we use character error rate (CER) as the primary terminal judge. Because the audit compares clean and perturbed parser outputs, the clean parser text serves as the reference text for each matched element. Using the maximum-IoU correspondence defined in Appendix B.2, let $r_e = s(e)$ and $h_e = s(m(e))$. The element-wise CER is

$$\text{CER}_e(e) = \begin{cases} \dfrac{d_{\text{Lev}}(r_e, h_e)}{|r_e|}, & \text{IoU}(e, m(e)) > 0, \\ 1, & \text{otherwise}, \end{cases} \tag{34}$$

where the second case represents total text loss when no perturbed element overlaps the clean element. Here d_{Lev} denotes character-level Levenshtein distance after stripping leading/trailing whitespace and case-folding to lowercase.

Let

$$E_{\text{text}} = \{e \in E \mid s(e) \neq \epsilon\} \tag{35}$$

denote the subset of clean elements with non-empty text. The page-level mean CER is

$$\overline{\text{CER}} = \frac{1}{|E_{\text{text}}|} \sum_{e \in E_{\text{text}}} \text{CER}_e(e), \tag{36}$$

with $\overline{\text{CER}} = 1$ when $|E_{\text{text}}| = 0$.

For the detection channel, we compute per-image mean Average Precision at IoU threshold 0.5:

$$\text{mAP} = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \text{AP}_c, \tag{37}$$

where $\mathcal{C} \subseteq \mathcal{C}_5$ is the set of normalized layout classes present in the page-level ground truth. The raw mAP is a detection quality score, so we use its clean-to-perturbed drop as the detection degradation signal:

$$\begin{aligned} \Delta_{\text{det}} &:= \Delta\text{mAP} \\ &= \text{mAP}(\mathcal{Y}, E) - \text{mAP}(\mathcal{Y}, E_{\text{adv}}). \end{aligned} \tag{38}$$

Positive ΔmAP indicates detection degradation, while negative values indicate an incidental mAP improvement after perturbation. CER remains the primary terminal judge for OCR-oriented failure propagation.

B.5 Operational Constants and Threshold Rationale

Coverage ratio ρ. The coverage ratio

$$\rho(e, P) = \frac{|b(e) \cap A(P)|}{|b(e)|} \tag{39}$$

measures the fraction of an element box directly covered by the perturbation mask. A structurally failed element with $\rho(e, P) \geq \eta_{\text{occ}}$ is assigned to the direct-occlusion subset $\mathcal{U}_{\text{miss}}$; otherwise, it is assigned to the topology-dominant subset $\mathcal{U}_{\text{topo}}$.

Alignment threshold rationale. The IoU threshold $\tau_{\text{iou}} = 0.1$ is intentionally permissive because the IoU gate is used for diagnostic alignment rather than detection-quality scoring. Unlike standard detection evaluation, where IoU thresholds such as 0.5 determine whether a detection is correct, our matching stage only admits a candidate perturbed element for subsequent identity checking. A strict IoU gate would prematurely mark spatially displaced but still identifiable counterparts as lost, thereby conflating geometric displacement with structural disappearance. This follows the spirit of text-aware detection evaluation, where geometric matching can be relaxed when text-level consistency provides the substantive quality signal (Lee et al., 2019; Baek et al., 2020). The TextSim threshold $\tau_{\text{text}} = 0.5$ requires at least half of the character-level LCS-normalized content to be preserved; below this value, textual identity is treated

Source family	Raw label(s)	Canonical target	Used in mAP
PubLayNet GT	Text; List	text	yes
PubLayNet GT	Title	title	yes
PubLayNet GT	Table	table	yes
PubLayNet GT	Figure	figure	yes
DocLayNet GT	Caption; Footnote; List-item	text	yes
DocLayNet GT	Section-header	title	yes
DocLayNet GT	Picture	figure	yes
DocLayNet GT	Formula	equation	yes
Parser / MinerU output	figure_caption; table_caption; reference; list; plain_text; table_footnote; formula_caption; index; normal_text	text	yes
Parser / MinerU output	image	figure	yes
Parser / MinerU output	formula; isolate_formula; embedding; isolated	equation	yes
Canonical passthrough	text; title; table; figure; equation	identity	yes
Non-content labels	Page-header; Page-footer; header; footer; abandon; seal	None	no
Fallback rule	Any unseen label	text (default)	yes

Table 7: Label normalization from raw dataset/parser labels to the five canonical layout categories used for detection evaluation.

