THINKING WITH ANCHORS: GROUNDED AND EFFICIENT DOCUMENT REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing document understanding benchmarks have largely focused on locating page elements, yet real-world document intelligence requires models to reason jointly about region semantics, spatial relations, and visual structure. We present `ADOPD 2026`, a reasoning-oriented extension of `ADOPD` that turns page decomposition into spatially grounded document understanding. `ADOPD 2026` enriches page anchors inherited from `ADOPD 2024` dataset with human-cleaned captions, semantic tags, and generated chain-of-thought (CoT) traces grounded to document regions. Instead of treating boxes, masks, and tags as independent supervision signals, we cast text blocks, visual entities, semantic labels, bounding boxes, and polygon masks as a shared vocabulary of *visual anchors*. This representation supports three connected capabilities. First, region-level semantic tagging asks models to identify document element types from both page context and local appearance, revealing long-tail semantic failures that standard layout benchmarks often hide. Second, unified vision-language grounding generates text regions and visual entities together with coordinates or polygonal outlines, transforming detection and segmentation outputs into structured anchors that can be reused by downstream reasoning systems. Third, current state-of-the-art models still struggle with dense counting tasks evaluated on `DocCount`, a benchmark derived from `ADOPD 2026`, highlighting the need for the Thinking-with-Anchors pipeline in document semantic understanding. By connecting page decomposition to verifiable visual-anchor reasoning, `ADOPD 2026` provides a task framework that moves document understanding beyond localization toward anchor-grounded document intelligence.

1 INTRODUCTION

Appearing as posters, menus, advertisements, infographics, etc., documents are a primary interface for real-world information exchange. Automating document understanding has motivated research on document analysis and OCR systems (Chaudhuri, 2007; Smith, 2007), as well as real-world applications ranging from information extraction and accessibility to business-process automation, document question answering, and multimodal assistants (Huang et al., 2022; Kim et al., 2022; Mathew et al., 2021; Mathur et al., 2023). Recent datasets and benchmarks have enabled important progress in layout analysis and document understanding (Zhong et al., 2019b; Pfitzmann et al., 2022b; Cheng et al., 2023a), largely by defining page elements that can be detected, segmented, or parsed. However, many downstream uses require more than a list of localized regions: a system may need to associate a price with the correct product block, connect a chart to its legend and caption, distinguish decorative regions from informative content, or justify an answer by pointing to specific visual evidence on the page. In these settings, page elements must function as grounded anchors that carry semantic roles, spatial intent, and meaningful relations to neighboring regions. This perspective exposes the gap between page-decomposition-centered supervision and the evidence-oriented reasoning required by practical document AI.

Current methods only partially fulfill this requirement. Specialized document parsers, detectors, and segmenters can localize text blocks or visual entities, but their outputs are often treated as coarse page-decomposition results rather than as grounded evidence with semantic roles and cross-region spatial relations. Open-vocabulary detectors and unified generative models make grounding more flexible via conditioning on language or emitting boxes and polygons as sequences (Liu et al., 2024;

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.

Chen et al., 2022; Xiao et al., 2024; Peng et al., 2023). However, such practice is vulnerable when applied to dense document pages: when every text block, entity, and polygon vertex is serialized autoregressively, latency and formatting errors increase with the number of input anchors. Recent work on multi-token prediction, multi-head decoding, masked generation, and diffusion language models suggests a complementary direction in which dense outputs can be predicted in larger parallel chunks while preserving language-level control (Gloeckle et al., 2024; Cai et al., 2024; Chang et al., 2022; Nie et al., 2025; Ye et al., 2025). This provides a brand-new angle to treat the visual grounding task as both a fine-grounding and a semantic reasoning problem.

Motivated by the aforementioned gap and new perspectives on visual grounding, we build `ADOPD 2026` to turn dense document annotations into a testbed for grounded document reasoning. Starting from the original `ADOPD 2024` dataset (Gu et al., 2024), which contains 120k visually diverse document images across more than 1000 document types, we enrich the inherited page element annotations with human-cleaned captions, semantic tags, and chain-of-thought (CoT) data. We organize OCR text boxes, human-labeled entity polygons, semantic labels, and polygon outlines around a shared abstraction: the *visual anchor*.

A *visual anchor* is a document region represented not only by its geometry but also by its semantic role, functions, and potential relations to other regions on the page. This abstraction builds on prior work that reasons over visual primitives (Lu et al., 2026). By proposing a visually anchored dataset and an additional anchor-grounded evaluation benchmark, our goal is to study how vision-language models (VLMs) can go beyond detection and segmentation to ground, tag, compose, and verify document anchors as intermediate evidence for reasoning.

This paper develops `ADOPD 2024` from the visual-anchor point of view along the following directions. First, we collect visual anchors over `ADOPD 2026`: human-cleaned captions, semantic tags for entity polygons, and OCR text boxes. We also generated CoT traces grounded to those visual anchors (Sec. 3). Second, we evaluate document localization on collected visual anchors (Sec. 4.1), focusing on conventional detection/segmentation tasks. Third, we evaluate entity-level semantic tagging (Sec. 4.2), where models must use both local appearance and full-page context to assign semantic roles to localized regions. Fourth, we derived `DocCount` from `ADOPD 2026`, a dense document-counting benchmark with Thinking-with-Anchors reasoning traces, and evaluated state-of-the-art VLMs in the zero-shot setting (Sec. 4.3).

In summary, `ADOPD 2026` reframes document page decomposition as grounded reasoning over visual anchors. By connecting semantic region tagging, unified grounding of text and visual entities, and verified grounded narration, it studies how document models can produce structured visual evidence that is spatially precise, semantically meaningful, and usable by downstream reasoning systems. This shifts `ADOPD` from a dataset for decomposing document pages into a benchmark for building, evaluating, and diagnosing grounded document reasoning.

2 BACKGROUND

From pages to evidence. Document understanding has long combined page segmentation, OCR, layout analysis, and task-specific extraction. Early systems used rule-based grouping, projection profiles, connected components, or handcrafted geometry for page segmentation and text-line extraction (Ouwayed & Belaïd, 2012; Lee et al., 2019), while modern datasets such as PubLayNet (Zhong et al., 2019b), DocBank (Li et al., 2020b), DocLayNet (Pfitzmann et al., 2022b), M^{6}Doc (Cheng et al., 2023a), IIIT-AR-13K (Mondal et al., 2020b), and the original `ADOPD 2024` page-decomposition dataset (Gu et al., 2024) support learning-based layout detection and segmentation. Document-language pretraining and OCR-free models further brings together text, layout, images for parsing and extraction, and question answering (Huang et al., 2022; Li et al., 2021; Gu et al., 2021; Tang et al., 2023; Kim et al., 2022; Mathew et al., 2021; Mathur et al., 2023). These works define important tasks, but they often evaluate either low-level decomposition or final answers without specifying the intermediate evidence layer. `ADOPD 2026` addresses this gap by making boxes, polygon masks, and semantic tags explicit visual anchors that can be evaluated as structured evidence for reasoning.

Grounding as an interface. Large vision-language models bring language-level reasoning to images through visual instruction tuning, multimodal chain-of-thought, and grounded generation (Liu et al., 2023; Bai et al., 2023; Zhang et al., 2023; Peng et al., 2023; Chen et al., 2023; You et al., 2024;

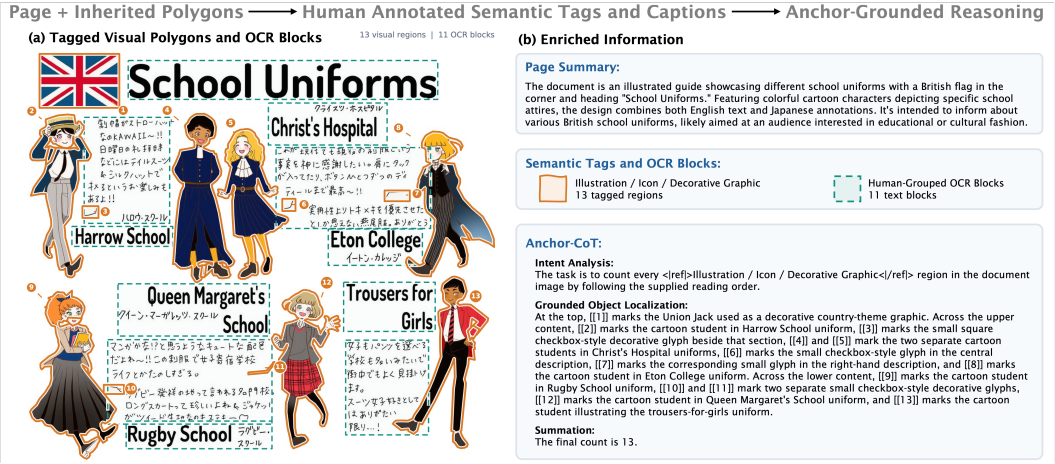


Figure 1: Overview of the ADOPD 2026 data enrichment. Starting from inherited ADOPD 2024 page anchors, the new data round adds human-cleaned captions, semantic tags over regions, and generated CoT traces whose answers are grounded to visual anchors such as boxes and polygons.

Xiao et al., 2024; Lu et al., 2024). Their interface is attractive for documents because region names, coordinates, counts, spatial order, and justifications can all be represented in one symbolic output space. Recent work has also used visual counting as a diagnostic of grounding and reasoning in MLLMs. HoloCount (Deng et al., 2026a) introduces semantic counting, analytical counting, and robustness testing tasks. It reveals substantial degradation from basic perception to analytical and robustness settings, underscoring that fluent multimodal responses do not guarantee numerically reliable visual grounding. HoloCount primarily works in the natural image domain and evaluates the final count using exact-match accuracy. ADOPD 2026 is complementary in the document domain and exposes the intermediate boxes, masks, and semantic tags that support such outputs and can be evaluated independently. Work on thinking with those visual anchors makes this idea more explicit by inserting spatial markers into reasoning, reducing the ambiguity of purely textual references. We derived a dense document counting benchmark, DocCount, from the ADOPD 2026 dataset. DocCount inherits the benefit of rich anchors and executable checks that tie reasoning back to verifiable visual evidence.

Composable perception and reasoning. Agentic visual systems connect perception modules with language-level decisions by calling tools, writing programs, or reasoning over marked inputs. VISPROG (Gupta & Kembhavi, 2023) and ViperGPT (Surís et al., 2023) compose vision modules through executable programs, MM-ReAct (Yang et al., 2023b) routes a language model through vision experts, and Set-of-Mark prompting (Yang et al., 2023a) lets a multimodal model refer to numbered visual evidence without directly regressing coordinates. These systems show that visual reasoning can be separated from low-level perception, but they are usually evaluated on natural images or general multimodal tasks with fewer regions and weaker requirements on calibrated grounding. In one downstream experiment, we proposed an agentic self-refinement workflow that studies agentic grouping over detector candidates, and our grounded verifier checks whether the resulting reasoning remains spatially precise and semantically accountable.

3 Data Construction

The original ADOPD 2024 corpus (Gu et al., 2024) contains dense page geometry for ∼120k visually diverse documents, including human-drawn entity polygons and OCR text blocks. ADOPD 2026 builds on top of these inherited anchors by adding three additional layers: human-cleaned captions, fine-grained semantic tags, and generated CoT traces grounded to visual anchors (Figure 1).

3.1 Entity-Level Semantic Tagging

Human-drawn entity polygons and OCR text-block locations from the original ADOPD 2024 dataset gives each page a dense geometric decomposition. These anchors are useful for document understand-

ing, but they are not sufficient for high-level reasoning: a model may know where a region is without knowing whether it functions as a photograph, a brand logo, a background image, a decorative motif, or a text role such as a title or caption. Clean page-level captions are also critical. In ADOPD 2026, we therefore first rewrite the original captions with human annotation, asking annotators to summarize the document type, visual style, salient content, and high-level intent so that the global description is better aligned with human interpretation.

However, global caption alone remains too coarse for fine-grained reasoning over page elements. We introduce entity-level semantic tagging on top of the visual anchors. The annotation unit is an existing entity polygon or OCR text block, and annotators assign a semantic role only after reading and understanding the full document context. For visual masks, this includes judging whether the region belongs to the foreground or background and then assigning a document-specific tag such as photograph, illustration, brand logo, background image, chart, icon, or color block. For OCR blocks, the tag captures text function, such as title, body text, header, footer, or image caption. Each tagged anchor thus becomes a reusable, semantically meaningful evidence unit of the form *geometry + modality + semantic role*.

This setting differs from conventional tagging tasks where the label space is usually clearer in advance, such as NLP sequence tagging and named-entity recognition (Tjong Kim Sang & De Meulder, 2003), or object-centric vision taxonomies in PASCAL VOC and MS COCO (Everingham et al., 2010; Lin et al., 2014). Document elements are shaped by layout, design intent, reading order, and page context, so their semantic roles cannot be reduced to a fixed object list. We therefore define the 30-class tagging taxonomy through a human-in-the-loop process: candidate tags are proposed from the data, annotators apply them to real pages, and ambiguous boundaries are refined through review. Despite such effort, document tagging is inevitably long-tailed and subjective. Annotators may differ in design knowledge, visual judgment, or interpretation of tags. Thus, we further apply disambiguation rules and post-processing checks. The complete taxonomy, foreground/background layer definition, disambiguation rules, worked examples, and resulting long-tail tag distribution are provided in Appendix A.5. The comparison between ADOPD 2026 and other existing document datasets is shown in Appendix Table 4.

3.2 Thinking with Anchors

The final enrichment layer turns the tagged visual anchors of Sec. 3.1 into grounded, step-by-step reasoning traces, so that each page is paired not only with *where* and *what* its regions are but also with *how* to reason over them. We instantiate this layer on dense counting, a task that remains challenging for multimodal large language models precisely because it requires grounding and semantic understanding at once rather than a single free-form answer. Each example starts from a document page plus polygon masks with human-annotated semantic classes, such as Photograph, Table, Brand Logo, etc. Following the Thinking-with-Visual-Primitives paradigm (Lu et al., 2026), we construct counting questions whose answers require grounding related regions on the document, rather than relying solely on free-form caption knowledge. Class labels are further selected during data preparation. A VLM verifier is applied to validate the labels of the polygon masks, followed by a manual human check. The CoT generator is prompted to write the natural-language reasoning for each grounded anchors in the samples, providing us with a CoT-style supervision that is linguistically natural but geometrically controlled as well. This procedure yields DocCount, a document-counting benchmark on which we evaluate 13 state-of-the-art VLMs in the zero-shot setting (Sec. 4.3). Appendix Table 5 shows the comparison between DocCount and existing counting-related benchmarks.

4 Experiments

This section is organized around three experimental threads, based on the same finalized 80k/20k/20k train/validation/test split. The split construction and balance diagnostics are detailed in Appendix A.2. First, we evaluate document decomposition and detection, comparing specialized non-VLM detectors/segmenters with the VLM-based LocateAnything Wang et al. (2026) grounder and an agentic grouping refinement. Second, we study entity-level semantic tagging, in which the goal is to assign taxonomy labels defined in Sec. 3 to localized document regions. Third, we evaluate VLMs' zero-shot performance on DocCount, our document-counting benchmark.

Table 1: Main results for Doc2Box detection and Doc2Mask segmentation tasks.

