# PERCEIVE–REASON–CODE

Active Perception for Document VQA

**Barış Deniz Sağlam**

Middle East Technical University • Informatics Institute • PhD Student

**ICDAR 2026 DocVQA Challenge • JOINT WINNER • 8–35B PARAMETER TIER**

## THE METHOD IN ONE TRACE

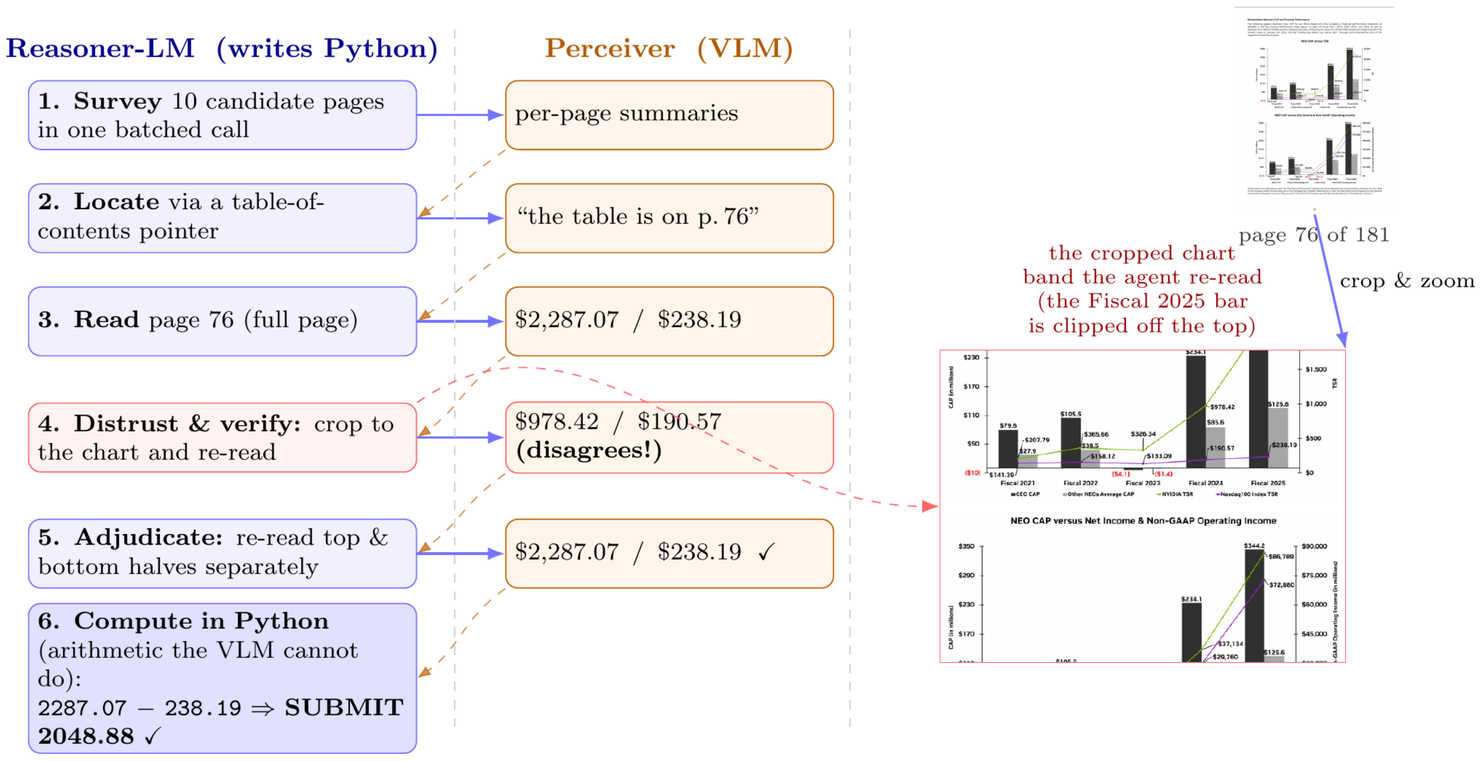


**Fig. 1.** **Perceive–Reason–Code** on a 181-page report. The reasoner-LM writes Python that calls the **Perceiver** (VLM) on regions it chooses: it **surveys** pages, **locates** the table, **distrusts** a full-page read, re-reads a tighter **crop** that disagrees, adjudicates, and **computes** the answer in Python — finding *across* pages and reading *within* one. The crop-verify catches a wrong number a single read would have submitted.