Symbol	Value	Role	Code reference
τ_{iou}	0.1	IoU gate for output matching	`calculate_b_slr()`
τ_{text}	0.5	TextSim gate for textual identity	`calculate_b_slr()`
η_{occ}	0.3	Occlusion threshold for miss/topo split	`_elem_is_occluded()`
δ	5 px	Boundary-mask dilation radius	`compute_anchor_mask()`
$\text{overlap}_{\text{px}}$	1 px	BOC/EIR box-intersection threshold	`compute_boc()`

Table 8: Operational constants fixed across all experiments.

as structurally lost. Both thresholds are fixed across all parsers, probes, and policies.

B.6 B-SLR Decomposition Algorithm

Algorithm 1 gives the implementation-level procedure used to decompose B-SLR into direct-occlusion and topology-dominant components. The algorithm first identifies structurally failed elements using the same IoU–TextSim gates as the main B-SLR definition. Failed elements are then split into direct-occlusion and topology-dominant subsets according to the coverage ratio $\rho(e, P)$. Within the topology-dominant subset $\mathcal{U}_{\text{topo}}$, failures are further assigned to three mutually exclusive categories under a fixed priority order: MERGE, MISCLASS, and DEGRADED. MERGE captures multiple clean elements assigned to the same perturbed prediction; MISCLASS captures retained spatial correspondence with a changed canonical category; and DEGRADED covers the remaining non-occluded failures, including text degradation without direct physical coverage.

B.7 Policy-Audit Metric Shorthands

Table 4 uses compact metric names for readability. The exposure descriptors TOR, BPO, BOC, and EIR follow the definitions in Appendix B.1. CER and ΔmAP are the terminal OCR and detection-side degradation signals defined in Appendix B.4, and B-SLR is the structural loss rate defined in Section 4.2. In the policy audit, *Miss* and *Topo* denote SLR_{miss} and SLR_{topo}, respectively. TopoShare measures the fraction of structural loss explained by topology-dominant failures:

$$\text{TopoShare} = \frac{\text{SLR}_{\text{topo}}}{\text{SLR}_{\text{miss}} + \text{SLR}_{\text{topo}}}. \tag{40}$$

Eff-B and Eff-C denote per-footprint efficiency for structural loss and OCR instability:

$$\text{Eff-B} = \frac{\text{B-SLR}}{\text{TOR}}, \qquad \text{Eff-C} = \frac{\text{CER}}{\text{TOR}}. \tag{41}$$

These efficiency scores are used only as relative indicators under comparable small-footprint regimes; they should be interpreted together with absolute damage and pathway composition.

C Experimental Protocol and Configuration Details

C.1 Auditing Protocol Overview and Identifier Convention

Table 9 summarizes the configuration groups used in our experiments. We use A01–A22 for fixed standard configurations, NT01–NT07 for EIR-targeted configurations, and S01–S13 for randomized sweep configurations. For compact reporting, each fixed configuration is encoded as

$$\mathsf{P}_n \cdot t[\xi], \tag{42}$$

where P_n identifies the probe family in Table 5, $t \in \{\mathtt{a}, \mathtt{c}, \mathtt{r}, \mathtt{b}\}$ denotes placement, and ξ records the

Algorithm 1: Decomposed B-SLR classification.

Input: $E,\ E_{\text{adv}},\ A(P)$
Output: $\text{SLR}_{\text{miss}},\ \text{SLR}_{\text{topo}}$, per-category counts

Compute $m(e) = \arg\max_{e' \in E_{\text{adv}}} \text{IoU}(e, e')$ for each $e \in E$
Count how many clean elements map to each perturbed prediction
foreach $e \in E$ **do**
  $\hat{e} \leftarrow m(e)$
  $\text{iou} \leftarrow \text{IoU}(e, \hat{e})$
  $\hat{s} \leftarrow \text{TextSim}(e, \hat{e})$
  $\text{failed} \leftarrow (\text{iou} < \tau_{\text{iou}}) \vee (\hat{s} < \tau_{\text{text}})$
  if *not* failed **then**
    INTACT
  else if $\rho(e, P) \geq \eta_{\text{occ}}$ **then**
    MISS
  else if *multiple clean elements map to* $\hat{e}$ **then**
    MERGE
  else if $\text{iou} \geq \tau_{\text{iou}}$ ***and*** $c(e) \neq c(\hat{e})$ **then**
    MISCLASS
  else
    DEGRADED
  end
end
$\text{SLR}_{\text{miss}} \leftarrow n_{\text{MISS}}/|E|$
$\text{SLR}_{\text{topo}} \leftarrow (n_{\text{MERGE}} + n_{\text{MISCLASS}} + n_{\text{DEGRADED}})/|E|$