		Zero-shot (ZS)			Fine-tuned (FT)						
Model	#Par.	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	mF1
Box detection											
YOLOv12-M	20.2 M	0.1	0.3	0.1	69.0	82.1	75.0	30.0	55.2	75.1	77.0
YOLOv12-X	59.1 M	0.1	0.3	0.1	69.1	82.0	75.1	29.7	55.1	75.2	77.0
RF-DETR-Nano	30.5 M	0.0	0.1	0.0	50.2	72.3	53.9	9.8	31.6	58.8	69.7
RF-DETR-Large	33.9 M	0.1	0.3	0.2	63.9	81.3	69.9	22.8	48.1	71.4	75.5
LocateAnything-3B	3.83 B	–	–	–	40.2	56.8	43.4	9.3	26.3	47.8	59.4
Mask segmentation											
YOLOv12-Seg-M	∼23.6 M	1.9	3.7	1.6	62.7	78.9	74.4	29.0	57.3	74.2	76.2
YOLOv12-Seg-X	∼62.8 M	2.2	4.0	2.0	64.4	80.5	76.7	30.0	59.5	76.8	77.1
RF-DETR-Seg-Nano	33.6 M	1.3	2.8	0.9	52.9	71.0	58.4	3.6	26.3	63.6	70.2
RF-DETR-Seg-2XL	38.6 M	1.3	2.6	0.9	69.5	81.9	75.8	20.9	51.6	77.6	76.8
SAM3	848 M	2.3	5.0	1.7	57.2	77.7	58.0	27.7	45.4	62.0	75.5
SAM3.1	848 M	2.5	5.4	2.0	59.4	78.7	60.6	27.2	45.5	64.7	76.0
LocateAnything-3B-DP	3.83 B	–	–	–	32.5	47.4	34.0	5.7	20.3	38.0	57.1

4.1 Visual Anchor Localization

4.1.1 Experimental Setup

Tasks. We evaluate two tag-agnostic page localization tasks based on OCR bounding boxes (Doc2Box) and visual entity masks (Doc2Mask). In Doc2Box detection, every text box region is treated as a single class, *entity*. In Doc2Mask segmentation, every foreground visual entity mask is also treated as an *entity*. Although these localization tasks do not require the model to distinguish semantic region tags, they are not intended to make the task trivial. The overall goal is to measure whether a model can recover the reusable visual anchors that downstream semantic and reasoning tasks depend on, such as OCR boxes and visual entity masks. In documents, failures such as missing text blocks, merged visual regions, or over-split entities can break overall document understanding. Following `ADOPD 2024` (Gu et al., 2024), we decouple localization from tagging: Doc2Box detection and Doc2Mask segmentation identify where visual anchors are and whether they are decomposed at the right level of granularity, while Doc2Tag task evaluates which semantic labels should be assigned to the localized regions. This separation allows us to diagnose geometric decomposition errors separately from semantic tagging errors.

Modeling. We evaluate six model families on these two tag-agnostic page localization tasks. The specialized detection and segmentation model families are fine-tuned on the 80k train split. The same models are also evaluated zero-shot when an off-the-shelf checkpoint is available for comparison. The non-VLM baselines include YOLOv12 for text box detection and YOLOv12-Seg for mask segmentation (Tian et al., 2025); RF-DETR for text box detection and RF-DETR-Seg for mask segmentation (Carion et al., 2020; Robinson et al., 2026); and promptable segmentation foundation models SAM3/SAM3.1 (Kirillov et al., 2023; Ravi et al., 2024; Carion et al., 2025). We also evaluate LocateAnything-3B, a VLM-based generative grounder that emits boxes and polygonal masks (Wang et al., 2026). For detailed training and inference setup, please refer to A.3.

Evaluation metrics. For the localization task, the standard evaluation metrics are COCO-style Average Precision (AP), which averages AP over ten IoU thresholds, and Average Recall (AR) computed on the single *entity* class. In Doc2Box detection, we evaluate box AP/AR against annotated OCR blocks. In Doc2Mask segmentation, we evaluate mask AP/AR against foreground entity masks. In addition, we also report AP_{50} (IoU threshold 0.5), AP_{75} (IoU threshold 0.75), AP_S (regions smaller than 32^2 pixel^2), AP_M (regions between 32^2 to 96^2 pixel^2), AP_L (regions greater than 96^2 pixel^2). These metrics reward both recall and spatial precision, penalizing the decomposition errors that this experiment thread is designed to expose, such as missing anchors, over-merged regions, over-split boxes, and loose masks. Unlike other non-VLM detectors and segmentors, LocateAnything will not output a deterministic confidence score by default. We thus use a pseudo-confidence score simulated from the generation positions of the emitted boxes. Such a pseudo-confidence score is used only for the AP-style metrics, which require detections to be ranked. Although we tried our best to adopt a pseudo-confidence score to align with the requirements of a confidence-score-based metric, this metric may not fairly evaluate confidence-free models such as LocateAnything, especially in cross-

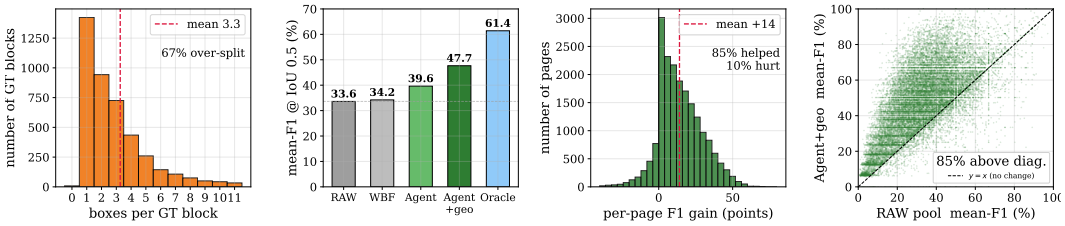
model comparisons. We also include mF1 as an auxiliary diagnostic metric. LocateAnything follows the confidence-free F1@IoU Mean used for generative box outputs (Wang et al., 2026), whereas non-VLM detectors and segmenters use confidence-swept IoU$=0.5$ F1. Given such model-specific calculation, mF1 should be interpreted within its stated definition rather than as a single, universal ranking score for cross-model comparison.

4.1.2 Analysis

From Table 1, we can see that `ADOPD 2026` creates a genuine transfer gap between the pretraining image domain and the document domain for segmentation and detection models. Zero-shot evaluations are near failure, while fine-tuning the same architectures lifts AP by 50–70 points. This indicates that natural-image category priors and promptable segmentation priors in existing models' training procedure do not directly transfer to OCR boxes (Doc2Box) or visual entity masks (Doc2Mask) localization task. LocateAnything-3B-DP's hybrid decoding achieves a 2.2x speedup for text-box detection while retaining 99% of autoregressive box AP. However, it retains only 26% of autoregressive mask AP for polygon segmentation, highlighting a task-dependent speed–quality tradeoff (Appendix Table 11).

The main box-detection error stems from over-fragmentation. Each ground truth text box block is covered by 3.3 predicted detector boxes on average, and 66% receive at least two predicted boxes (Fig. 2a). A GT-guided perfect merge can lift mean-F1 by a large margin (Fig. 2b). We therefore treat the detected box pool as proposals and ask a reasoning VLM to group the numbered boxes only through an agentic workflow. Appendix Fig. 8 shows the agentic workflow.

This agentic refinement stage requires no additional training and does not use ground-truth annotations when deciding which boxes to merge. RF-DETR and YOLOv12 are run with a low confidence threshold of (0.05) to retain a high-recall collection of candidate text boxes. Overlapping predictions from the two detectors are combined using Weighted Box Fusion (WBF), remaining duplicates are removed by Non-Maximum Suppression (NMS) at IoU (0.6). The remaining boxes are drawn on the original page and assigned integer identifiers. Keye-VL-2.0-30B Team et al. (2026) is asked to examine the numbered candidate boxes and, based on the page's visual layout and text organization, explain which boxes likely belong to the same coherent text block (Fig. 8 **Reason over marks**). The explanation is then included in a second model request, prompting the model to complete only a final list of box-identifier groups (Fig. 8 **Parse groups**). For each proposed group, we compute the smallest enclosing rectangle of its member boxes rather than directly predicting coordinates. A deterministic geometric check rejects groups whose enclosing rectangle contains excessive empty space, covers most of the page, or contains more than 20 boxes. When a group is rejected, its original boxes are retained separately instead of being merged (Fig. 8 **Post-check**). The full algorithm and numeric table are in Appendix A.7 and Appendix Table 15.



(a) Boxes per GT block. (b) mean-F1 by method. (c) Per-page gain. (d) RAW vs. Agent+geo.

Figure 2: Agentic refinement over the 20k-pages validation split for RF-DETR-Large proposed text boxes pool. Panel (a) shows why the agentic workflow is needed: detector boxes per ground-truth block have a long over-split tail. Panel (b) separates the sources of improvement: WBF alone barely changes mean-F1, semantic grouping by agent gives the first lift, and the agent with geometric over-merge guard further improves the mean-F1. Panels (c) and (d) show that the improvement is broad across the validation split rather than driven by a few pages.

The agentic workflow recovers a large fraction of ground truth text boxes without any additional training. On the same fixed pool of predicted text boxes (RF-DETR-Large, denoted as RAW), geometry-based fusion (WBF) increases mean-F1 by 0.6, agent grouping raises the performance by

6.0 (33.6 to 39.6), and the agentic workflow with geometric over-merge guard reaches 47.7, closing 49% of the 61.4 merge-oracle ceiling (Fig. 2b). A worked example of grouping and geometric validation is shown in Appendix Fig. 10.

From Fig. 2d, 85% of per-page points lie above the diagonal of the RAW- vs. Agent+geo- graph, which means that 85% of pages achieve improvements. We use mean-F1@IoU0.5 for this experiment because the target output is one clean box per text block region without a direct confidence score, and AP can decrease even when the final decomposition is more useful for downstream document processing.

4.2 Entity-Level Semantic Tagging

After the document regions are localized, entity-level tagging asks the model to decide what semantic role the localized region plays in the document. This isolates semantic interpretation from geometric decomposition. We evaluate this setting with a 12-class region taxonomy derived from the annotation scheme in Sec. 3.1, comparing three zero-shot open VLM families (Qwen2.5-VL, InterVL3, Gemma4-12B) against a fine-tuned LocateAnything-3B tagger.

Evaluation Metrics The evaluation is controlled around the same localized region. Each open VLM sees the full page with a red-box marker, pixel coordinates, and the candidate 12-class label list, while LocateAnything uses its native `<ref>·</ref><box>·</box>` input format. We report two region presentations: a *single-image* view, where the model sees the marked full page, and a *dual-image* view, where the marked page is paired with a crop around the target region. All rows in Table 2 are class accuracy scored on the same class-balanced validation set (30 regions per class, n=360, so micro and macro accuracy coincide).

Table 2: Region content-type tagging on the 12-class taxonomy: zero-shot open VLMs vs. the fine-tuned LocateAnything-3B tagger.

	Zero-shot						Fine-tuned	
	Qwen2.5-VL		InternVL3		Gemma4-12B		LocateAnything	
Class	Single	Dual	Single	Dual	Single	Dual	Single	Dual
Text Block / Content	73.3	73.3	56.7	66.7	50.0	46.7	86.7	53.3
Photograph	40.0	20.0	56.7	60.0	76.7	83.3	86.7	40.0
Line / Divider	16.7	16.7	13.3	40.0	46.7	60.0	76.7	90.0
Icon	30.0	46.7	33.3	43.3	33.3	36.7	83.3	63.3
Brand Logo	76.7	70.0	63.3	66.7	80.0	90.0	46.7	40.0
Table	70.0	86.7	80.0	83.3	76.7	83.3	46.7	66.7
Chart / Graph	53.3	56.7	56.7	60.0	63.3	66.7	66.7	70.0
Color Block	10.0	10.0	10.0	10.0	16.7	20.0	43.3	86.7
Dialog Box	36.7	30.0	13.3	10.0	26.7	36.7	40.0	60.0
Illustration / Artwork	13.3	20.0	46.7	30.0	50.0	56.7	50.0	63.3
Decorative / Pattern Graphic	6.7	6.7	0.0	3.3	3.3	3.3	13.3	26.7
Background Image	3.3	0.0	3.3	0.0	3.3	0.0	30.0	43.3
Avg	35.8	36.4	36.1	39.4	43.9	48.6	55.8	58.6

4.2.1 Analysis

From the results in Table. 2, we can see that VLMs overuse top classes while almost never recovering the rare classes in the long tail of the label distribution. Fine-tuning is helpful in recovering the rare classes in the long-tail distribution, as we could see from LocateAnything's performance on Background Image, Color Block, and Dialog Box.

The single-image and dual-image views expose a second dimension of difficulty. Cropping is consistently helpful for zero-shot models, but the gain is modest, which suggests that the primary bottleneck is not simply seeing the region at higher resolution. For the fine-tuned tagger, cropping helps label decisions based on local visual form, such as Color Block, Table, Dialog Box, and Line / Divider, but it hurts label accuracy for page-level contexts, especially Photograph, Text Block, and Icon. This is probably because the functionality of regions in a document may depend on the region's own semantics, as well as on the role of the whole document. A single crop cannot preserve page-level signal.

The long-tail distribution of classes continues to influence tagging accuracy. Under-exposed classes such as Photograph, Table, and Dialog Box remain below where their visual separability suggests they should be. We conducted experiments with resampling that oversamples rare classes by $\min(\sqrt{N_{\max}/N_c},\, 8)$, giving the tail more training exposure. Results confirm the frequency diagnosis by lifting Photograph ($30 \rightarrow 57$), Table ($63 \rightarrow 81$), Chart ($74 \rightarrow 88$), and Dialog Box ($56 \rightarrow 69$), but it sacrifices some top-class accuracies at the same time (Appendix Table. 12). Another diagnostic shows that decoding choices can also affect rare-class recall without retraining (Appendix Table 13).

4.3 Anchor Thinking

To construct `DocCount`, a benchmark that evaluates models' counting ability in a dense setting, we build a pipeline using the `ADOPD 2026` dataset that contains 12 class labels. The data preparation pipeline starts with selecting target labels, simplifying human-labeled polygons, and converting the target regions into both a grounded Anchor-CoT generation and a counting benchmark.

Table 3: Reasoning-on exact-count accuracy on the `DocCount` benchmark (400+ clean-label documents), with per-class accuracies. Total accuracy is shown in the rightmost column. Underline marks the second-best result in each column.

Model	Brand Logo	Photograph	Table	Chart / Graph	Acc.
Open-source models					
Kimi-K2.5 (1T)	63.80	87.78	58.67	66.67	72.85
Qwen3.5 (397B-A17B)	54.60	<u>87.22</u>	<u>65.33</u>	75.00	<u>70.81</u>
Qwen3.6 (35B-A3B)	55.83	85.00	56.00	75.00	68.78
GLM-4.6V (106B)	50.92	84.44	46.67	62.50	64.48
Gemma-4 (31B)	55.21	72.78	41.33	<u>70.83</u>	60.86
MiMo-VL (7B)	46.63	75.56	48.00	58.33	59.28
InternVL3.5 (30B-A3B)	41.10	61.11	30.67	50.00	47.96
Cosmos3-Super (64B)	28.22	52.22	16.00	58.33	37.56
Proprietary models					
Claude-Sonnet-4.5	<u>56.44</u>	81.67	54.67	54.17	66.29
GPT-5.2	46.63	82.78	66.67	75.00	66.29
Gemini-2.5-Pro	50.92	80.56	42.67	66.67	62.44

4.3.1 Construction of Evaluation Dataset

Target Vocabulary To more directly assess MLLMs' dense counting capability, we construct the evaluation set using 4 target labels from the 12-class taxonomy: `Brand Logo`, `Photograph`, `Table`, `Chart/Graph`. We select this subset for its especially clear and visually separable class definitions, thereby reducing potential ambiguity in category interpretation during evaluation. A.9 attaches the complete class definitions of the 4 target labels.

Polygon Simplification Human-annotated object masks may contain duplicate polygon vertices or redundantly close vertices. To construct a clean polygon mask input for `Anchor-CoT` generation, the polygon mask is simplified by deleting consecutive duplicates and collinear vertices, removing the "least important" vertex via the Visvalingam-Whyatt algorithm, and selecting the candidate with the fewest vertices that still significantly overlaps the original polygons. If no candidate passes, the cleaned original is kept. This helped us reduce around 23.49% percent of total vertices in polygon masks.

Strong-VLM Polygon-Label Verification After polygon simplification, we input the polygon masks for the top-1 class (by count) from each image into a VLM verifier (GPT-5.5) for class label verification. To build a dense counting dataset, only documents in which the top-1 class contains more than 3 objects are retained. The verifier prompt contains the original image, an overlay image with indexed polygons, and a prompt asking whether each visible polygon belongs to the expected class. The verifier vocabulary includes the four target classes, plus a `Other/None of the Above` tag. The verifier only checks whether selected polygons are mislabeled, given the class definition. It does not check whether the entire document contains additional unmarked polygons of the same class. Missing instances are handled by the following manual review process.