## ABSTRACT

DocVQA (Mathew et al., 2021) asks a model to answer questions over documents from a one-page infographic to a 281-page report. General VLMs read pages at fixed resolution and miss fine detail on dense ones. We give a code-capable model (Qwen 3.5 27B) a **Python REPL** and **one perception primitive** — an on-demand VLM call it invokes region by region — so it can crop, zoom, loop, and compute over pages. It was a **joint winner** of the ICDAR 2026 DocVQA challenge (8–35B tier), ahead of the far larger Gemini 3 Pro and GPT-5.2. Ablations show the lift comes **jointly** from the REPL and the perception call; the trajectory format is a free choice, and a general sub-agent or added OCR even costs accuracy. Perception is the **binding constraint** — no model resolves a dense page in one look — but the **reasoner is the lever**: scaling it moves accuracy twice as much as scaling the VLM.

## THE CHALLENGE

Documents are hard along several axes at once:

- **Across pages** — up to **281 pages**; the evidence must first be *found* among them.
- **Within a page** — a page can be **large** (high-resolution) and **dense** (crowded with labels, cells, lines), so the answer region is tiny relative to the page.

Hand a VLM all pages at once and it reads each at fixed resolution with a fixed slice of attention — fine on a sparse page, but on a large, dense one it cannot resolve the answer. The task is both **finding the right page** and **reading the right region on it**.

## THE METHOD

Give a code-capable model a persistent REPL and one perception primitive — an on-demand VLM call on any region of any page — and let it **direct** perception (Fig. 1):

- **Perceive** — a VLM call on a chosen crop.
- **Reason** — in language, in the transcript.
- **Code** — act in Python; observations flow back in.

Builds on Recursive Language Models (Zhang et al., 2025), CodeAct (Wang et al., 2024), and visual programming (Gupta & Kembhavi, 2023; Surís et al., 2023).

## THE RESULT

| System (held-out test) | Score |
|---|---|
| **Qwen 3.5 27B (ours) — tuned** | **43.8%** |
| **Qwen 3.5 27B (ours) — general** | **39.4%** |
| Gemini 3 Pro | 37.5% |
| GPT-5.2 | 35.0% |
| Gemini 3 Flash | 33.8% |
| GPT-5 Mini | 22.5% |

Baselines are **bare frontier models** (no scaffold). The **tuned entry** that won the tier added DocVQA-specific prompts + OCR/search (43.8%); the **general** method here strips those and still clears the frontier. Specializing buys a few points; generalizing gives them back.

## ABLATIONS

Qwen 3.5 27B as reasoner *and* VLM, DocVQA-2026 val (25 docs / 80 Q), 8 trials, ANLS. Remove one part at a time:

| | with sub-VLM | no sub-VLM |
|---|---|---|
| **with REPL** | **41.9%** full method | 22.3% display() only |
| **without REPL** | 27.2% ReAct | 20.9% raw multi-image |

**Both halves are essential** — drop either and the score falls to the low-20s. **Enrichments don't help:** a general sub-agent costs points (36.7%), the trajectory format is a wash (39.5% vs. 41.9%), and OCR + search on top **costs accuracy** (36.6%): misparsed pages arrive as confident text and crowd out the verifying look.

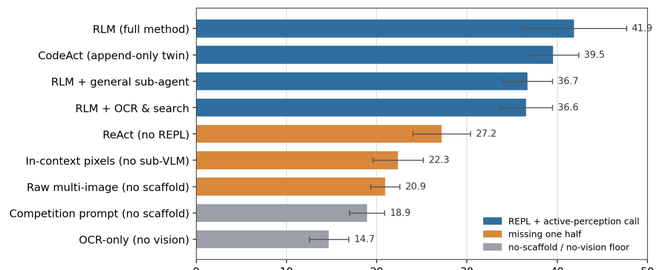


**Fig. 2.** Three clean tiers: REPL + perception (∼36–42%), missing one half (21–27%), OCR-only floor (15%).

## THE ADVANTAGE GROWS WITH LENGTH

On short MP-DocVQA (≤20 pages) RLM leads the raw multi-image baseline by **∼4 pts**; on long MMLongBench-Doc (∼47 pages) by **∼42 pts**. RLM stays flat across both — it navigates the document — while the baseline's "Unknown" rate climbs from 22% to 87% as evidence falls off a fixed page budget. On moderate DocVQA-2026 the edge comes instead from **within-page density** (Fig. 4).