Protocol	Configs	Per config	Goal
Phase 1 fixed	29	1,000 imgs	probe effect profiling
Phase 1 sweep	13	1,000 imgs	randomized-parameter check
Phase 2 policies	5 policies	1,000 imgs	targeting-policy comparison

Table 9: Overview of auditing protocols and configuration groups.

dominant varied parameter. Tables 10, 11, and 12 provide the full decoding.

C.2 Phase 1 Fixed Configurations

Table 10 decodes the 22 standard Phase 1 configurations. These configurations are used to profile probe-level effects under controlled parameter and placement choices.

Label	Probe tuple label	Key parameters	Purpose
A01	$\mathsf{P}_1 \cdot \mathtt{a}[w1]$	$w=1,\ l=W$	P1 lower bound
A02	$\mathsf{P}_1 \cdot \mathtt{a}[w8]$	$w=8,\ l=W$	P1 upper bound
A03	$\mathsf{P}_2 \cdot \mathtt{a}[w1]$	$w=1,\ l=H$	P2 lower bound
A04	$\mathsf{P}_2 \cdot \mathtt{a}[w8]$	$w=8,\ l=H$	P2 upper bound
A05	$\mathsf{P}_3 \cdot \mathtt{a}[\alpha.3]$	$r=60,\ \alpha=0.3$	stamp faint
A06	$\mathsf{P}_3 \cdot \mathtt{a}[\alpha1]$	$r=60,\ \alpha=1.0$	stamp opaque
A07	$\mathsf{P}_4 \cdot \mathtt{c}[5\%]$	$a_{\text{area}}=5\%,\ \beta=0.3$	erasure mild
A08	$\mathsf{P}_4 \cdot \mathtt{c}[20\%]$	$a_{\text{area}}=20\%,\ \beta=1.0$	erasure severe
A09	$\mathsf{P}_5 \cdot \mathtt{b}[w1]$	$w=1,\ l=0.5W$	separator thin
A10	$\mathsf{P}_5 \cdot \mathtt{b}[w3]$	$w=3,\ l=0.5W$	separator thick
A11	$\mathsf{P}_6 \cdot \mathtt{a}[\alpha.1]$	$\alpha=0.1,\ w=5$	ghost imperceptible
A12	$\mathsf{P}_6 \cdot \mathtt{a}[\alpha.3]$	$\alpha=0.3,\ w=5$	ghost visible
Placement control groups (matched probe, varying $\mathcal{T}$):			
A13	$\mathsf{P}_1 \cdot \mathtt{c}[w3]$	$w=3,\ l=W$	crease on text
A14	$\mathsf{P}_1 \cdot \mathtt{r}[w3]$	$w=3,\ l=W$	crease random
A15	$\mathsf{P}_3 \cdot \mathtt{c}[\alpha.5]$	$r=60,\ \alpha=0.5$	stamp on text
A16	$\mathsf{P}_3 \cdot \mathtt{r}[\alpha.5]$	$r=60,\ \alpha=0.5$	stamp random
A17	$\mathsf{P}_5 \cdot \mathtt{c}[w2]$	$w=2,\ l=0.5W$	sep. on text
A18	$\mathsf{P}_5 \cdot \mathtt{r}[w2]$	$w=2,\ l=0.5W$	sep. random
A19	$\mathsf{P}_4 \cdot \mathtt{b}[20\%]$	$a_{\text{area}}=20\%,\ \beta=1.0$	erasure bridge
A20	$\mathsf{P}_5 \cdot \mathtt{c}[w3]$	$w=3,\ l=0.5W$	sep. extended
A21	$\mathsf{P}_3 \cdot \mathtt{a}[\alpha.5]$	$r=60,\ \alpha=0.5$	stamp anchor
A22	$\mathsf{P}_1 \cdot \mathtt{a}[w3]$	$w=3,\ l=W$	crease anchor

Table 10: Phase 1 fixed configurations under the A01–A22 identifier convention. Tuple labels decode probe family, placement, and dominant parameter; W/H denote page width/height.