Manual Missing-Label Review Document candidates that pass the verifier's check are manually reviewed for missing top-1-class instances and documents with missing labels are excluded from the

evaluation dataset. During the same pass, the human reviewer also sanity-checks the visible labeled polygons: if a marked polygon clearly does not belong to the target class, the image is excluded together with missing-label cases. This pipeline finally produces a conservative but reliable evaluation set for `DocCount`.

`Anchor-CoT` Input Construction The 400+ documents are packaged for `Anchor-CoT` generation. Only the verified top-1 class per document is used. For each sample, the CoT generator (GPT 5.5) receives the image, the target class definition, and a list of simplified target polygons as placeholder assignments: `[[1]]` $= [[x, y], \ldots]$, `[[2]]` $= [[x, y], \ldots]$, $\ldots$, the CoT generator is prompted to write natural first-person reasoning with what target label is required to be counted in the document, what each polygon instance is about, and what the final count is to the question. This design preserves the original separation between geometry and language. The CoT generator controls the prose, but the count, instance input order, and polygon coordinates remain deterministic functions of the verified clean labels. Examples are presented in A.10.

4.3.2 Evaluation

The 400+ documents now serve as `DocCount`, a benchmark dataset for dense counting evaluation of current state-of-the-art VLMs. For evaluation experiments in this paper, target polygon masks, polygon overlays, and target object lists are not included in the evaluation prompt. Each model receives only the original image and a question asking how many regions of the target class appear on the document page. The default prompt also includes the target class definition and counting instructions (see A.9) and asks the model to return an integer count inside an answer tag.

The benchmark uses ten different question phrasings to avoid giving every sample the identical wording. The official metric is exact integer accuracy over all samples.

Overall Model Ranking Table 3 reports the updated model counting results. Kimi-K2.5 is the best performer, followed by Qwen3.6-35B-A3B. At the same time, the task is not saturated: the best model still misses around 25% of the samples.

Analysis The counting evaluation shows that current state-of-the-art VLMs still struggle with exact visual enumeration in document pages. The best condition reaches 72.85% accuracy, but the task remains far from solved (selected cases in A.11). The class analysis suggests two kinds of failure. `DocCount` evaluates VLM's capability from both perception- and policy-angle: VLMs must interpret definitions consistently aligned with the annotation policy and understand the whole images in order to give out the final correct answer.

5 Conclusion

This paper documented the `ADOPD 2026` pipeline. First, we described the **data enrichment** protocol that adds human-cleaned captions and closed-vocabulary tags on top of `ADOPD 2024`'s inherited entity polygons and OCR text blocks (Sec. 3). Second, we built a model-guided 80k/20k/20k split whose validation and test sets are representative, mutually matched, and moderately challenging, selected from multiple candidates by an ensemble model-challenge calibration (Appendix A.2). Third, our **document decomposition and localization** study showed that YOLOv12, RF-DETR, SAM3/SAM3.1, and the VLM-based LocateAnything grounder all face a substantial domain gap on `ADOPD 2026`, while an agentic grouping refinement over predicted text boxes can recover part of the over-fragmentation error without additional training (Sec. 4.1). Fourth, our **entity-level tagging** study showed that zero-shot VLMs struggle with document-specific region semantics, while fine-tuning a LocateAnything-style tagger recovers part of the class label in the long tail distribution (Sec. 4.2). Finally, our **`DocCount` benchmarking** also reveals a current challenge for state-of-the-art VLMs in performing the counting task with definition-following, semantic understanding, and dense grounding on documents (Sec. 4.3).

Together, `ADOPD 2026` opens multiple directions for future work: improving small-entity detection under dense layouts, bridging the domain gap for CJK documents, scaling the grounded-narration corpus to the full reasoning-question taxonomy, and training MLLMs that leverage grounded visual anchors to reason about real-world document pages.

AI USE STATEMENT

As described in Sections 3.2 and 4.3, we used GPT-5.5 to verify candidate polygon labels and generate grounded natural-language reasoning traces for `DocCount`. We then manually reviewed the labels and constrained generated counts, instance ordering, and polygon coordinates to the verified annotations. Appendix A.2 and Section 4.1.2 document the model-assisted split construction and agentic grouping procedures, respectively. AI assistance was also used to polish the writing of the paper.

REPRODUCIBILITY STATEMENT

The data enrichment and evaluation protocols are described in Sections 3–4. Appendix A.2 details the split construction and balance diagnostics; Appendix A.3 gives the localization training and inference setup; Appendix A.5 specifies the annotation taxonomy and rules; and Section 4.3.1 documents the counting-data construction. The agentic refinement algorithm and its numerical results are provided in Appendices A.7 and A.8.

REFERENCES

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.

Tianle Cai, Yuhong Li, Zhengyang Geng, Hongwu Peng, Jason D. Lee, Deming Chen, and Tri Dao. Medusa: Simple LLM inference acceleration framework with multiple decoding heads. *arXiv preprint arXiv:2401.10774*, 2024.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision (ECCV)*, pp. 213–229. Springer, 2020.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. SAM 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. MaskGIT: Masked generative image transformer. In *CVPR*, 2022.

Bidyut B. Chaudhuri (ed.). *Digital Document Processing: Major Directions and Recent Advances*. Springer London, 2007. doi: 10.1007/978-1-84628-726-8.

Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal LLM's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023.

Ting Chen, Saurabh Saxena, Lala Li, David J. Fleet, and Geoffrey Hinton. Pix2seq: A language modeling framework for object detection. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=e42KbIw6Wb.

Yufan Chen, Ruiping Liu, Junwei Zheng, Di Wen, Kunyu Peng, Jiaming Zhang, and Rainer Stiefelhagen. Graph-based document structure analysis. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/cf3d7d8e79703fe947deffb587a83639-Abstract-Conference.html.

Hiuyi Cheng, Peirong Zhang, Sihang Wu, Jiaxin Zhang, Qiyuan Zhu, Zecheng Xie, Jing Li, Kai Ding, and Lianwen Jin. M6doc: A large-scale multi-format, multi-type, multi-layout, multi-language, multi-annotation category dataset for modern document layout analysis. In *CVPR*, 2023a.

Hiuyi Cheng, Peirong Zhang, Sihang Wu, Jiaxin Zhang, Qiyuan Zhu, Zecheng Xie, Jing Li, Kai Ding, and Lianwen Jin. M^6Doc: A large-scale multi-format, multi-type, multi-layout, multi-language, multi-annotation category dataset for modern document layout analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15138–15147, June 2023b. URL https://openaccess.thecvf.com/content/CVPR2023/html/Cheng_M6Doc_A_Large-Scale_Multi-Format_Multi-Type_Multi-Layout_Multi-Language_Multi-Annotation_Category_Dataset_CVPR_2023_paper.html.

Thibault Clérice, Juliette Janès, Hugo Scheithauer, Sarah Bénière, Florian Cafiero, Laurent Romary, Simon Gabay, and Benoît Sagot. Diachronic document dataset for semantic layout analysis. *arXiv preprint arXiv:2411.10068*, 2024. doi: 10.48550/arXiv.2411.10068. URL https://arxiv.org/abs/2411.10068.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 91–104. IEEE, 2025.

Jinhong Deng, Limeng Qiao, and Guanglu Wan. Holocount: A holistic visual counting benchmark for MLLMs. *arXiv preprint arXiv:2607.06420*, 2026a. URL https://arxiv.org/abs/2607.06420.

Jinhong Deng, Limeng Qiao, and Guanglu Wan. Holocount: A holistic visual counting benchmark for mllms. *arXiv preprint arXiv:2607.06420*, 2026b.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88 (2):303–338, 2010.

Fabian Gloeckle, Badr Youbi Idrissi, Baptiste Rozière, David Lopez-Paz, and Gabriel Synnaeve. Better & faster large language models via multi-token prediction. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 15706–15734. PMLR, 2024. URL https://proceedings.mlr.press/v235/gloeckle24a.html.

Jiuxiang Gu, Jason Kuen, Vlad I. Morariu, Handong Zhao, Rajiv Jain, Nikolaos Barmpalios, Ani Nenkova, and Tong Sun. UniDoc: Unified pretraining framework for document understanding. In *NeurIPS*, pp. 39–50, 2021.

Jiuxiang Gu, Xiangxi Shi, Jason Kuen, Lu Qi, Ruiyi Zhang, Anqi Liu, Ani Nenkova, and Tong Sun. ADoPD: A large-scale document page decomposition dataset. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=x1ptaXpOYa.

Tanmay Gupta and Aniruddha Kembhavi. Visual programming: Compositional visual reasoning without training. In *CVPR*, 2023.

Ahmed Heakl, Youssef Mohamed, Abdullah Sohail, Rania Elbadry, Ahmed Nassar, Peter W. J. Staar, Fahad Shahbaz Khan, Imran Razzak, and Salman Khan. DocAtlas: Multilingual document understanding across 80+ languages. *arXiv preprint arXiv:2605.12623*, 2026. doi: 10.48550/arXiv.2605.12623. URL https://arxiv.org/abs/2605.12623.

Meng-Ru Hsieh, Yen-Liang Lin, and Winston H Hsu. Drone-based object counting by spatially regularized regional proposal network. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 4165–4173. IEEE, 2017.

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *ACMMM*, 2022.

Haroon Idrees, Muhmmad Tayyab, Kishan Athrey, Dong Zhang, Somaya Al-Maadeed, Nasir Rajpoot, and Mubarak Shah. Composition loss for counting, density map estimation and localization in dense crowds. In *European Conference on Computer Vision*, pp. 544–559. Springer, 2018.

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 (ECCV)*, pp. 498–517, 2022. doi: 10.1007/978-3-031-19815-1_29.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4015–4026, 2023.

Joonho Lee, Hideaki Hayashi, Wataru Ohyama, and Seiichi Uchida. Page segmentation using a convolutional neural network with trainable co-occurrence features. In *International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1023–1028, 2019. doi: 10.1109/ICDAR.2019.00167.

Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. DocBank: A benchmark dataset for document layout analysis. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 949–960, Barcelona, Spain (Online), December 2020a. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.82. URL https://aclanthology.org/2020.coling-main.82/.

Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. DocBank: A benchmark dataset for document layout analysis. In *Proceedings of the 28th International Conference on Computational Linguistics (COLING)*, pp. 949–960, 2020b. doi: 10.18653/v1/2020.coling-main.82.

Peizhao Li, Jiuxiang Gu, Jason Kuen, Vlad I. Morariu, Handong Zhao, Rajiv Jain, Varun Manjunatha, and Hongfu Liu. Selfdoc: Self-supervised document representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5652–5660, 2021.

Zhang Li, Yuliang Liu, Qiang Liu, Zhiyin Ma, Ziyang Zhang, Shuo Zhang, Zidun Guo, Jiarui Zhang, Xinyu Wang, and Xiang Bai. MonkeyOCR: Document parsing with a structure-recognition-relation triplet paradigm. *arXiv preprint arXiv:2506.05218*, 2025. doi: 10.48550/arXiv.2506.05218. URL https://arxiv.org/abs/2506.05218.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pp. 740–755. Springer, 2014. doi: 10.1007/978-3-319-10602-1_48. URL https://doi.org/10.1007/978-3-319-10602-1_48.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection. In *ECCV*, 2024.

Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savya Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. Unified-IO 2: Scaling autoregressive multimodal models with vision, language, audio, and action. In *CVPR*, 2024.

Ruijie Lu, Yiyang Ma, Xiaokang Chen, Lingxiao Luo, Zhiyu Wu, Zizheng Pan, Xingchao Liu, Yutong Lin, Hao Li, Wen Liu, Zhewen Hao, Xi Gao, Shaoheng Nie, Yixuan Wei, Zhenda Xie, Ting Chen, and Gang Zeng. Thinking with visual primitives. Technical report, DeepSeek-AI, 2026.

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

Puneet Mathur, Rajiv Jain, Jiuxiang Gu, Franck Dernoncourt, Dinesh Manocha, and Vlad I. Morariu. Docedit: Language-guided document editing. In *AAAI*, pp. 1914–1922, 2023. doi: 10.1609/aaai.v37i2.25282.

Ajoy Mondal, Peter Lipps, and C. V. Jawahar. IIIT-AR-13K: A new dataset for graphical object detection in documents. In *Document Analysis Systems*, volume 12116 of *Lecture Notes in Computer Science*, pp. 216–230, Cham, 2020a. Springer International Publishing. doi: 10.1007/978-3-030-57058-3_16. URL https://doi.org/10.1007/978-3-030-57058-3_16.

Ajoy Mondal, Peter Lipps, and CV Jawahar. Iiit-ar-13k: a new dataset for graphical object detection in documents. In *DAS*, 2020b.

Oikantik Nath, Sahithi Kukkala, Mitesh Khapra, and Ravi Kiran Sarvadevabhatla. IndicDLP: A foundational dataset for multi-lingual and multi-domain document layout parsing. In *Document Analysis and Recognition – ICDAR 2025*, volume 16023 of *Lecture Notes in Computer Science*, pp. 23–39, Cham, 2026. Springer Nature Switzerland. ISBN 978-3-032-04614-7. doi: 10.1007/978-3-032-04614-7_2. URL https://doi.org/10.1007/978-3-032-04614-7_2.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *arXiv preprint arXiv:2502.09992*, 2025.

Nazih Ouwayed and Abdel Belaïd. A general approach for multi-oriented text line extraction of handwritten documents. *IJDAR*, 2012.

Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. Teaching clip to count to ten. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3147–3157. IEEE, 2023.

Alejandro Peña, Aythami Morales, Julian Fierrez, Javier Ortega-Garcia, Iñigo Puente, Jorge Cordova, and Gonzalo Cordova. Continuous document layout analysis: Human-in-the-loop AI-based data curation, database, and evaluation in the domain of public affairs. *Information Fusion*, 108:102398, 2024. doi: 10.1016/j.inffus.2024.102398. URL https://doi.org/10.1016/j.inffus.2024.102398.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023.

Birgit Pfitzmann, Christoph Auer, Michele Dolfi, Ahmed S. Nassar, and Peter W. J. Staar. DocLayNet: A large human-annotated dataset for document-layout segmentation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3743–3751, New York, NY, USA, 2022a. Association for Computing Machinery. doi: 10.1145/3534678.3539043. URL https://doi.org/10.1145/3534678.3539043.

Birgit Pfitzmann, Christoph Auer, Michele Dolfi, Ahmed S. Nassar, and Peter W. J. Staar. DocLayNet: A large human-annotated dataset for document-layout segmentation. In *SIGKDD*, pp. 3743–3751, 2022b. doi: 10.1145/3534678.3539043.

Viresh Ranjan, Udbhav Sharma, Thu Nguyen, and Minh Hoai. Learning to count everything. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3394–3403, June 2021.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

Isaac Robinson, Peter Robicheaux, Matvei Popov, Deva Ramanan, and Neehar Peri. RF-DETR: Neural architecture search for real-time detection transformers. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=qHm5GePxTh.

Vishwanath A Sindagi, Rajeev Yasarla, and Vishal M Patel. Jhu-crowd++: Large-scale crowd counting dataset and a benchmark method. *IEEE transactions on pattern analysis and machine intelligence*, 44(5):2594–2609, 2020.

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

Dídac Surís, Sachit Menon, and Carl Vondrick. ViperGPT: Visual inference via python execution for reasoning. In *ICCV*, 2023.

Jayant Sravan Tamarapalli, Rynaa Grover, Nilay Pande, and Sahiti Yerramilli. Countqa: How well do mllms count in the wild? *arXiv preprint arXiv:2508.06585*, 2025.

Zineng Tang, Ziyi Yang, Guoxin Wang, Yuwei Fang, Yang Liu, Chenguang Zhu, Michael Zeng, Cha Zhang, and Mohit Bansal. Unifying vision, text, and layout for universal document processing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19254–19264, 2023.

Kwai Keye Team, Bin Wen, Changyi Liu, Chengru Song, Chongling Rao, Guowang Zhang, Han Li, Haonan Fan, Hengrui Ju, Jiankang Chen, et al. Kwai keye-vl-2.0 technical report. *arXiv preprint arXiv:2606.10651*, 2026.