## BETTER EYES, OR A BETTER DIRECTOR?

Swap the **actively driven VLM** for a **precomputed OCR channel** (page text + figure captions + search). Same reasoner, REPL, and text interface. It falls to **14.7%**, the study floor, with **0/10 in all 8 trials** on layout-bound categories (drawings, maps). What collapses is not perception but **agency over it**: question-blind reading cannot replace directed looks.

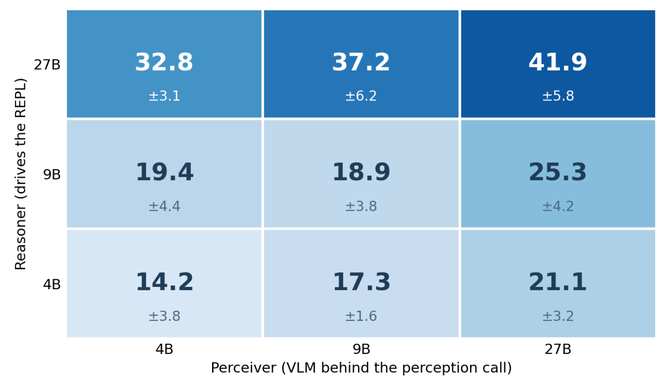


**Fig. 3.** Reasoner × VLM matrix (val ANLS). Across a row (to 27B eyes) +6–9 pts; down a column (better reasoner) +21. A 27B reasoner with 4B eyes (32.8) beats the reverse (21.1).

## ADVANTAGE TRACKS VISUAL DENSITY

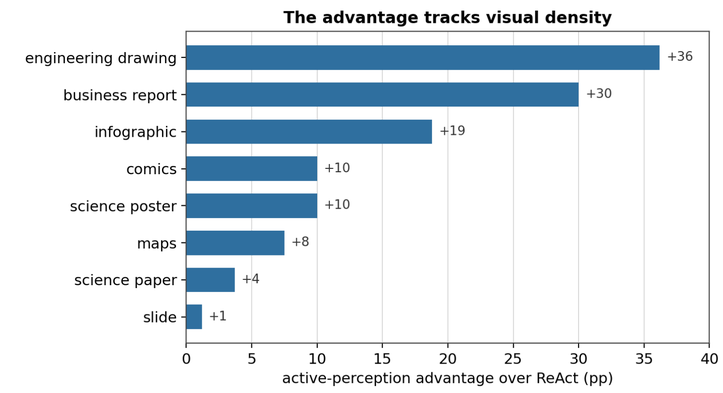


**Fig. 4.** Advantage over ReAct (Yao et al., 2023) by category. Largest on dense pages — engineering drawings (+36), business reports (+30), infographics (+19) — smallest on text-linear ones: science papers (+4), slides (+1).

## LIMITATIONS

**Generality costs calls** — perception is a region at a time, each a sequential VLM call (∼13 steps/Q); unconstrained recursion can be ruinously expensive (Borchmann et al., 2026). More steps mark a *hard* document, not a better answer. Cheap OCR retrieval or a smaller VLM behind the call are open efficiency levers.

**And a floor**: the harness amplifies a strong-enough coder (31B Gemma: +15 pts over ReAct) but cannot rescue one too weak to code (Gemma E4B collapses).

## OUTLOOK

The REPL is a **symbolic substrate** for a neural model — a medium to hold, compose, and compute what its context cannot. That code helps at *test time* is established. The open question: could the symbolic substrate become **part of the model itself** — woven into how it learns and computes, not bolted on at inference?

**The signal is there.** Over 8 trials the right answer is reached **∼69%** of the time (oracle, pass@8) but reliably produced only **∼42%** — a 27-point gap a learning procedure could capture.

**Full technical write-up**

barisdeniz.is-a.dev

github.com/bdsaglam/docvqa

## REFERENCES

Zhang, A.L., Kraska, T., Khattab, O. (2025). *Recursive Language Models.* arXiv:2512.24601.

Wang, X. et al. (2024). *Executable Code Actions Elicit Better LLM Agents (CodeAct).* ICML.

Yao, S. et al. (2023). *ReAct: Synergizing Reasoning and Acting in LMs.* ICLR.

Gupta, T., Kembhavi, A. (2023). *Visual Programming (VisProg).* CVPR.

Surís, D., Menon, S., Vondrick, C. (2023). *ViperGPT.* ICCV.

Borchmann, Ł. et al. (2026). *Strategic Navigation or Stochastic Search? (MADQA).* arXiv:2603.12180.

Mathew, M., Karatzas, D., Jawahar, C.V. (2021). *DocVQA.* WACV.