Label	Probe tuple label	Target τ	Method
NT01	$\mathsf{NT}[.05]$	0.05	targeted stamp placement
NT02	$\mathsf{NT}[.10]$	0.10	targeted stamp placement
NT03	$\mathsf{NT}[.20]$	0.20	targeted stamp placement
NT04	$\mathsf{NT}[.35]$	0.35	targeted stamp placement
NT05	$\mathsf{NT}[.50]$	0.50	targeted stamp placement
NT06	$\mathsf{NT}[.70]$	0.70	targeted stamp placement
NT07	$\mathsf{NT}[1.0]$	1.00	targeted stamp placement

Table 11: EIR-targeted configurations under the NT01–NT07 identifier convention. The probe family and placement mechanism are fixed; only the target τ varies.

C.3 EIR-Targeted NT Configurations

The NT series uses iterative multi-stamp placement to reach target element-interference levels. The target τ specifies the intended fraction of clean elements whose boxes overlap the probe mask at injection time.

C.4 Randomized Sweep Configurations

In addition to the fixed Phase 1 matrix, we run randomized sweeps to test whether structural-exposure signals persist when probe parameters vary stochastically. Table 12 lists the randomized configurations. Paired configurations share sampled geometry, appearance, and behavior parameters while varying only placement.

ID	Probe	Random parameters	$\mathcal{T}$	Notes
S01	P1	$w \sim U(1, 10)$	anchor	
S02	P2	$w \sim U(1, 10)$	anchor	
S03	P3	$r \sim U(30, 90),\ \alpha \sim U(0.2, 1)$	anchor	
S04	P4	$a_{\text{area}} \sim U(3\%, 25\%),\ \beta \sim U(0.2, 1)$	content	
S05	P5	$w \sim U(1, 5),\ l \sim U(0.2W, 0.8W)$	bridge	
S06	P6	$\alpha \sim U(0.05, 0.4),\ w \sim U(2, 10)$	anchor	
S07	P7	$n \sim U(10, 100),\ r \sim U(1, 4)$	random	
S08	P8	$r_b \sim U(30, 80),\ \kappa \sim U(0.1, 0.5)$	anchor	
S09	P9	$\theta \sim U(20, 70),\ w \sim U(1, 6)$	anchor	
S10	P1	$w \sim U(1, 10)$	content	paired w/ S01
S11	P1	$w \sim U(1, 10)$	random	paired w/ S01
S12	P3	$r \sim U(30, 90),\ \alpha \sim U(0.2, 1)$	content	paired w/ S03
S13	P3	$r \sim U(30, 90),\ \alpha \sim U(0.2, 1)$	random	paired w/ S03

Table 12: Randomized sweep configurations. Paired configurations share sampled $\mathcal{H}$, $\mathcal{V}$, and $\mathcal{B}$; only $\mathcal{T}$ varies.

Parameter	LLM policies	VLM policy
Backend	DeepSeek-Chat	Qwen-VL-Max
Temperature	0.7	0.7
Max tokens	API default	1,024
Response format	`json_object`	free-form (parsed)
Max retries	3	3
Image long edge	—	1,024 px
JPEG quality	—	85

Table 13: LLM/VLM inference parameters.

C.5 Inference, Randomness, and Software

Randomness control. All experiments use `numpy.random.default_rng(seed=42)` as the base random generator. The randomized-sweep paired configurations use deterministic per-image seeds $(\text{image_index} + 1) \times 10^5 + \text{pair_id}$ so that $\mathcal{H}$, $\mathcal{V}$, and $\mathcal{B}$ are sampled identically across paired placement strategies while only $\mathcal{T}$ varies. LLM/VLM policies inherit stochasticity from API sampling; all other components are deterministic given the seed.

Software versions. We use MinerU v2.7.6 with DocLayout-YOLO, PP-StructureV3 with PaddleOCR 3.4.1 and PaddlePaddle 3.3.1, Python 3.10, NumPy 2.2.6, and OpenCV 4.x.

Hardware. GPU experiments used one NVIDIA GeForce RTX 5090 with 32 GiB device memory.