Yunjie Tian, Qixiang Ye, and David Doermann. YOLOv12: Attention-centric real-time object detectors. *arXiv preprint arXiv:2502.12524*, 2025.

Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL*, pp. 142–147, 2003.

Nobuhiro Ueda, Yuyang Dong, Krisztián Boros, Daiki Ito, Takuya Sera, and Masafumi Oyamada. SCAN: Semantic document layout analysis for textual and visual retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 1618–1637, Rabat, Morocco, March 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-eacl.82. URL https://aclanthology.org/2026.findings-eacl.82/.

Qi Wang, Junyu Gao, Wei Lin, and Xuelong Li. Nwpu-crowd: A large-scale benchmark for crowd counting and localization. *IEEE transactions on pattern analysis and machine intelligence*, 43(6): 2141–2149, 2020.

Shihao Wang, Shilong Liu, Yuanguo Kuang, Xinyu Wei, Yangzhou Liu, Zhiqi Li, Yunze Man, Guo Chen, Andrew Tao, Guilin Liu, Jan Kautz, Lei Zhang, and Zhiding Yu. Locateanything: Fast and high-quality vision-language grounding with parallel box decoding. *arXiv preprint arXiv:2605.27365*, 2026.

Renqiu Xia, Song Mao, Xiangchao Yan, Hongbin Zhou, Bo Zhang, Haoyang Peng, Jiahao Pi, Daocheng Fu, Wenjie Wu, Hancheng Ye, Shiyang Feng, Bin Wang, Chao Xu, Conghui He, Pinlong Cai, Min Dou, Botian Shi, Sheng Zhou, Yongwei Wang, Bin Wang, Junchi Yan, Fei Wu, and Yu Qiao. DocGenome: An open large-scale scientific document benchmark for training and testing multi-modal large language models. *arXiv preprint arXiv:2406.11633*, 2024. doi: 10.48550/arXiv.2406.11633. URL https://arxiv.org/abs/2406.11633.

Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *CVPR*, 2024.

Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in GPT-4V. *arXiv preprint arXiv:2310.11441*, 2023a.

Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. MM-REACT: Prompting chatgpt for multimodal reasoning and action. *arXiv preprint arXiv:2303.11381*, 2023b.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models. *arXiv preprint arXiv:2508.15487*, 2025.

Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. In *ICLR*, 2024.

Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi-column convolutional neural network. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 589–597, 2016. doi: 10.1109/CVPR.2016.70.

Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. Multimodal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*, 2023.

Zhiyuan Zhao, Hengrui Kang, Bin Wang, and Conghui He. DocLayout-YOLO: Enhancing document layout analysis through diverse synthetic data and global-to-local adaptive perception. *arXiv preprint arXiv:2410.12628*, 2024. doi: 10.48550/arXiv.2410.12628. URL https://arxiv.org/abs/2410.12628.

Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. PubLayNet: Largest dataset ever for document layout analysis. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1015–1022. IEEE, 2019a. doi: 10.1109/ICDAR.2019.00166. URL https://doi.org/10.1109/ICDAR.2019.00166.

Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. PubLayNet: Largest dataset ever for document layout analysis. In *International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1015–1022, 2019b. doi: 10.1109/ICDAR.2019.00166.

A Appendix

A.1 Benchmark Comparisons

Table 4: Comparison of document-layout datasets. **Size** is the complete dataset size (training, validation, and test combined), reported in the dataset's native unit: pages, images, documents (docs), or annotated block instances (inst.). **Provenance** describes how the annotations were produced: **H** denotes manual annotation, **HV** human verification or correction, **A** automatic or weak-supervision annotation, and **S** synthetic generation. **Box** denotes native axis-aligned boxes, rotated boxes, or polygons with at most four vertices. **Arbitrary polygon** denotes native per-instance boundaries with more than four vertices. Converting a box into a four-point polygon does not count. **Region-linked OCR** denotes transcription explicitly associated with a localized region instance. **Semantic class** denotes a semantic category assigned to a localized region. **Grounded spatial tasks** denotes released annotations requiring localization or spatial/relational reasoning over identified regions. Coordinates alone do not count. ✓ indicates explicit native support, △ indicates subset-only, indirect, inherited, or structurally related support, and — indicates that the feature is absent.

Dataset	Size	Provenance	Box	Arbitrary polygon	Region-linked OCR	Semantic class	Grounded spatial tasks
PubLayNet Zhong et al. (2019a)	358,353 pages	A	✓	△	—	✓	—
DocBank Li et al. (2020a)	500K pages	A	✓	—	✓	✓	—
IIIT-AR-13K Mondal et al. (2020a)	13K pages	H	✓	—	—	✓	—
DocLayNet Pfitzmann et al. (2022a)	80,863 pages	H	✓	—	△	✓	—
M^{6}Doc Cheng et al. (2023b)	9,080 pages	H	✓	✓	—	✓	—
DocGenome Xia et al. (2024)	500K docs	A+HV	✓	—	✓	✓	✓
PALdb Peña et al. (2024)	441K pages	A+HV	✓	—	✓	✓	—
Diachronic Doc Clérice et al. (2024)	7,254 pages	H	✓	—	—	✓	—
DocSynth-300K Zhao et al. (2024)	300K images	S	✓	—	—	✓	—
DocStructBench Zhao et al. (2024)	11,314 images	H	✓	—	—	✓	—
GraphDoc Chen et al. (2025)	80K images	H+A	✓	—	✓	✓	✓
IndicDLP Nath et al. (2026)	119,806 images	H+HV	✓	—	—	✓	—
MonkeyDoc Li et al. (2025)	3.9M inst.	H+A+S	✓	—	✓	✓	△
SCAN Ueda et al. (2026)	24,577 pages	H	✓	—	—	✓	—
DocAtlas Heakl et al. (2026)	365,862 pages	A+S	✓	—	✓	✓	△
`ADOPD 2026` (ours)	120K pages	H+HV	✓	✓	✓	✓	✓

Table 5: Comparison of evaluation protocols in representative visual-counting benchmarks. **# Samples** reports the official test split in the datasets, or the full released evaluation collection for evaluation-only benchmarks. **Counted Object** means the visual target whose instance count contributes to the final answer, before any requested arithmetic calculation that may appear in the query. **Granularity** means whether the counted targets are semantically defined or only labeled with their class name. **Grounding** means instance-level spatial ground truth annotations available for objects that contribute to the final answer, including points, boxes, or polygons. **Grounded-reasoning** means whether an explicit reasoning representation exists for every instance-level spatial target in the evaluation collection.

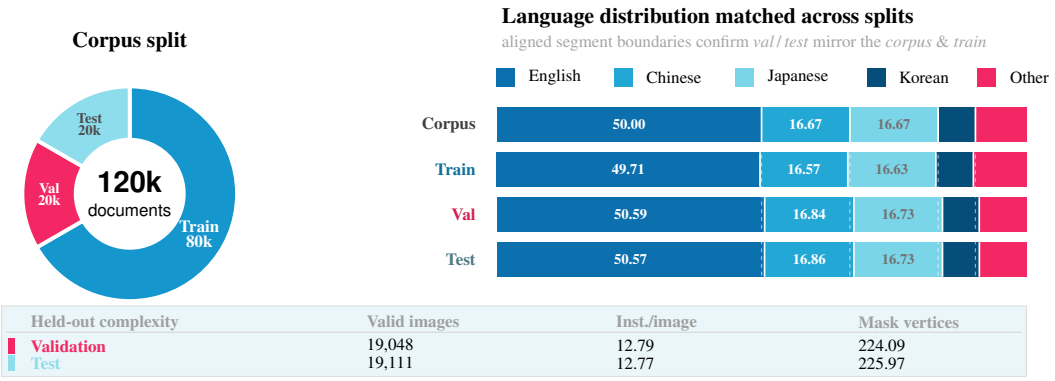
Benchmark	Image Domain	# Samples	Counted Object	Granularity	Grounding	Grounded-reasoning
ShanghaiTech (Zhang et al., 2016)	Natural scenes	498	Person	Fixed class	Points	No
NWPU-Crowd (Wang et al., 2020)	Natural scenes	1,500	Person	Fixed class	Points, Boxes	No
JHU-CROWD++ (Sindagi et al., 2020)	Natural crowd scenes	1,600	Person	Fixed class	Points	No
CARPK (Hsieh et al., 2017)	Aerial parking lots	459	Car	Fixed class	Boxes	No
UCF-QNRF (Idrees et al., 2018)	Dense natural scenes	334	Person	Fixed class	Points	No
FSC-147 (Ranjan et al., 2021)	Natural images	1,190	Target objects[1]	Exampler boxes[1]	Points	No
CountBench (Paiss et al., 2023)	General images	540	Target labels	Class name	None	No
PixMo-Count (Deitke et al., 2025)	General images	540	Target labels	Class name	None	No
CountQA (Tamarapalli et al., 2025)	Natural scenes	1,528	Target labels	Class name	None	No
HoloCount (Deng et al., 2026b)	General images[2]	2,480	Target labels	Class name[3]	None	No
`DocCount` (ours)	Document pages	442	Semantic document region	Semantically defined classes	Polygon-grounded	Yes[4]

Notes. [1]The model is given three examples with bounding boxes and is asked to count all objects visually similar to those examples. [2] HoloCount also includes adversarial synthetic images. [3] Depending on the task query, HoloCount may include attribute constraints, regions of interest (ROIs), coordinates, exclusions, set operations, and categorical name in the input prompt. [4]`DocCount` provides grounded-CoT, namely, `Anchor-CoT`, together with the final scalar count.

A.2 Model-Guided Data Split Construction

Motivation. The released `ADOPD 2026` corpus contains approximately 120k densely annotated samples. A random split is undesirable because the dataset is intentionally long-tailed: taxonomy labels, languages, layout styles, text density, bounding box density, and segmentation-mask complexity are all correlated. We therefore construct an 80k/20k/20k train/validation/test split with two goals. First, the validation and test sets should be representative of the full corpus and closely matched to each other. Second, they should contain sufficiently challenging samples for modern detection and segmentation systems, while avoiding an adversarially hard subset on which models would receive little learning signal.

Taxonomy-balanced sampling. We start from the sample-level metadata and human annotations. The fine-grained taxonomy strings are normalized and semantically merged into 100 coarse taxonomy groups. All validation and test sampling is then performed around these 100 groups rather than around raw folder names or annotation counts alone. Within each taxonomy group, we further stratify by language, box/mask complexity, and document profile. The primary balancing fields are: (i) taxonomy cluster, (ii) language group, (iii) segmentation/bounding-box complexity tier, (iv) text-rich versus visual-rich profile, (v) dominant mask label family, and (vi) layout aspect-ratio profile. This procedure ensures that the validation and test splits cover the same semantic regions of the corpus while still preserving rare document styles.



Held-out complexity	Valid images	Inst./image	Mask vertices
Validation	19,048	12.79	224.09
Test	19,111	12.77	225.97

Figure 3: The ADOPD 2026 corpus is partitioned into an 80k/20k/20k train/validation/test split. The left donut shows split size; the stacked bars give the language composition (English/Chinese/Japanese/Korean/other) of the full corpus and of each split, where the aligned segment boundaries show that the validation and test language distributions mirror the corpus and train; and the bottom strip reports the matched annotation complexity (valid images, instances per image, mask vertices) of the held-out sets.

Annotation-complexity features. For each sample, we compute annotation statistics from the human entity masks and text boxes. These include the number of valid mask instances, the number of polygons, polygon vertices, mask-area coverage, box counts, box-area coverage, and the balance between text-heavy and visual-heavy entities. We group samples into five segmentation/box complexity tiers. Validation and test are drawn so that every taxonomy cluster contains a mixture of low-, medium-, and high-complexity examples. Thus, a category is not represented only by simple pages; for example, a financial-document group contains both sparse price cards and dense table-like layouts.

Candidate split generation. After constructing the initial balanced split, we generate five candidate validation/test partitions from the same 40k held-out pool. The training set remains fixed at 80k samples. The candidates are: **(1)** the current taxonomy/language/complexity-balanced split, **(2)** a taxonomy-language-tier alternating split that pairs hard and easy samples within each bucket, **(3)** a taxonomy-language-tier split matched by model-challenge deciles, **(4)** a complexity-profile-matched split focused on text/visual and mask-family balance, and **(5)** a greedy minimax split that jointly minimizes taxonomy, language, complexity, and model-challenge gaps. We explicitly exclude an extreme-hard stress-test split from the final candidates because our goal is to evaluate progress on learnable document decomposition rather than to create an adversarial benchmark.

Model-challenge calibration. We evaluate the held-out 40k pool with multiple third-party codebases and multiple inference settings. For each model m and image i, we compute box recall and mask recall at IoU 0.5 over ground-truth instances and define

$$c_i^m = 1 - \frac{1}{2}\left(R_{i,0.5}^{m,\text{box}} + R_{i,0.5}^{m,\text{mask}}\right). \tag{1}$$

The ensemble challenge score is the average of c_i^m over all completed model/parameter runs. We use this score only as a calibration signal: a good validation/test split should be moderately challenging, representative, and well matched across val and test. It should not simply maximize failure rate.

Selecting the final validation and test sets. For each candidate split S, we recompute COCO metrics by filtering the full-pool predictions to that candidate's validation and test IDs. We then rank candidates using a composite score:

$$\begin{aligned}\mathcal{L}(S) = {} & 6\Delta_{\text{mAP}}(S) + 3\Delta_{\text{challenge}}(S) \\ & + 2\Delta_{\text{feature}}(S) + \Delta_{\text{representative}}(S) \\ & + 2\Delta_{\text{base}}(S) - 0.25\overline{\text{AP}}(S).\end{aligned} \tag{2}$$

Here Δ_{mAP} is the average val/test gap in box and mask AP over all model settings, $\Delta_{\text{challenge}}$ is the val/test gap in ensemble challenge score, Δ_{feature} is the mean distribution gap over taxonomy, language, complexity, and layout fields, and $\Delta_{\text{representative}}$ measures how close val and test are to

Table 6: Planned entity-level model suite for model-challenge calibration. All configurations use a single class, *entity*; taxonomy and language labels are used only for split balancing and subgroup analysis.

Task	Codebase	Version / Variant	Size(s)	Checkpoint / Source	Primary Metrics
Detection	RF-DETR	Detection	Nano, 2XL	local / downloadable	AP_{box}, AP^{box}_{50}, AP^{box}_{75}
	YOLOv12	Turbo detection	M, X	downloadable	AP_{box}, AP^{box}_{50}, AP^{box}_{75}
	GroundingDINO	v1.0 SwinT	T	local	AP_{box}, prompt sensitivity
	GroundingDINO	v1.5 / DINO-X	API / large	API checkpoint	AP_{box}, prompt sensitivity
	SAM3	text-conditioned detector	default	HF checkpoint	AP_{box}, recall@IoU
	SAM3.1	text-conditioned detector	default	HF checkpoint	AP_{box}, recall@IoU
Segmentation	RF-DETR	Segmentation	Nano, 2XL	local	AP_{mask}, AP^{mask}_{50}, AP^{mask}_{75}
	YOLOv12	Segmentation release	M, X	local / downloadable	AP_{mask}, AP_{box}
	Grounded-SAM2	GDINO v1.0 + SAM2.1	Tiny, Large	local	AP_{mask}, AP_{box}
	Grounded-SAM2	DINO-X + SAM2.1	Large	API + local SAM2.1	AP_{mask}, prompt sensitivity
	SAM3	text-conditioned segmentation	default	HF checkpoint	AP_{mask}, recall@IoU
	SAM3.1	text-conditioned segmentation	default	HF checkpoint	AP_{mask}, recall@IoU

Notes. Prediction target: every valid human entity is mapped to category id 1. Detection and segmentation are evaluated as separate tracks. Confidence, box-threshold, and text-threshold sweeps are tuned on validation and frozen for test. During this calibration phase, foundation-model rows are evaluated zero-shot to rank candidate splits; in the final baseline study (Sec. 4) the trainable architectures are fine-tuned and additionally reported zero-shot for reference. All results are stratified by taxonomy, language, text richness, and visual complexity.

Table 7: Final candidate ranking after recomputing COCO mAP for each candidate validation and test set. The mAP gap, challenge gap, feature gap, representativeness gap, and mean AP columns are reported in percentage points; lower score is better.

Candidate	Score	mAP Gap	Chal. Gap	Feat. Gap	Rep. Gap	Mean AP
Greedy minimax all signals	0.0115	0.20	0.05	0.16	0.08	2.45
Taxonomy-language challenge decile	0.0454	0.29	0.12	1.17	0.59	2.36
Taxonomy-language hard alternating	0.0485	0.24	0.23	1.18	0.59	2.37
Complexity-profile matched	0.0671	0.41	0.16	1.68	0.84	2.31
Initial balanced split	0.0733	0.33	0.72	1.01	0.51	2.44

the full held-out pool. The small AP reward prevents selecting a pathological ultra-hard split when two candidates have otherwise similar balance. The lowest-scoring candidate is used as the final 20k validation and 20k test split.

Selected split statistics. The selected split is the greedy minimax all-signal candidate. It contains 80,000 training samples, 20,000 validation samples, and 20,000 test samples, with files stored as candidate 05 in the model-challenge split root. After removing samples without valid entity masks under the COCO conversion rules, the full-pool model calibration evaluates 38,159 images: 19,048 from validation and 19,111 from test, with 249,113 ground-truth entity-mask instances. Duplicate sample leakage between train, validation, and test is zero after sample-ID and metadata-path checks. The selected validation/test split is also closely matched in annotation complexity: validation and test contain 12.79 and 12.77 boxes or polygons per image on average, 224.09 and 225.97 polygon vertices, and nearly identical text-rich and visual-rich scores.

A.3 Training and Inference Setup for Visual Anchor Localization

All supervised baselines are trained on the 80k train split, and model checkpoints used for evaluation are selected based on validation performance. YOLOv12 and RF-DETR are fine-tuned for 10 epochs with an input-resolution sweep (384/640/768/1024) and task-dependent global batches (YOLOv12 96–256; RF-DETR 32–48). SAM3/SAM3.1 use the same COCO-style data pipeline and are evaluated as mask predictors. Because their presence head becomes unstable in a naive long run with focal-loss γ=0, we use a numerically stable loss path. SAM3/SAM3.1 compute the ($\gamma = 0$) presence-head focal loss as alpha-weighted binary cross-entropy loss instead of the unstable Triton kernel in focal-loss backward computation, preventing NaN gradients without changing the objective. After this correction, training is stable for the full 10 epochs, with mask AP peaking around epoch 3 and remaining within $\sim$1 point through epoch 10. For LocateAnything, we convert each page into a unified grounding target: text blocks contribute boxes, and visual entities contribute paired boxes and adaptive Douglas–Peucker polygons, simplified to at most 16 vertices. We fine-tune LocateAnything

end-to-end at long edge 1024 with multi-token-prediction supervision for 50k steps (≈5 epochs), and report the best validation checkpoint (epoch 3) in Table 1. The zero-shot columns evaluate off-the-shelf checkpoints without ADOPD 2026 fine-tuning, mapping model outputs to the single *entity* class when needed.

A.3.1 FINE-TUNING SAM3/SAM3.1

As we mentioned in the paragraph above, SAM3/SAM3.1 require a numerically stable loss path, and training converges quickly under such a loss function. The additional training epochs did not degrade the model's performance, as shown in Table 8, which provides subset AP measured on a fixed 1,240-image validation subset. The full-split values use the same 20k validation split reported in the main results.

Table 8: SAM3/SAM3.1 fine-tuning schedule on ADOPD 2026 (validation mask AP). After the γ=0 focal-loss fix, AP peaks at epoch 3 and stays on a narrow plateau through epoch 10. The full 20k-split AP is measured at the epoch-3 peak and epoch-10 end (and reported in Table 1). The per-epoch trajectory ep1→ep10 is traced on a fixed 1,240-image subset, where the same peak-then-plateau shape is evident.

	Subset trajectory (1.24k imgs)				Full 20k
Model	ep1	ep3	ep5	ep10	ep3 / ep10
SAM3	45.1	48.1	47.7	47.6	57.2 / 56.3
SAM3.1	43.8	49.3	48.9	49.2	59.4 / 58.5

We use a small, fixed evaluation subset solely to illustrate the trajectory's shape. Subset AP is systematically lower than full-split AP. Only the *shape* (which peaks at ep3 and plateaus thereafter) is the claim. On the full split, epoch 10 remains within ∼1 AP of the epoch-3 peak for both models.

A.3.2 LOCATEANYTHING UNIFIED-GROUNDING DETAILS

Tables 9–11 provide the training, evaluation, and decoding details. The unified grounding target is learnable for LocateAnything, while its lower detector-style AP mainly reflects confidence ranking, tokenized geometry, polygon-mask limitations, and decoding brittleness rather than failed document-domain adaptation.

Table 9: LocateAnything unified-grounding training summary. Loss is cross-entropy; the selected checkpoint is step 30k (epoch 3).

Model	Entity targets	First loss	Loss @ 30k
Unified (box + polygon)	<box> + <quad>	4.18	0.33

Table 10: LocateAnything-3B-DP unified-grounding evaluation on the ADOPD 2026 validation split (%).

Output	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	mF1
Box (<box>)	35.9	49.4	38.0	7.2	22.9	41.2	57.1
Mask (<quad>)	32.5	47.4	34.0	5.7	20.3	38.0	–

AP uses pseudo-confidence by generation order; mF1 is F1@IoU Mean on boxes. The mask row is the LocateAnything-3B-DP entry reported in Table 1.

Table 11: LocateAnything-3B-DP decoding speed/quality trade-off on ADOPD 2026. Speed is the mean over the image (median in parentheses); AP is the COCO AP on matched 1k-image validation subsets.

Mode	s/image	Speedup	AP	AP_{50}	AP retain
Detection (text block, box-only; six-token box frame)					
slow (AR)	3.94 (3.25)	1.0×	41.1	57.5	100%
hybrid (MTP+AR)	1.81 (1.25)	2.2×	40.8	57.0	99%
fast (MTP)	0.97 (0.92)	4.1×	32.9	47.0	80%
Segmentation (entity, box+polygon; 13+ tokens/region)					
slow (AR)	7.40 (4.00)	1.0×	23.1	36.2	100%
hybrid (MTP+AR)	1.85 (1.30)	4.0×	5.9	9.4	26%
fast (MTP)	1.72 (1.02)	4.3×	3.1	5.6	13%

A.4 Semantic Tagging Detailed Tables

Tables 12–13 provide the supporting values for the semantic-tagging analysis in Sec. 4.2.

Table 12: Class accuracy for the fine-tuned tagger on the class-balanced dual validation set.

	Baseline	Resampling
Text Block / Content	53.8	60.2
Photograph	30.4	56.6
Line / Divider	87.8	85.6
Icon	67.0	73.0
Brand Logo	44.4	49.6
Table	62.8	81.0
Chart / Graph	74.0	87.8
Color Block	89.2	88.4
Dialog Box	56.0	68.8
Background Image	55.4	55.0
Illustration / Artwork	65.4	63.6
Decorative / Pattern Graphic	12.6	24.0

We also run an auxiliary VLM-tagger diagnostic on the original 30-class taxonomy to separate representation errors from decoding errors. A Qwen3.5-9B-VL tagger trained on a 15-label set achieves about 60–61% accuracy, but its greedy decoder never emits the rare Visual Motif/Pattern label. Rather than allowing unconstrained generation, we then compute the teacher-forced likelihood for each candidate label and select the highest-scoring label as the closed-set prediction. The teacher-forced way means that, when scoring a candidate label, the evaluator supplies that label's preceding tokens to the model rather than letting the model generate its own preceding tokens. When candidate labels are scored using such teacher-forced likelihood rather than free generation, length-normalizing the score recovers Visual Motif recall from 0 to 23% with no retraining, and a mild logit adjustment reaches 43% at the cost of top-class accuracy (Appendix Table 13).

Table 13: Closed-set decoding diagnostic for Qwen3.5-9B-VL tagger on 15 merged labels.

Decoding policy	Visual Motif recall	Overall micro
free-gen / summed log-prob	0/30 (0%)	61%
mean (length-normalized)	7/30 (23%)	62%
mean + logit-adjust τ=0.5	13/30 (43%)	53%
mean + logit-adjust τ=1.0	10/30 (33%)	52%

A.5 ADOPD 2026 Annotation Guidelines

This section documents the annotation protocol used for the ADOPD 2026 data collection round, which focuses on visually rich documents such as posters, advertisements, magazine pages, and infographics. The primary task was to supplement existing pre-annotations by (i) correcting OCR boxes whose scope was too large, and (ii) assigning a semantic tag from the 30-class taxonomy (Table 14) to every entity mask and every OCR text block.

Table 14: ADOPD 2026 entity tag taxonomy. All 30 tags form a closed vocabulary; each annotated element receives exactly one tag.

Mode	Group	Tags
Mask	Visual (17)	background image, prominent pattern, natural background, document content, table, illustration, brand logo, photograph, color block (borderless), icon, line element, chart, color block (bordered), background pattern, dialog box (bordered), decorative pattern, dialog box (borderless)
OCR	Text (13)	body text, title, list item, legend, table title, footer, note, header, emphasized text, chart title, hyperlink, citation, image caption

A.5.1 Annotation Output and Tag Distribution

In total the corpus contains ∼2.5M tagged elements, split almost evenly between the two annotation modes (∼1.22M visual entity polygons and ∼1.25M text blocks). At the region level, the resulting supervision remains strongly long-tailed: frequent anchors such as text/content blocks, photographs, lines, illustrations, and icons dominate, while backgrounds, dialog boxes, charts, tables, and decorative elements form the tail. This long-tailed structure is important for ADOPD 2026 because reasoning traces often depend on rare but semantically decisive anchors rather than only on the most frequent page regions.

A.5.2 Layer Concept: Foreground and Background

Every annotated element is conceptually assigned to one of two layers:

Background. Elements at the very bottom layer that span the full page width or height (full bleed, left–right bleed, or top–bottom bleed). Background elements support the foreground but do not directly attract attention. Typical examples: background image, background pattern, natural background.

Foreground. All other elements above the background. These are the primary visual or textual elements the viewer notices first: titles, body text, charts, icons, logos, color blocks, dialog boxes, *etc*

A.5.3 OCR Text Label Rules

OCR boxes receive a tag from the **text group** (13 tags). Key disambiguation rules:

- **Title** – Main and sub-titles; typically larger or bolder font. If a pre-annotated box groups title and sub-title together, label it *title*.
- **Body text** – Main content paragraphs and bullet text. If a box mixes title and body text, label it *body text*.
- **List item** – Elements in a numbered or bulleted list where items are visually cohesive. Split list items from body text/title if they appear in the same pre-annotated box. A single-item list still requires its own box with the *list item* tag.
- **Note** – Supplementary comments or footnotes, typically at the end of an article or at the bottom of the document. In newspapers/magazines, side columns adjacent to articles are also notes.
- **Header / Footer** – Content at the very top or bottom of the page that is visually distinct from body text and unrelated to the main content (e.g., page numbers, chapter names, copyright lines).
- **Legend** – Small, inconspicuous text adjacent to a figure or chart that provides explanation.
- **Emphasized text** – Bold, italic, or underlined text within a paragraph. Annotate a separate box only if the segment stands alone; emphasized text embedded in a large body-text block is part of that block.
- **Hyperlink** – Underlined text linking to another resource. Annotate separately only if the hyperlink stands alone or has a visible prefix; hyperlinks embedded inside a large body-text block are part of body text.
- **Table title / Chart title / Image caption** – Titles or captions placed above or below the corresponding table, chart, or image.

- **Brand logo (text)** – Text portion of a logo, including purely text-based logos and the text component of text+graphic logos; select *brand logo* in OCR mode.
- **Citation** – Text excerpted from another source, typically styled differently from body text.

A.5.4 MASK VISUAL LABEL RULES

Mask contours receive a tag from the **visual group** (17 tags). Key rules:

- **Photograph** – Realistic images captured by a camera. Annotate only the overall boundary; do not annotate internal text or sub-objects. If a photograph is used purely as a background with text layered on top, label it as the appropriate background type and annotate the overlaid elements separately.
- **Illustration** – Drawings, paintings, or stylized/conceptual images. Cartoons that do not resemble icons are illustrations; cartoon-style symbolic images are icons.
- **Icon** – Symbols with specific semantic meaning (weather symbols, restroom signs, toolbar buttons, decorative stars/arrows/snowflakes, app logos, QR codes, barcodes). Icons are generally not the bottommost layer.
- **Chart** – Data visualizations (bar, pie, line, scatter). Annotate the entire chart boundary.
- **Table** – Table grid structures. Do not annotate any content inside the table.
- **Brand logo (graphic)** – The graphic component of a logo, including fully graphic logos and the graphic portion of text+graphic logos; select *brand logo* in mask mode.
- **Line element** – Thin lines or curves used as dividers or decorations. A line-like shape that is visually thick should be labeled as a color block instead.
- **Color block (bordered) / Color block (borderless)** – Solid or semi-transparent colored areas, with or without a visible bounding edge. Each contiguous block is a separate instance.
- **Dialog box (bordered) / Dialog box (borderless)** – Speech-bubble shapes with or without a tail.
- **Background image** – A full-bleed or near-full-bleed non-photographic image at the bottom layer.
- **Natural background** – A photographic image that simultaneously functions as the full-bleed page background.
- **Background pattern** – A non-photographic decorative texture or pattern at the background layer (grids, dot matrices, shadow shapes). Shadows count as foreground background patterns.
- **Prominent pattern** – A single large, visually striking, semantically meaningful graphic that is clearly not an icon (e.g., a large standalone silhouette, a hero illustration).
- **Decorative pattern** – Minor ornamental graphics that are neither icons nor prominent patterns; typically few in number and purely decorative.
- **Document content** – An embedded document image within the main document. Do not annotate any internal content of the embedded document.

A.5.5 DETAILED ANNOTATION RULES

1. **Icon.** Icons include: (a) symbols with specific semantic meaning (weather, restroom, toolbar buttons); (b) small decorative markers (snowflakes, arrows, stars); (c) app logos; (d) QR/barcodes.

2. **Icons and illustrations co-occurring.** Elements inside a highlighted box may be icons or illustrations — inspect each individually.

3. **Prominent pattern vs. icon.** A single large, visually striking standalone graphic = *prominent pattern*. A small semantically specific symbol = *icon*.

4. **Icon vs. color block.** Thin arrow-shaped strokes = *icon*. Visually thick bar-shaped elements = *color block*.

5. **Natural background.** If an image is simultaneously a natural photograph *and* the full-bleed page background, label it *natural background*.

6. **Titles in newspapers and magazines.** Mastheads, article headlines, and section names = *title*. Image labels beside photos = *image caption*.

7. **Header / Footer.** Content at the very top or bottom of the page clearly unrelated to the main content (page numbers, chapter names, copyright) = *header* / *footer*.

8. **Background pattern, borderless color block, hyperlink.** Grid texture inside an element = *background pattern*. Plain colored region without border = *color block (borderless)*. Underlined link text = *hyperlink*.

9. **Logo with text and graphic combined.** Annotate the graphic part as *brand logo* in mask mode; annotate the text part as *brand logo* in OCR mode. Do not split the tag across modes.

10. **Mixed pre-annotated OCR boxes.** Resolve conflicts: body text + title together → *body text*; main + sub-title together → *title*; list item + body text → split into two boxes. A single-item list must still be separated.

11. **Line element.** A thin arrow whose arrowhead is small and inconspicuous = *line element*, not icon.

12. **Background image vs. background pattern vs. prominent pattern.** Full-bleed white base = *background image*. Shadow overlay above it = *background pattern* (foreground). Large graphic poster on top = *prominent pattern* (foreground).

13. **Prominent pattern; newspaper notes and footer.** A standalone eye-catching graphic = *prominent pattern*. In newspaper layouts, side-column text adjacent to the main article = *note*; text at the very bottom = *footer*.