C.6 Statistical Verification Layers

We use the six verification layers in Table 14 to check that the main conclusions do not depend on a single aggregation view. The main text emphasizes configuration-level analyses, while the remaining layers provide robustness checks against document heterogeneity, nonlinearity, and attack-design confounds.

Layer	Method	Unit	Threat checked
0	Raw OLS	per-image record	baseline association
1	Config-level OLS	29 config means	within-image noise
2	Image fixed effects	de-meaned residuals	document heterogeneity
3	Per-image Spearman	per-image ρ + win rate	nonlinearity
4	Dose-response bins	EIR quintile bins	reverse causality
5	Quasi-natural	within-config quartiles	attack-design confound

Table 14: Six-layer statistical verification framework.

		$\mathcal{G}_{\text{area}}$			$\mathcal{G}_{\text{struct}}$	
Pipeline	Channel	TOR	ACR	BPO	BOC	EIR
MinerU	composite	.780	.760	**.821**	.336	.280
	iou-only	.782	.755	**.869**	.194	.149
	text-only	**.629**	.622	.613	.477	.424
PP-V3	composite	.150	.149	.161	**.856**	.852
	iou-only	.704	.678	**.781**	.203	.205
	text-only	.017	.018	.017	**.738**	.733

Table 15: Channel-by-exposure regression breakdown: configuration-level R^2 from univariate OLS fits between each diagnostic descriptor and each B-SLR channel. Bold marks the strongest predictor per row.

D Additional Results and Robustness Checks

D.1 Channel-by-Exposure Breakdown

Table 15 expands the exposure analysis in Section 5.2.4 by reporting configuration-level R^2 between each diagnostic descriptor and each B-SLR channel. This breakdown complements Table 3 by showing how the five exposure descriptors relate to composite B-SLR and its IoU-fail and text-fail-only subchannels.

D.2 Randomized Sweep Robustness

The randomized sweep configurations in Appendix C.4 test whether the structural-exposure signal persists when probe size and appearance vary stochastically rather than being fixed by the Phase 1 matrix. At the configuration level, EIR remains more predictive of B-SLR than footprint-style descriptors: $R^2(\text{EIR} \rightarrow \text{B-SLR}) = 0.613$ on MinerU and 0.530 on PP-V3, while TOR/ACR/BPO reach at most 0.048 on MinerU and 0.020 on PP-V3. BOC shows the same trend ($R^2 = 0.591$ on MinerU and 0.532 on PP-V3). Thus, the dominance of structure-count descriptors is not an artifact of hand-selected fixed configurations.

D.3 Dose-Response Monotonicity

We further check whether structural interference produces a monotonic dose-response in B-SLR. Binning observations by EIR, PP-V3 exhibits strict monotonic growth: mean B-SLR increases from 0.018 in the lowest bin to 0.480 in the highest bin.

MinerU also increases monotonically, from 0.025 to 0.181, but with a shallower slope. Within fixed configurations, partitioning images into EIR quartiles yields monotonically increasing B-SLR from Q1 to Q4 in the majority of configurations for both pipelines. Together with the randomized sweep in Section D.2, this supports that the structural-exposure effect is not an artifact of either specific configuration choices or a single aggregation view.

D.4 Downstream Propagation Details

The downstream propagation experiment tests whether parser-level structural loss affects practical document use. We construct 975 extractive QA pairs from clean parser outputs over a 500-page subset. Each QA pair contains a question, a gold answer span, an evidence text span, and the corresponding page identifier. The QA set covers 349 unique pages, with at most four QA pairs per page. The answer types include span answers, titles, numeric answers, and captions.

Perturbation conditions. We compare four conditions: clean pages, area-matched erasure (AM), structural probe (Str.), and large-area erasure (LA). AM and Str. have nearly identical footprint, with mean TORs of 0.10% and 0.11%, respectively. LA serves as a large-footprint control with mean TOR 16.61%, roughly $150\times$ larger than the matched-footprint conditions.

Downstream QA. For each parser and condition, the parsed page text is serialized as the context for a fixed QA answerer. The answerer is instructed to answer using only the provided parsed text; if the answer is absent, it returns `NOT_FOUND`. We report exact-match accuracy (EM), token-level F1, answer-missing rate, and the EM drop from the clean condition. A QA instance is counted as answer-missing when the gold answer string is absent from the parsed page text. Empty parser outputs are retained and counted as incorrect and answer-missing when applicable.