14. **Emphasized text and hyperlinks inside body text.** Annotate separately if they stand alone. Hyperlinks or emphasis embedded in a large body-text block without their own pre-annotation belong to the body-text box.

15. **Note vs. image caption; table title.** Contextual explanatory text at the top-left or bottom of an image block = *note*. A centered label pointing directly to an image = *image caption*. A heading above a table = *table title*.

16. **Image caption in context; magazine header/footer vs. note.** A label directly beside or below a photo = *image caption*. Magazine top/bottom elements: judge by their relationship to the main article content.

17. **Background layer determination.** Only the element(s) at the absolute bottom layer spanning the full page width or height = *background*. All elements above = *foreground*.

18. **Decorative pattern vs. prominent pattern; legend vs. chart title.** Minor ornamental element with no direct meaning = *decorative pattern*. Meaningful, conspicuous standalone graphic = *prominent pattern*. Inconspicuous small text adjacent to a figure = *legend*. Prominent label above a figure = *chart title* / *image caption*.

19. **Image caption vs. legend.** Text directly below a photo pointing to it = *image caption*. Text that serves as a note for multiple figures = *legend*.

20. **List item.** Numbered or bulleted list elements that are cohesive and clearly part of a list structure.

21. **Illustration vs. icon (cartoons).** Cartoon resembling a recognizable icon-style symbol = *icon*. Other cartoons = *illustration*.

22. **Background image vs. background pattern (full-bleed non-solid).** If the bottommost layer is a non-solid graphic spanning the full width or height, label it *background image*, not *background pattern*.

23. **Comprehensive annotation examples.** Apply all rules above holistically: identify the bottommost background layer first, then label foreground elements from largest to smallest.

A.5.6 LABEL REFINEMENT

The visual mask rules above describe the fine-grained annotation protocol used during dataset construction. For downstream model training and evaluation in Section 4.2, we use a coarser 12-class visual taxonomy. This coarsening removes distinctions that are useful for annotation guidance but difficult to learn reliably from pixels alone. In particular, *color block (bordered)* and *color block (borderless)* are merged into *Color Block*, and *dialog box (bordered)* and *dialog box (borderless)* are merged into *Dialog Box*. Several labels are renamed into canonical model-facing classes, e.g. *line element* becomes *Line / Divider*, *chart* becomes *Chart / Graph*, and *brand logo*, *photograph*, *icon*, and *table* are kept as direct counterparts. Background- and pattern-related labels are harmonized by visual role: broad page or section backdrops such as *background image* and *natural background* map to *Background Image*, while ornamental or motif-like regions such as *background pattern*, *decorative pattern*, and parts of *prominent pattern* are mapped to either *Decorative / Pattern Graphic* or *Illustration / Artwork* depending on whether the region functions mainly as decoration or as a standalone depicted graphic. Likewise, *document content* regions are absorbed into the closest semantic class in the 12-class taxonomy, most often *Text Block / Content* when the region functions as a readable content panel. Thus, the 17 visual labels should be understood as the fine annotation guideline, while the 12 labels form the normalized visual taxonomy used by the released mask-tag field and downstream tagging experiments.

A.6 QUALITATIVE COMPARISONS OF VISUAL ANCHOR LOCALIZATION

Figures 4, 5, and 6 show one representative validation page for each of three challenging settings: dense small-entity layouts, CJK documents, and visually rich pages. Each comparison shows ground-truth entity masks alongside predictions from fine-tuned YOLOv12-Seg and SAM3; panel headers report the instance counts. Figure 7 extends the comparison to RF-DETR-Seg and the VLM-based LocateAnything grounder.



Figure 4: Qualitative comparison on a representative *dense small-entity* `ADOPD 2026` validation page (sample `8e046669`; 63 OCR blocks; language `zh`). Left to right: ground-truth entity masks, YOLOv12-Seg, SAM3. Per-instance polygons; panel headers give instance counts.



Figure 5: Qualitative comparison on a representative *CJK* `ADOPD 2026` validation page (sample `8fe26edd`; 84 OCR blocks; language `zh`). Left to right: ground-truth entity masks, YOLOv12-Seg, SAM3. Per-instance polygons; panel headers give instance counts.



Figure 6: Qualitative comparison on a representative *visual-rich* `ADOPD 2026` validation page (sample `568c436b`; 90 OCR blocks; language `other`). Left to right: ground-truth entity masks, YOLOv12-Seg, SAM3. Per-instance polygons; panel headers give instance counts.

Comparison including LocateAnything. Fig. 7 shows the two qualitative segmentation comparisons that include the VLM-based LocateAnything grounder. This illustrates that specialist segmenters better preserve instance granularity after fine-tuning, while the unified grounder provides a useful but coarser polygon reference.



Figure 7: Additional qualitative segmentation comparisons. Columns show ground truth, SAM3, RF-DETR-Seg, YOLOv12-Seg, and LocateAnything; colors denote instances and `n` gives the instance count.

A.7 AGENTIC SELF-REFINEMENT: ALGORITHM

Agentic grouping and geometric validation of predicted text boxes

Visual input. Page image with fused detector boxes drawn in color and numbered $0, \ldots, N-1$ ($N \leq 30$).

User prompt. *The image is a document page with detected boxes drawn and numbered. Detectors often over-split one real text block into several boxes. Group the numbered boxes so each group is one text block a human would annotate together. Use visual layout: same column, contiguous lines/items, shared paragraph/list/table context. Do not invent boxes. Output all groups as a JSON list of lists; leave unrelated boxes as singletons.*

Reason-then-parse protocol.

Reason over marks. Send the marked page and prompt; stop generation at `</think>` and save the bounded reasoning trace t.

Parse groups. Reuse the same marked page and prefill the assistant with t followed by `FINAL: [[`, forcing JSON continuation, e.g. `[[1,2,3],[4],[5,6]]`.

Post-check. Each returned group is merged to one union box only if a deterministic geometric guard accepts it; otherwise the member boxes are kept separate.

Figure 8: Prompted agent workflow for self-refinement. The VLM reasons over numbered detector boxes and then reuses the saved trace to produce parseable JSON groups; no coordinate regression is performed.

The algorithm in Fig. 9 gives the full procedure behind the workflow of Fig. 8 (agentic self-refinement in Sec. 4.1.2).

Algorithm 1. Agentic Self-Refinement (per page)

Input: page I ($W \times H$), detectors $\{D_1, .., D_m\}$, VLM $\mathcal{V}$
Param: pool threshold τ=0.05, NMS θ=0.6, cap N=30

$P \leftarrow \bigcup_j D_j(I, \text{conf} \geq \tau)$ — *over-recall pool*
$P \leftarrow \textsc{Wbf}(P)$ — *fuse detectors*
$B \leftarrow \textsc{Nms}(P, \theta)[:N]$ — *dedup, cap N*
if $|B| < 2$ **return** B
$I_m \leftarrow \textsc{DrawNumbered}(I, B)$ — *Set-of-Marks*
$t \leftarrow \mathcal{V}(I_m;\ \text{stop}=\langle/\text{think}\rangle)$ — *reason over marks*
$r \leftarrow \mathcal{V}(I_m;\ \text{prefill}=t \,\|\, \texttt{FINAL:[[})$ — *structured parse*
$G \leftarrow \textsc{ParseGroups}(r)$ — *$\emptyset$ on fail*
$R \leftarrow \{\}$, $U \leftarrow \emptyset$
for $g \in G$: $S \leftarrow \{B[i] : i \in g\}$
  if $|g| \geq 2 \wedge \neg\, \textsc{GeoOk}(S, W, H)$:
    $R \leftarrow R \cup S$ — *keep separate*
  else $R \leftarrow R \cup \{\textsc{Union}(S)\}$ — *merge to 1 box*
  $U \leftarrow U \cup g$
$R \leftarrow R \cup \{B[i] : i \notin U\}$ — *singletons*
return R

$\textsc{GeoOk}(S, W, H)$: *reject* a $\geq$2-box merge if any of: (1) some box's x-overlap with the union x-span < 0.5 (cross-column); (2) $\sum_b \text{area}(b)/\text{area}(\textsc{Union}(S)) < 0.55$ (fat-empty); (3) union height $> 0.9H$, or area $> 0.6WH$, or $|S| > 20$ (page-spanning); else *accept*.

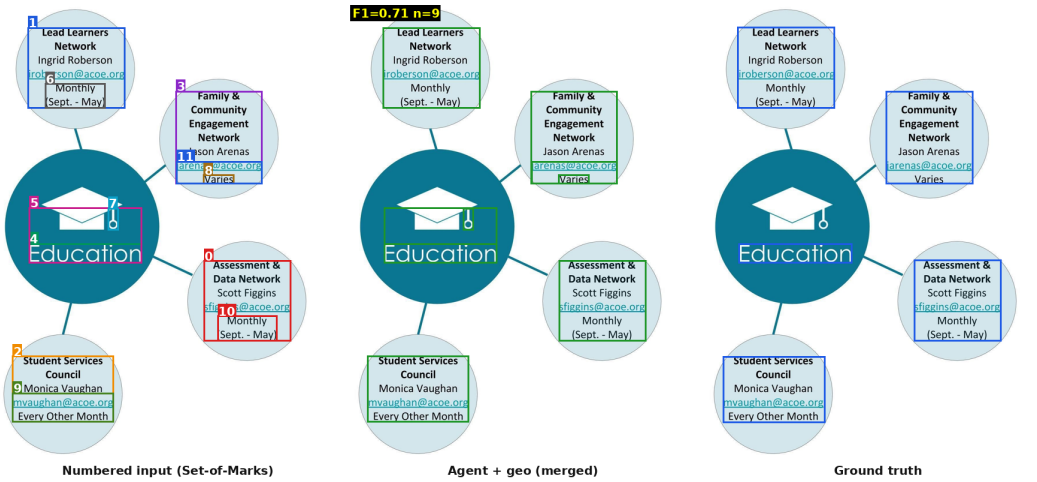
Figure 9: Agentic self-refinement. The agent ($\mathcal{V}$ = Keye-VL-2.0-30B) only *groups* numbered boxes drawn on the page. It never outputs coordinates. A reason-then-parse assistant-prefill procedure yields a parseable grouping despite the reasoner's tendency to over-think, and GeoOk caps over-merges with a deterministic post-check.

A.8 AGENTIC SELF-REFINEMENT: RESULTS AND WORKED EXAMPLE

Table 15 reports the exact values behind the main-text bar trend in Fig. 2b. Fig. 10 connects the numbered detector input to a proposed JSON grouping and the output after geometric validation. In this education infographic, the pairs `[0,10]`, `[1,6]`, and `[2,9]` each refer to text within one network panel. However, the proposed group `[3,7,8,11]` includes both the Family & Community Engagement Network panel and the central graduation-cap tassel (box 7), illustrating why grouping suggestions require a geometric post-check. Boxes 4 and 5 are absent from the JSON; the algorithm retains omitted detections as singletons (Fig. 9, line 15).

Table 15: Agentic self-refinement on the `ADOPD 2026` validation split (mean-F1@IoU0.5, %, over 19,382 pages). All rows share the same over-recall detection pool.

Method	mean-F1	Δ vs. pool
RF-DETR-Large pool (raw)	33.6	—
+ WBF (geometric fusion)	34.2	+0.6
+ Agent grouping	39.6	+6.0
+ Agent grouping + geo	47.7	+14.1
Merge oracle (GT upper bound)	61.4	+27.8



Proposed grouping: `FINAL: [[0,10],[1,6],[2,9],[3,7,8,11]]`

Figure 10: Worked example of agentic self-refinement: numbered input (left), output after grouping and geometric validation (middle), and ground truth (right). The output has 9 boxes from 12 input detections, with per-page F1 increasing from 0.59 to 0.71. The four proposed groups should not be confused with four final output boxes: rejected groups retain their member boxes, and omitted detections are preserved.

A.9 Complete Definitions and Counting Instructions for Target Labels in the DocCount Evaluation Corpus

Target Label Definitions

Brand Logo
A visual identity mark used to identify a brand, organization, product, service, certification, standard, award, or official program. It may appear as a logomark, emblem, seal, badge, stylized wordmark, designed text, certification mark, quality seal, or endorsement badge. A Brand Logo can be the designed mark alone, the mark with its associated name, or the identifying text alone when it functions as the visual identifier. Generic decorative icons are excluded.

Count one Brand Logo as one distinct visual identity instance. If a designed mark and its associated brand name appear together as one logo unit for the same brand, count them as one Brand Logo. If only a designed mark appears, count that designed mark as one Brand Logo. If a stylized wordmark is the brand identifier, count that stylized wordmark as one Brand Logo.

Photograph
Realistic or naturalistic image content depicting real-world people, objects, places, products, scenes, textures, and related content. It may appear as a foreground image or as a background image. Flat illustrations, icons, abstract graphics, patterns, charts, tables, and ordinary text are excluded.

Table
A grid-like arrangement of information organized into rows, columns, or cells for comparison or lookup. Visible grid lines are not required. General page-layout grids and decorative aligned blocks that are not meant to function as tables are excluded.

Menu price lists count as tables when item names and prices or options are arranged in consistent rows and aligned columns, such as item-name columns with corresponding price columns. Menu item grids or product cards do not count as tables when the grid mainly serves a page-layout purpose.

Chart / Graph
Count each self-contained quantitative or categorical data visualization as one chart/graph. A chart/graph represents data values using visual encodings such as position, length, area, color scale, angle, bars, lines, points, slices, heatmap cells, contours, map regions, or network nodes and edges.

Count one chart/graph for each independent visualization unit. A unit is independent if it presents its own dataset, metric, coordinate, radial, or geographic layout, or analytical result. Count small multiples as multiple charts when each repeated panel is a self-contained visualization with its own data display. Count them as one chart only when the repeated elements are clearly components of a single shared comparison structure.

A.10 DocCount Grounded-Narration Qualitative Examples

Fine-grained Counting

[Trigger_Placeholder]
How many distinct Brand Logo regions can you find on the page?



Original Image



Image with Visual Anchors

Anchor Thinking Reasoning

1. Intent Analysis
The task is to count every `[[T]]` region in the document image by following the supplied reading order.

2. Grounded Object Localization
Across the top row, `[[1]]` encloses the Tourism Malaysia mark, `[[2]]` encloses the Discovery wordmark, `[[3]]` encloses the Holiday Rail emblem, `[[4]]` encloses the KTM wordmark, and `[[5]]` encloses the Firefly airline logo. Below them on the right, `[[6]]` marks the First in Malaysia badge. Farther down, `[[7]]` marks the floral Visit Truly Asia Malaysia 2020 emblem.

3. Summation
That makes 7 in total.

Fine-grained Counting

[Trigger_Placeholder]
How many regions on this page should be counted as Photograph?


Original Image



Image with Visual Anchors

Anchor Thinking Reasoning

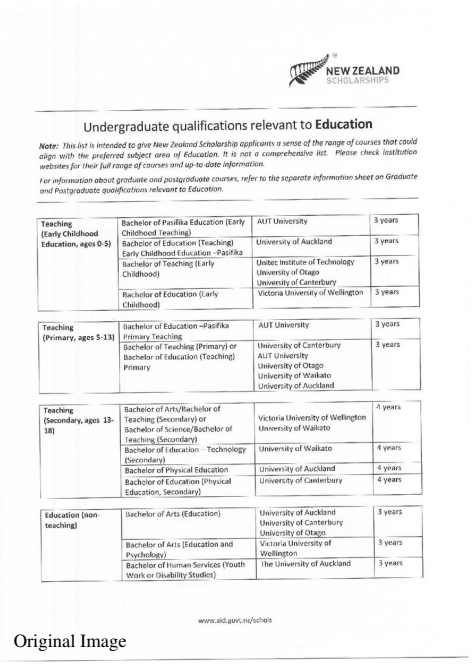
1. Intent Analysis
The task is to count every `[[T]]` region in the document image by following the supplied reading order.

2. Grounded Object Localization
From left to right, `[[1]]` is the large product photograph showing two girls' dresses, `[[2]]` is the upper central photograph of a child modeling an outfit, `[[3]]` is the lower central photograph of another child model, and `[[4]]` is the right-side product photograph showing two clothing sets.