Page-internal retrieval. For retrieval, the corpus is restricted to chunks from the QA's own page under the same parser and perturbation condition. This page-internal setting aligns retrieval with the single-page QA task and isolates whether the perturbed parser output still contains retrievable evidence. We use block-aware chunking with a target length of 400 characters, minimum length 80, maximum length 700, and overlap 80. Two retrieval backends are evaluated: BM25 and a dense encoder based on `sentence-transformers/all-MiniLM-L6-v2`. For each QA, we retrieve the top 10 chunks.

Retrieval metrics. We report two retrieval signals. Evidence Recall@k checks whether any top-k chunk contains the evidence text, while Answer Hit@k checks whether any top-k chunk contains the gold answer. The main text reports Answer Hit@5 because it directly measures whether the answer-bearing content remains retrievable. We additionally report evidence Recall@5 and evidence MRR@10 in Table 16.

The full retrieval metrics are consistent with the main-text AnsHit@5 results. In the main-text table, BM25 AnsHit@5 and Dense AnsHit@5 refer to Answer Hit@5, i.e., whether any top-5 retrieved chunk contains the gold answer string. Under matched footprint, the structural probe causes much larger retrieval degradation than area-matched erasure across both parsers and both retrieval backends. For example, on MinerU, BM25 evidence R@5 drops from 66.7% under AM to 49.2% under Str., while AnswerHit@5 drops from 93.8% to 81.0%. On PP-V3, the same pattern holds: BM25 evidence R@5 drops from 63.9% to 52.7%, and AnswerHit@5 drops from 92.9% to 83.1%. Thus, the downstream effect is not specific to the QA answerer or to a single retrieval backend.

D.5 Policy Pathway Composition

Figure 6 provides the fine-grained pathway composition behind the Phase 2 policy comparison in Table 4. Across policies, topology-dominant components account for most structural loss, while direct-occlusion loss remains small for targeted policies. This corroborates the main-text observation that efficient targeting exposes topology-level vulnerability rather than simply increasing physical coverage.

D.6 Policy-Level Rank Consistency

We further examine whether the policy-level ordering induced by structural damage is consistent with terminal degradation signals. For each parser, we rank the five Phase 2 policies by B-SLR, CER, and ΔmAP, where rank 1 denotes the most severe degradation. As a footprint-based baseline, we also rank policies by TOR and compare the induced ordering with the same terminal metrics. Figure 7 shows that B-SLR yields a policy ordering that is more consistent with CER and ΔmAP than TOR. The B-SLR-based rank correlations are $0.80/0.80$ on MinerU

Parser	Condition	BM25 R@5$_{\text{evid}}$	BM25 MRR@10$_{\text{evid}}$	BM25 AnsHit@5	Dense R@5$_{\text{evid}}$	Dense MRR@10$_{\text{evid}}$	Dense AnsHit@5
MinerU	Clean	68.7	61.6	94.6	67.9	56.9	93.6
	AM	66.7	59.6	93.8	65.7	55.2	92.8
	Str.	49.2	43.3	81.0	48.4	41.2	80.4
	LA	35.6	31.2	58.8	35.4	28.7	59.4
PP-V3	Clean	66.4	59.3	94.7	64.5	53.1	95.1
	AM	63.9	56.9	92.9	62.2	51.2	93.4
	Str.	52.7	46.8	83.1	51.5	41.5	83.5
	LA	32.6	29.2	57.9	31.9	26.8	58.7

Table 16: Additional page-internal retrieval metrics for downstream propagation. All scores are percentages and higher is better. R@5$_{\text{evid}}$ and MRR@10$_{\text{evid}}$ use evidence-text containment as the relevance signal, while AnsHit@5 checks whether a top-5 retrieved chunk contains the gold answer.