3. Summation
I count 4.

Fine-grained Counting

[Trigger_Placeholder]
Count the Table regions visible on this page.



Original Image

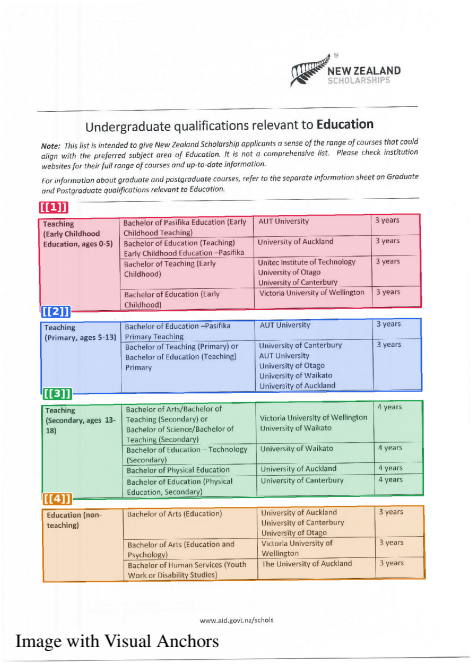


Image with Visual Anchors

Anchor Thinking Reasoning

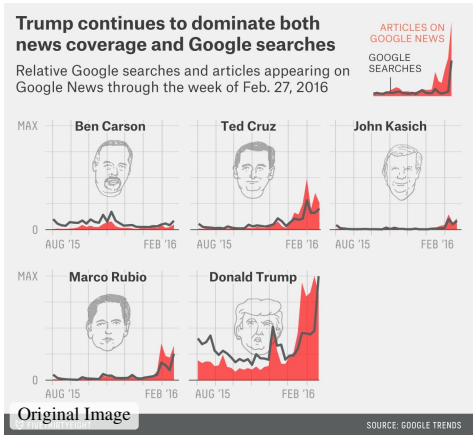
1. Intent Analysis
The task is to count every `[[T]]` region in the document image by following the supplied reading order.

2. Grounded Object Localization
Scanning downward, `[[1]]` is the gridded table for early-childhood teaching qualifications, `[[2]]` is the separate gridded table for primary teaching qualifications, `[[3]]` is the larger gridded table for secondary teaching qualifications, and `[[4]]` is the bottom gridded table for non-teaching education qualifications.

3. Summation
The total is 4.

Fine-grained Counting

[Trigger_Placeholder]
What is the total number of Chart / Graph regions in the page image?



Original Image

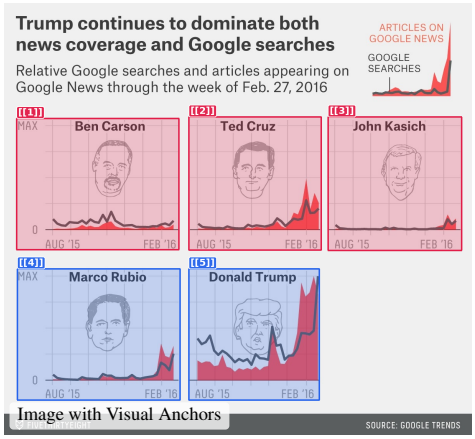


Image with Visual Anchors

Anchor Thinking Reasoning

1. Intent Analysis

The task is to count every `[[T]]` region in the document image by following the supplied reading order.

2. Grounded Object Localization

In the top row of small multiples, `[[1]]` is the Ben Carson time-series panel, `[[2]]` is the Ted Cruz panel, and `[[3]]` is the John Kasich panel. In the second row, `[[4]]` is the Marco Rubio panel and `[[5]]` is the Donald Trump panel; each contains plotted search and news-article trends.

3. Summation

I count 5 total.

Fine-grained Counting

[Trigger_Placeholder]
What count of Brand Logo regions do you see in this page?



Original Image



Image with Visual Anchors

Anchor Thinking Reasoning

1. Intent Analysis
The task is to count every `[[T]]` region in the document image by following the supplied reading order.

2. Grounded Object Localization
In the first awards row, `[[1]]` is the Los Angeles olive-oil gold-medal seal and `[[2]]` is the NYIOOC gold-award emblem. In the next row, `[[3]]` is the CINVE 2017 seal and `[[4]]` is the Mezquita ORG 2017 seal. The following row contains `[[5]]`, the Zurich Olive Oil Award mark, and `[[6]]`, the TerraOlivo emblem. Continuing downward on the left, `[[7]]` is the Athena competition medal, `[[8]]` is the AVPA Paris award seal, and `[[9]]` is the DOMINA olive-oil competition medal.

3. Summation
That gives me 9.

A.11 EXAMPLES IN DOCCOUNT EVALUATION

Brand Logo

Ground Truth **7** Tag **brand logo**

How many distinct Brand Logo regions can you find on the page?



Original Overlay

Kimi K2.5 (reasoning on) predicted 7 correct

Looking at the image, I need to identify distinct visual identity marks that function as brand logos, official program marks, certification seals, or endorsement badges.

Here is the breakdown:

1. **Tourism Malaysia Logo** (top, "Supported by" row): A designed mark with the national flag element and identifying text, functioning as the tourism board's visual identity.
2. **Discovery Logo** (top row, next to Tourism Malaysia): The stylized "D" icon with the "Discovery" wordmark, representing the media/brand partner.
3. **Circular Partner/Destination Logo** (top row, center): A circular badge with a landscape graphic and text, placed as a distinct sponsoring partner identity.
4. **KTM Logo** (top row): The stylized "KTM" lettermark with the signature blue and yellow striped design, identifying the rail service brand.
5. **Firefly Logo** (top right): Consists of the insect icon, the "firefly" stylized wordmark, and the "Beyond Convenience" tagline. Per the definition, a designed mark and its associated brand name appearing together as one unit counts as **one** distinct brand logo.
6. **"FIRST IN MALAYSIA" Badge** (right side, diamond-shaped): This is an endorsement badge/seal making a specific claim about the product's exclusivity. The definition explicitly includes endorsement badges as a type of brand logo.
7. **Visit Truly Asia Malaysia 2020 Logo** (right side, hibiscus and hornbill design): This is the official program/campaign logo for Malaysia's tourism year, functioning as a visual identity mark for that specific government initiative.

The large "FLYRAIL" text is the product/package name presented as a headline, not a distinct corporate visual identity mark. The "SPECIAL LAUNCH PROMO 15% OFF" circle is a promotional callout graphic, not a brand logo. The "Book now!" element in the bottom right is a generic call-to-action with a decorative icon, not a brand identifier.
<answer>7</answer>

Qwen3.6 35B A3B (reasoning off) predicted 7 correct

To determine the number of distinct Brand Logo regions on the page, we need to identify visual identity marks that represent brands, organizations, or official programs — excluding generic decorative icons.

Let's go through the image systematically:

1. **Top right corner group:**
   - "Supported by TOURISM MALAYSIA" — This includes the Tourism Malaysia logo (a stylized flower emblem) with its name. This counts as **one Brand Logo**.
   - "Discovery" — The orange sun-like icon next to the word "Discovery" is the Discovery Channel brand logo. Counts as **one Brand Logo**.
   - "Malaysia Rail" — A circular emblem with a train and text "Malaysia Rail". This is an official program/brand logo. Counts as **one Brand Logo**.
   - "KTM" — The green and blue stylized logo with "KTM" text. This is the Kuala Lumpur Transport Corporation brand logo. Counts as **one Brand Logo**.
   - "firefly Beyond Convenience" — The orange bird-like icon with "firefly" text. This is Firefly airline's brand logo. Counts as **one Brand Logo**.

   → That's 5 logos in the top right.

2. **Right side, middle section:**
   - "FIRST IN MALAYSIA" — This is a badge/seal indicating an official certification or program. It has a distinct design (blue diamond shape with white text). Counts as **one Brand Logo** (official program mark).

- Below it: “Visit Malaysia 2020” — This is the official tourism campaign logo for 2020, featuring a bird and hibiscus. Clearly a brand/logo for a national tourism initiative. Counts as **one Brand Logo**.

→ That’s 2 more.

3. **Bottom right corner:**
- “Book now!” with a stylized sail/boat icon — This appears to be a call-to-action button graphic, not a brand logo. Exclude.

4. **Airplane in sky:**
- The airplane has “firefly” written on its side — but this is part of the aircraft livery, not a separate logo instance. Since we already counted the Firefly logo above, we don’t double-count.

5. **Train:**
- No visible brand logo on the train itself (just locomotive number “2521”), so no additional logo here.

6. **Left side:**
- “FLYRAIL” — This is the name of the package, styled in bold orange letters. However, it’s presented as a product/package name, not necessarily a standalone brand logo. Given the context (“3D2N Holiday Package”), it’s likely a marketing title rather than a registered brand mark. We’ll exclude it unless clearly branded — but note: if “Flyrail” is a co-branded service, it might qualify. However, since it’s not accompanied by a unique emblem or official branding signal (like a seal or trademark symbol), and it’s more of a promotional headline, we will **not count it** as a distinct brand logo per the definition.

7. **Circular orange promo badge:**
- “SPECIAL LAUNCH PROMO 15% OFF Limited time only.” — This is a promotional graphic, not a brand logo. Exclude.

8. **Small airplane on runway:**
- No branding visible. Exclude.

9. **Bottom left corner:**
- “*Terms & Conditions apply.” — Text only, not a logo. Exclude.

Now, summing up the confirmed Brand Logos:

- Tourism Malaysia → 1
- Discovery → 1
- Malaysia Rail → 1
- KTM → 1
- Firefly → 1
- FIRST IN MALAYSIA badge → 1
- Visit Malaysia 2020 → 1

Total = **7**

<answer>7</answer>

GPT-5.5 (reasoning off) predicted 11 wrong

I identified the visible Brand Logo regions as:

1. Tourism Malaysia logo
2. Discovery logo
3. Malaysia Rail logo
4. KTM logo
5. Firefly logo at top right
6. FlyRail stylized wordmark
7. “First in Malaysia” badge
8. Visit Truly Asia Malaysia 2020 logo
9. Firefly branding on the aircraft tail
10. Fireflyz.com.my branding on the aircraft fuselage
11. Firefly-style logomark in the bottom-right orange section

<answer>11</answer>

Claude Sonnet 4.5 (reasoning on) predicted 8 wrong

Detailed Analysis of Brand Logos

Let me systematically examine the image to identify all distinct brand logos:

Top Section - Sponsor Logos (Left to Right):
1. **Tourism Malaysia** - Official tourism organization logo with "Supported by TOURISM MALAYSIA" text
2. **Recovery** - Orange/yellow logo with arrow design (second sponsor)
3. **Circular green/teal logo** - Appears to be an environmental or certification organization (third sponsor)
4. **KTMB** - Blue and yellow railway company logo (Keretapi Tanah Melayu Berhad)
5. **firefly** - Orange/yellow airline logo with "Beyond Convenience" tagline

Main Content Area:
6. **FLYRAIL** - The large orange stylized wordmark serves as the product/service brand identifier for this fly-and-rail package

Right Side Elements:
7. **"FIRST IN MALAYSIA"** - Diamond-shaped badge/seal that functions as a program identifier and promotional emblem
8. **Visit Terengganu Malaysia 2021** - Floral hibiscus design logo for the tourism campaign

Excluded Elements:
- "Book now!" with arrow design - This is a call-to-action graphic element, not a brand logo
- The airplane and train images themselves - These are photographs, not logos
- Numerical identifiers on vehicles - Not brand logos

Each identified element meets the definition criteria as it serves as a visual identity mark for a brand, organization, product, service, or official program.

<answer>8</answer>

Photograph

Ground Truth **4** Tag **photograph**

How many regions on this page should be counted as Photograph?



Original



Overlay

Kimi K2.5 (reasoning on) predicted 6 wrong

Looking at this JCPenney catalog page, I need to identify all regions that are realistic photographs of real-world products or people, excluding text, illustrations, charts, and flat graphics.

Scanning the page layout, I can identify the following distinct photographic regions:

- **Region A (top left):** A realistic product photograph of a red plaid dress with a white lace collar, showing natural fabric texture, folds, and shadows.
- **Region B (bottom left):** A realistic product photograph of a blue jumper and white blouse set, again showing natural textile textures and dimensional draping.
- **Region C (center top):** A realistic photograph of a child model wearing a light blue chambray dress with a dark blue skirt, showing natural skin tones, hair texture, and fabric details.
- **Region D (center bottom):** A realistic photograph of a child model wearing a purple and black striped two-piece outfit, showing natural photographic qualities like hair detail and fabric sheen.
- **Region E (top right):** A realistic product photograph of a cream-colored top and red pants arranged on a flat surface, showing knit texture and garment shaping.
- **Region F (bottom right):** A realistic product photograph of a red, white, and blue dress with vest effect, laid flat or on a form, displaying natural fabric textures and stitching details.

The remaining areas consist of ordinary text (product descriptions, prices, item numbers), typographic headings, and catalog page background—none of which qualify as photographs under the given definition.

There are no illustrations, abstract graphics, or charts present; all six clothing/item displays are naturalistic photographs.

<answer>6</answer>

Qwen3.6 35B A3B (reasoning off) predicted 6 wrong

To determine how many regions on this page should be counted as Photograph, we must apply the given definition:

> Realistic or naturalistic image content depicting real-world people, objects, places, products, scenes, textures, and so on. Exclude flat illustrations, icons, abstract graphics, patterns, charts, tables, and ordinary text.

The page is a catalog spread from JC Penney (1970s), featuring children's clothing. It contains both textual descriptions and visual images of garments and models.

Let's break down the page into distinct visual regions:

1. **Region A** – Top left: Shows a red plaid dress with white collar and apron-style front. The garment is photographed realistically against a neutral background. This qualifies as a photograph.

2. **Region B** – Bottom left: Shows a dark blue polka-dot jumper with white blouse underneath, worn by a child (partially visible). The image is a realistic product shot — qualifies as a photograph.

3. **Region C** – Top middle: Features a girl modeling a denim-style dress with ruffled hem. This is a full-body photo of a real person wearing the item — qualifies as a photograph.

4. **Region D** – Bottom middle: Shows another girl modeling a striped dress with ruffled sleeves. Again, a realistic photo of a model in the garment — qualifies as a photograph.

5. **Region E** – Top right: Displays a two-piece set (top and pants) laid out flat but still presented as a photographic product shot — not illustrated, but photographed. Qualifies.

6. **Region F** – Bottom right: Shows a child's dress with vest-style bodice and floral skirt, again photographed as a product image — qualifies.

Additionally, note that the entire layout includes photographs of actual garments and models — these are all realistic depictions of physical products and people. There are no flat illustrations, icons, charts, or abstract graphics being used to represent the products — everything shown is either a photo of a real garment or a child modeling it.

Therefore, each of the six labeled regions (A through F) represents a distinct photographic region.

There is also the JC Penney logo and page number at bottom left — but those are small textual/logo elements and do not constitute "regions" under the context of counting photographic content for product display. The question implies counting major image regions showcasing products/models.

Thus, we count 6 photographic regions.

<answer>6</answer>

GPT-5.5 (reasoning off) predicted 6 wrong

The page is a catalog layout with text descriptions and several product images. The regions that qualify as **Photograph** are the realistic images of clothing/products and children modeling outfits.

Countable photograph regions:

1. Red plaid dress labeled A
2. Jumper-and-blouse set labeled B
3. Child modeling chambray dress labeled C
4. Child modeling striped skirt set labeled D
5. Pants and top set labeled E
6. Dress with vest-effect labeled F

Text blocks, labels, and background page areas are not counted.

<answer>6</answer>

Claude Sonnet 4.5 (reasoning on) predicted 4 correct

Looking at this catalog page carefully, I need to identify distinct photographic regions based on the definition provided.