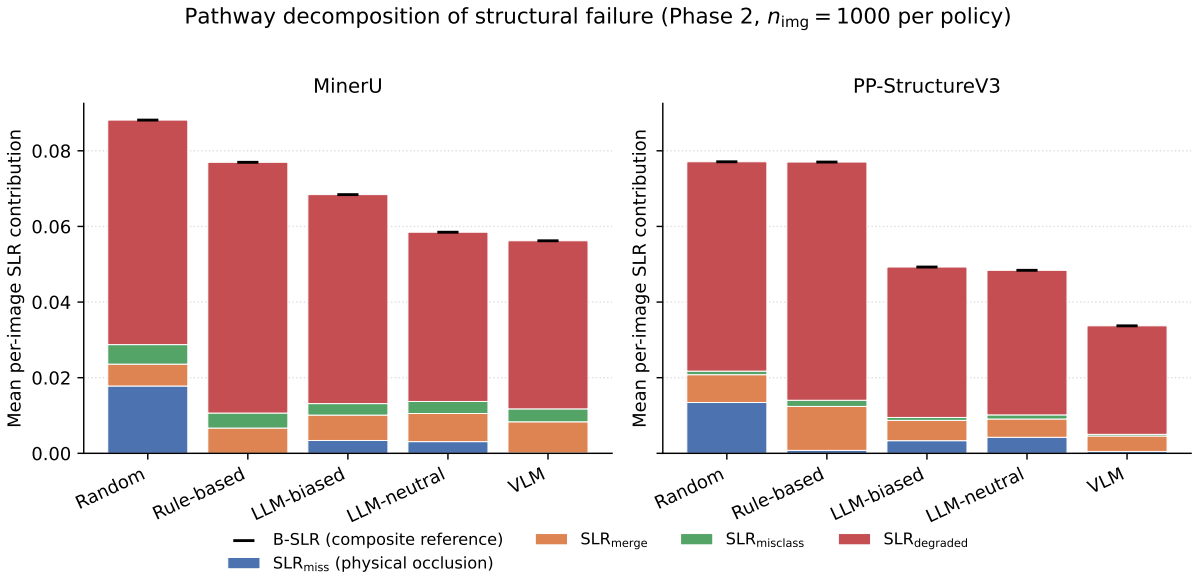

Figure 6: Phase 2 pathway composition of mean per-image structural loss for each policy (n=1,000 pages). Stacks show SLR$_{\text{miss}}$, SLR$_{\text{merge}}$, SLR$_{\text{misclass}}$, and SLR$_{\text{degraded}}$; black ticks mark composite B-SLR, and higher stacks indicate greater structural loss.

and $0.97/0.87$ on PP-StructureV3 for CER/ΔmAP, whereas the corresponding TOR-based correlations drop to $0.40/0.40$ and $0.30/0.40$. This supports the use of B-SLR as a policy-level structural diagnostic: it preserves the relative vulnerability pattern reflected by terminal quality signals better than the area-based footprint proxy.

E Prompts, Reproducibility, and Scope

E.1 Reproducibility Checklist

- **Data**: PubLayNet and DocLayNet validation splits; pages with fewer than five original spans are filtered; the final shared evaluation pool contains 1,000 pages.
- **Pipelines**: MinerU v2.7.6 with DocLayout-YOLO, and PP-StructureV3 with RT-DETR-H and PaddleOCR 3.4.1. Software and hardware details are reported in Appendix C.5.
- **Matching**: maximum-IoU lookup with unified IoU and TextSim gates ($\tau_{\text{iou}} = 0.1$, $\tau_{\text{text}} = 0.5$); no one-to-one Hungarian assignment is enforced.
- **Probes**: P1–P9 with bounded configuration ranges, listed in Table 5.
- **Configurations**: A01–A22 and NT01–NT07 for Phase 1 fixed auditing (Tables 10 and 11); S01–S13 for randomized sweeps (Table 12); five policy families for Phase 2.
- **Statistics**: six-layer verification protocol, including configuration-level OLS, image fixed effects, per-image rank checks, dose-response bins, and within-configuration quartile checks (Table 14).
- **Randomness**: base seed 42 for deterministic components; paired randomized-sweep configurations use deterministic per-image seeds; LLM/VLM policies inherit stochasticity from API sampling.
- **Key output columns**: `image_id`, `config_id`, `TOR`, `ACR`, `BPO`, `BOC`, `EIR`, `B_SLR`, `B_SLR_iou_only`, `SLR_miss`, `SLR_topo`, `CER_matched_mean`, `mAP_clean`, `mAP_adv`, `delta_mAP`, and `n_orig_spans`.