Let me analyze each section of the page:

Left Column:
- Shows dress A (red plaid dress with white trim) at the top
- Shows dress B (navy sparkly dress with white blouse) below it
- These appear to be on the same continuous tan/beige background without a dividing line between them
- This appears to be **1 photographic region** containing both items

Center Column:
- Shows a model wearing dress C (blue chambray dress) in the top photo
- Shows a model wearing dress D (purple striped dress) in the bottom photo
- There is a clear horizontal dividing line between these two photographs
- These are **2 separate photographic regions**

Right Column:
- Shows outfit E (red pants and cream top) at the top
- Shows dress F (red, white, and blue dress) below it
- These appear to be on the same continuous tan/beige background without a dividing line between them
- This appears to be **1 photographic region** containing both items

All of these images are photographs of real products and real people (the models), fitting the definition of "realistic or naturalistic image content depicting real-world people, objects, places, products."

The text descriptions at the top are ordinary text and should not be counted.

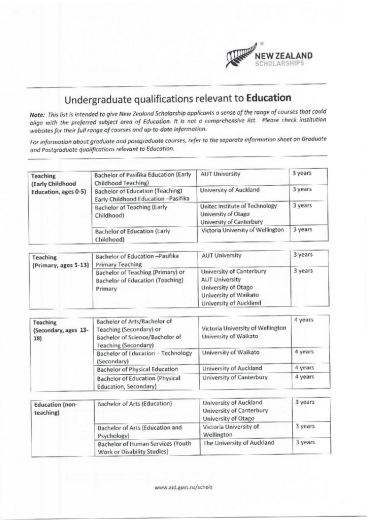
Total count: 1 (left column) + 2 (center photos) + 1 (right column) = 4 photographic regions

<answer>4</answer>

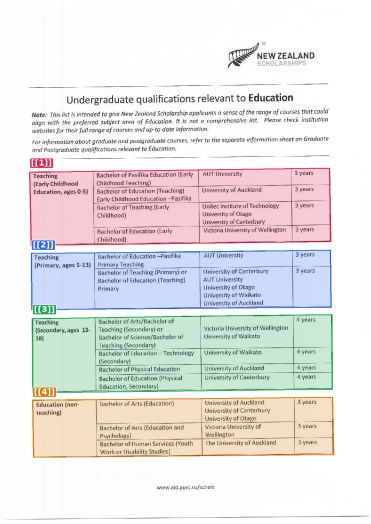
Table

Ground Truth **4** Tag **table**

Count the Table regions visible on this page.



Original



Overlay

Kimi K2.5 (reasoning on) predicted 4 correct

Looking at the page, I can identify distinct arrangements of data organized into rows and columns for the purpose of comparing qualifications across institutions and durations.

Detailed rationale:
- The first table covers **Teaching (Early Childhood Education, ages 0-5)** with 4 rows listing specific bachelor's degrees, their respective institutions, and duration (3 years).
- The second table covers **Teaching (Primary, ages 5-13)** with 2 rows listing pasifika and general primary teaching degrees, institutions, and duration (3 years).
- The third table covers **Teaching (Secondary, ages 13-18)** with 4 rows listing various secondary teaching qualifications, institutions, and duration (4 years).
- The fourth table covers **Education (non-teaching)** with 3 rows listing non-teaching education degrees, institutions, and duration (3 years).

Each of these four sections functions as an independent data table with its own header row and aligned columns for qualification name, institution name, and length of study. They are separated by whitespace and have distinct categorical groupings.

<answer>4</answer>

Qwen3.6 35B A3B (reasoning off) predicted 1 wrong

The document contains one large, structured table that organizes information about undergraduate qualifications relevant to education in New Zealand. This table is divided into four main sections (or row groups), each corresponding to a different teaching qualification area:

1. Teaching (Early Childhood Education, ages 0–5)
2. Teaching (Primary, ages 5–13)
3. Teaching (Secondary, ages 13–18)
4. Education (non-teaching)

Each section consists of multiple rows listing specific degrees, the institutions offering them, and the duration (years). The entire structure functions as a single cohesive table with clear columnar alignment: columns for qualification type, degree name, university/institution, and duration.

Although there are visual separators between the four major sections, they are part of the same overarching table — not separate tables. There are no other distinct tables on the page (e.g., no menus, product grids, or side tables).

Therefore, only one table region is visible.

<answer>1</answer>

GPT-5.5 (reasoning off) predicted 4 correct

The page contains four distinct grid-like table blocks, each separated by whitespace and organized with rows and columns:

1. Teaching — Early Childhood Education
2. Teaching — Primary
3. Teaching — Secondary
4. Education — Non-teaching

Therefore, the number of visible Table regions is:

<answer>4</answer>

Claude Sonnet 4.5 (reasoning on) predicted 4 correct

Looking at this document about undergraduate qualifications relevant to Education, I need to identify all table regions that display information in a grid-like arrangement with rows and columns.

Let me examine the page systematically:

1. **First table**: "Teaching (Early Childhood Education, ages 0-5)" - This section contains a structured grid with columns showing qualification names, institutions, and duration. The information is organized in cells with visible borders.

2. **Second table**: "Teaching (Primary, ages 5-13)" - This is a separate table below the first one, with the same columnar structure showing Bachelor degrees, institutions, and years.

3. **Third table**: "Teaching (Secondary, ages 13-18)" - Another distinct table section showing secondary education qualifications with institutions and 4-year durations organized in rows and columns.

4. **Fourth table**: "Education (non-teaching)" - The final table section at the bottom showing non-teaching education qualifications with their respective institutions and durations.

Each of these sections is clearly separated and functions as an independent table with:
- A left column identifying the category
- Middle columns listing qualifications
- Columns showing institutions
- Right columns showing duration in years

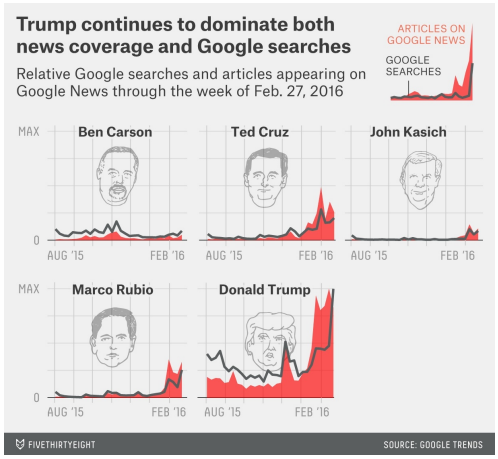
All have visible grid lines forming cells for data organization and comparison purposes.

<answer>4</answer>

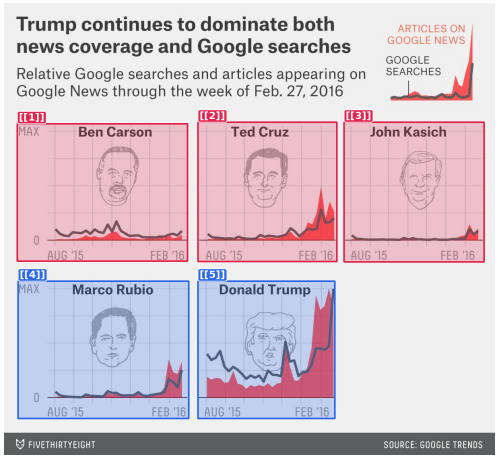
Chart / Graph

Ground Truth **5** Tag **chart / graph**

What is the total number of Chart / Graph regions in the page image?



Original



Overlay

Kimi K2.5 (reasoning on) predicted 6 wrong

The image contains six distinct chart/graph regions. Five of these are the self-contained small-multiple time-series panels, one for each presidential candidate (Ben Carson, Ted Cruz, John Kasich, Marco Rubio, and Donald Trump), where each panel independently plots its own dataset of relative Google searches and news articles over time. The sixth is the separate inset sparkline chart in the top-right corner, which visually encodes the two metrics (Articles on Google News and Google Searches) as a self-contained quantitative graphic distinct from the five candidate panels.

<answer>6</answer>

Qwen3.6 35B A3B (reasoning off) predicted 6 wrong

Let's carefully analyze the image to count the number of Chart / Graph regions.

The image is a data visualization from FiveThirtyEight, titled:
"Trump continues to dominate both news coverage and Google searches"

It includes:

1. A **main header section** with text and a small legend/key graphic in the top right corner — this legend shows two lines (red for "Articles on Google News", black for "Google Searches") over a simplified x-axis. This is a **small illustrative chart/graph** used as a key or example — it has axes, plotted lines, and data representation. So we count this as **1 chart**.

2. Then there are **five distinct panels**, each showing time-series graphs for a different candidate:
   - Ben Carson
   - Ted Cruz
   - John Kasich
   - Marco Rubio
   - Donald Trump

Each of these five panels:
- Has its own title (candidate name)
- Contains a line graph (black = Google Searches, red = Articles on Google News)
- Has its own x-axis (time from AUG '15 to FEB '16)
- Has its own y-axis labeled "MAX" and "0"
- Contains a portrait illustration of the candidate (not part of the data, but the graph itself is quantitative)

These are **independent visualizations** — each represents a separate dataset (for each candidate), with their own coordinate layout and analytical result. They are presented as "small multiples" — repeated structure but each panel is self-contained and presents its own data display.

According to the definition:
> "Count small multiples as multiple charts when each repeated panel is a self-contained visualization with its own data display."

So we count each of the five candidate panels as **separate charts**.

Thus:
- 1 small legend/chart in top right
- 5 candidate-specific charts

Total = 1 + 5 = **6**

Note: The entire composite figure might be considered one infographic, but the question asks for “Chart / Graph regions” — meaning individual graphical representations of data. Each panel qualifies as such.

<answer>6</answer>

GPT-5.5 (reasoning off) predicted 5 correct

The image contains a set of small-multiple time-series visualizations for presidential candidates. Each candidate panel shows quantitative data over time using a red filled area and a gray line, so each panel is a distinct chart/graph.

The self-contained chart regions are:

1. Ben Carson
2. Ted Cruz
3. John Kasich
4. Marco Rubio
5. Donald Trump

The small graphic in the upper-right appears to function as a legend/key explaining the red area and gray line encodings, rather than a separate independent chart.

<answer>5</answer>

Claude Sonnet 4.5 (reasoning on) predicted 6 wrong

Let me analyze this image systematically to count all chart/graph regions.

Analysis:

1. **Top Right Corner**: There is a small chart labeled "ARTICLES ON GOOGLE NEWS" with "GOOGLE SEARCHES" indicator. This shows two overlaid time series (one appears to be searches, one appears to be articles) - this is **1 chart**.

2. **Main Section - Small Multiples Grid**: The image displays 5 individual time-series charts arranged in a grid, each showing data for a different political candidate:
   - **Ben Carson** (top row, left) - 1 chart
   - **Ted Cruz** (top row, center) - 1 chart
   - **John Kasich** (top row, right) - 1 chart
   - **Marco Rubio** (bottom row, left) - 1 chart
   - **Donald Trump** (bottom row, right) - 1 chart

Each of these 5 panels is a self-contained visualization with:
- Its own independent coordinate system and axes
- Its own time range (AUG ’15 to FEB ’16)
- Two overlaid data series (Google searches in dark line, articles on Google News in red/coral fill)
- Its own data specific to that candidate

According to the definition, these small multiples should be counted as separate charts since "each repeated panel is a self-contained visualization with its own data display."

Total Count: 1 (reference chart) + 5 (candidate charts) = **6 charts**

<answer>6</answer>

Brand Logo

Ground Truth **9** Tag **brand logo**

What count of Brand Logo regions do you see in this page?



Original Overlay

Kimi K2.5 (reasoning on) predicted 12 wrong

Rationale: The image displays an awards page with 12 distinct award entries, each accompanied by a unique visual identity mark. These marks include circular seals, rectangular certificates, and badges representing specific olive oil competitions and certifications (e.g., L.A. Olive Oil Awards, CINVE, Zurich, Athena, AVPA Paris, DOMINA, NYIOOC, Mezquita, TerraOlivo, Les Olivalies, Expo Azeite Brazil, and Japan Olive). Each of these 12 marks functions as a brand identifier, emblem, certification seal, or award badge for its respective organization or program. There are no other brand marks present, and the background sunflower is purely decorative.

<answer>12</answer>

Qwen3.6 35B A3B (reasoning off) predicted 12 wrong

Let's carefully analyze the image to count the number of Brand Logo regions.

The image is titled "國際橄欖油競賽獲獎 (Awards)" and lists 12 international olive oil competition awards, each accompanied by a visual award emblem or logo on the left side of the text description.

We are to count **Brand Logo regions** — defined as visual identity marks used to identify brands, organizations, products, services, certifications, standards, awards, or official programs. This includes emblems, seals, badges, stylized wordmarks, etc. We exclude generic decorative icons.

Each award entry has one distinct emblem/logo representing that specific competition or award body. Let's go through them:

Left column (6 entries):
1. L.A. Olive Oil Awards (USA) — Gold Medal → circular gold seal with "LOS ANGELES" and olive branch → 1 Brand Logo
2. CINVE Awards (Spain) — Gold Medal → black and gold emblem with "CINVE 2017" → 1 Brand Logo
3. Zurich Olive Oil Awards (Switzerland) — Gold Medal → white and gold circular emblem with "OLIVE OIL AWARD ZURICH 2017" → 1 Brand Logo
4. Athena Contest (Greece) — Gold Medal → dark circular emblem with "ATHENA CONTEST 2017" → 1 Brand Logo
5. AVPA Paris (France) — Gourmet Gold Medal → golden circular emblem with star design and "AVPA PARIS" → 1 Brand Logo
6. DOMINA Olive Oil Competition (Italy) — Gold Medal → golden circular emblem with "DOMINA IOOC Gold Medal 2017" → 1 Brand Logo

Right column (6 entries):
7. NYIOOC 2017 (USA) — Gold Medal → black and gold laurel wreath emblem with "GOLD AWARD 2017" → 1 Brand Logo
8. Mezquita Awards (Spain) — Gold Medal → greenish-gold circular emblem with "ORO 2017" and "MEZQUITA" → 1 Brand Logo
9. TerraOlivo 2017 (Israel) — Grand Prestige Gold → circular emblem with olive oil pitcher and "TERRAOLIVO ISRAEL" → 1 Brand Logo
10. Les Olivalies (France) — Silver Medal → rectangular certificate-style badge with "Les Olivalies Argente" → 1 Brand Logo (functions as official award seal)
11. Expo Azeite Brazil (Brazil) — Silver Medal → rectangular certificate-style badge with "Expo Azeite IOOC 2017" → 1 Brand Logo
12. Japan Olive (Japan) — Silver Medal → vertical silver badge with "SILVER" and Japanese characters → 1 Brand Logo

All 12 emblems are distinct visual identity marks associated with specific award competitions or organizations. None are generic decorative icons — they all serve as official seals or badges for those programs.

Thus, total Brand Logo regions = 12.

<answer>12</answer>

GPT-5.5 (reasoning off) predicted 12 wrong

I counted the distinct award/certification badge or seal images shown on the page, treating each medal, badge, or certificate-style award mark as one Brand Logo region.

- Left column: 6 award logo/badge regions
- Right column: 6 award logo/badge/certificate regions

I did not count the plain descriptive text lines or the page title as separate logos.

<answer>12</answer>

Claude Sonnet 4.5 (reasoning on) predicted 12 wrong

Let me carefully examine each award shown in this image to count the distinct Brand Logos.

Left Column:
1. **L.A. Olive Oil Awards** - Circular gold seal/badge
2. **CINVE Awards** - Black circular medal with "CINVE 2017" branding
3. **Zurich Olive Oil Awards** - Circular medal with olive branch design and "GOLD" label
4. **Athena Contest** - Circular black and gold medal badge
5. **AVPA Paris** - Circular gold medal with decorative sunburst design
6. **DOMINA Olive Oil Competition** - Circular gold medal with "IOOC Gold Medal" branding

Right Column:
7. **NYIOOC 2017** - Circular black badge with "GOLD AWARD 2017" text
8. **Mezquita Awards** - Circular gold medal with "ORO" designation
9. **TerraOlivo 2017** - Circular olive-green colored badge
10. **Les Olivalies** - Certificate/document displaying the award brand identity
11. **Expo Azeite Brazil** - Certificate/document displaying the award brand identity
12. **Japan Olive** - Rectangular badge/seal showing the award branding

Each of these represents a distinct visual identity mark for a different olive oil competition or award organization. Whether displayed as a medal, badge, seal, or certificate, each serves as the brand logo identifying that particular award program.

<answer>12</answer>