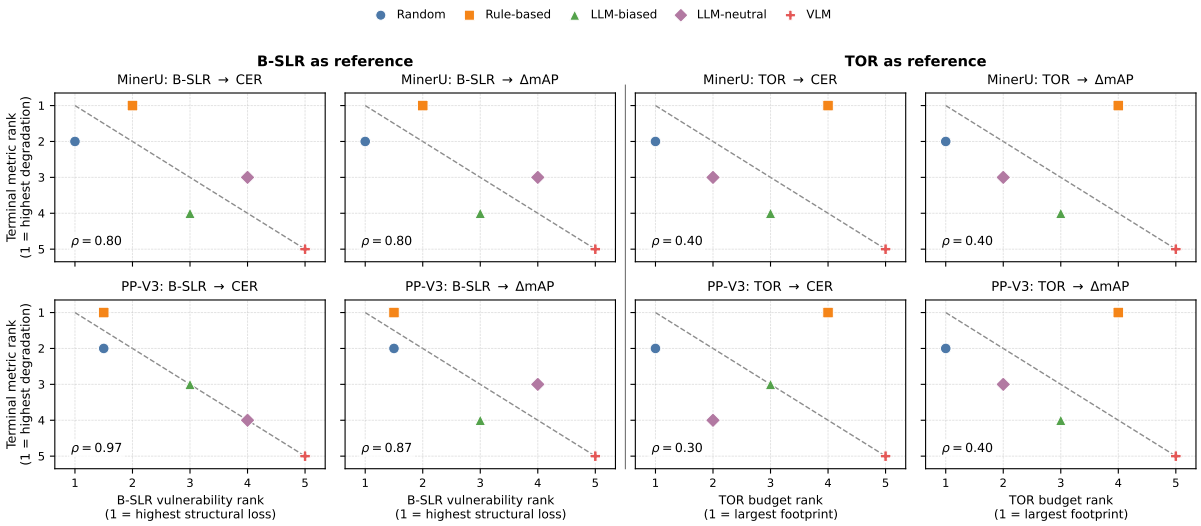

Figure 7: **Policy-level rank consistency.** Ranks are computed within each parser over the five Phase 2 policies, where rank 1 denotes the highest degradation or largest footprint. Compared with TOR, B-SLR gives policy rankings that align more closely with CER and ΔmAP.

- **Metrics**: $\overline{\mathrm{CER}}$ is the primary terminal OCR judge; per-image mAP@0.5 over the five canonical layout classes defines the supplementary detection-channel drop $\Delta\mathrm{mAP}$.

E.2 Prompt Templates

The policy prompts are designed to emit the same output schema: a probe type, bounded probe parameters, and a placement strategy. The biased and neutral LLM variants differ in role framing, objective wording, strategy labels, and context rendering, as summarized in Table 6. The boxed excerpts below summarize the three prompt-based policy variants. For the text-only policies, the biased variant uses a structure-aware page description through `context.to_text_description()`, whereas the neutral variant uses a coordinate-only rendering through `context.to_neutral_description()`. The VLM policy instead receives the raw page image and predicts only the probe type and placement strategy. For the neutral and VLM prompts, strategy names are mapped internally as `between`→`bridge`, `edge`→`anchor`, `inside`→`content`, and `anywhere`→`random`.

LLM-biased prompt excerpt. Role: expert adversarial tester for DLA. Goal: choose one visual probe and placement to maximize parsing failure, including merge, split, or missing-block errors. Probes: P1–P9; strategies: `bridge`/`anchor`/`content`/`random`; output: valid JSON.
Context: structure-aware page description with block coordinates, gap information, and candidate vulnerable regions.

LLM-neutral prompt excerpt. Role: quality-assurance analyst for DLA robustness. Goal: select one realistic perturbation probe and placement strategy to test model response. Strategies: `between`/`edge`/`inside`/`anywhere`; output: valid JSON.
Context: coordinate-only page description with block types and bounding boxes, without explicit structural vulnerability hints.

VLM strategy-only prompt excerpt. Input: raw page image only. Goal: inspect visible text blocks, figures, tables, and whitespace, then select a probe type and placement strategy. Exact coordinates are determined automatically from the chosen strategy; output: valid JSON.

If a returned strategy is unrecognized, the fallback is `random`; if the probe type is invalid, the fallback is `P5`.

E.3 Context Encoder for Non-VLM Policies

For random, rule-based, and LLM-based policies, the context encoder provides three groups of page-level features:

- **Visual statistics**: grayscale moments, edge density, and page size.
- **Layout structure**: block count, category entropy, and area distribution.
- **Spatial relations**: gap density, nearest-neighbor spacing, and multi-column cues.