Efficient Document Parsing via Parallel Token Prediction

Supplementary Material

7. Details of Dataset

7.1. Document Resource Pool

We collect raw document and layout-level OCR data from three channels:

- **Open-source datasets:** We aggregate diverse datasets including layout analysis (DocIIENet[46]), handwriting (GNHK[17], CASIA-HWDB2[22]), and mathematical formulas (Unimernet-1M[39], HME100K[47]). These primarily contain page-level or layout-level images. After format standardization and normalization, we obtain nearly 200K image-text pairs.
- **In-house dataset:** Sourced from our internal document collections, featuring complex layouts and diverse document types. All sensitive and personally identifiable information has been rigorously filtered.
- **Synthetic dataset:** To address specific scenarios such as handwriting recognition, we render images with varying CSS styles, fonts, and corpus, improving model robustness on challenging cases.

7.2. Data Annotation, Cleaning and Augmentation

Following layout-based segmentation of raw documents, we perform comprehensive data annotation, cleaning, and augmentation:

- **Annotation.** We employ a multi-model annotation pipeline: Qwen2.5-VL-72B and MonkeyOCR-Pro-3B generate initial annotations, with confidence scores computed via edit distance between predictions. Low-confidence samples undergo refinement with Gemini-2.5-Pro. All annotations are standardized through rule-based post-processing, followed by LLM-assisted and manual verification for quality assurance.
- **Cleaning.** We implement a multi-stage filtering process: (1) removal of corrupted images and samples with abnormal aspect ratios; (2) filtering of extremely low-confidence annotations; (3) duplicate detection via CLIP embedding similarity (cosine distance threshold) and perceptual hashing (pHash) for pixel-level redundancy removal.
- **Augmentation.** We apply stochastic augmentations during training, including blur, color jitter, geometric distortion, horizontal flipping, Gaussian noise injection, and perspective transformation, to improve model robustness and generalization.

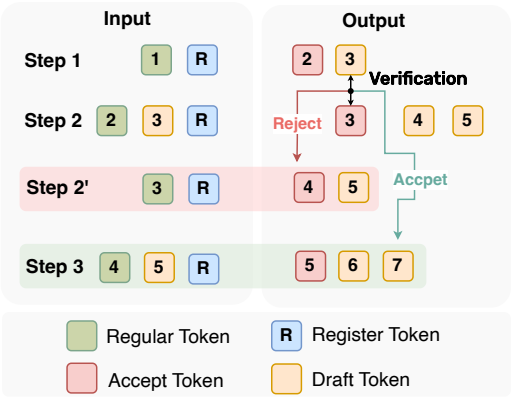

Figure 5. Speculative Decoding

Dataset	Task	NTP	MTP	PTP-1
OmnidocBench	OCR	0.0431	83%	95%
ScienceQA	VLU	92.2	76%	82%
GSM8K	Math	73.3	70%	88%

Table 6. The performance of NTP, and acceptance ratios of MTP / PTP on various tasks.

8. Combining PTP with Speculative Decoding

For tasks beyond image-to-text transcription, such as visual-language understanding (VLU), PTP may result in performance degradation relative to standard NTP. However, our PTP method integrates seamlessly with speculative decoding through a self-verification mechanism for register token predictions, ensuring output consistency with standard NTP. As shown in Fig. 5, we exploit the model's inherent predictions to validate register tokens from the previous decoding step, eliminating the need for external draft models. Specifically, tokens predicted by register tokens serve as draft candidates, which are subsequently verified against regular token predictions in the current step. Only verified tokens are retained in the final output sequence. This self-verification approach is parameter-free, requiring no draft models or auxiliary layers. Compared to standard PTP, it introduces zero computational overhead when draft tokens are accepted, incurring additional cost only upon rejection when re-prediction is necessary. As shown in Tab. 6, using self-speculative decoding, PTP achieves significantly higher acceptance rates on OCR tasks and maintains superior rates on VLU and LM tasks.

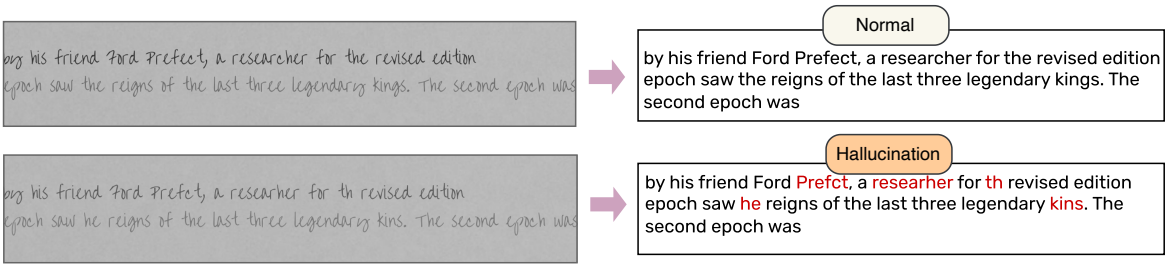

Figure 6. Sampled rendered images and ground truth texts from normal and hallucinated datasets.

9. Details of Experiments

9.1. Detailed Evaluation Metrics

In this section, we elaborate on the evaluation metrics employed in our experiments, which can be categorized into two groups: metrics for assessing model accuracy and metrics for measuring inference efficiency.

The following are the metrics used to evaluate the model's performance:

- **Edit Distance:** we use the Levenshtein distance to measure the minimum number of single-character operations (insertions, deletions, or substitutions) required to transform the prediction into ground truth.
- **Character Detection Matching (CDM):** CDM is proposed by [41] and is used to evaluate formula recognition performance, which renders both the model-predicted LaTeX and the ground-truth LaTeX formulas into image-formatted formulas, then employs visual feature extraction and localization techniques for precise character-level matching, incorporating spatial position information
- **Acceptance Rate:** Accpet rate is an evaluation metric in speculative decoding, which is the percentage of draft tokens accepted during verification. In PTP, we treat future tokens predicted by register tokens as draft tokens and compute their acceptance rate during verification.

The following are the metrics used to evaluate the model's efficiency:

- **Latency:** The time from sending the request to receiving the final token on the user end, which directly affects perceived responsiveness.
- **Time Per Output Token (TPOT):** TPOT measures the average time required to generate each output token during the decoding stage of inference. The average time gap between generating each output token during the decoding stage of inference. A lower TPOT means the model can produce tokens faster, leading to higher tokens per second.
- **Inter-Token Latency (ITL):** The exact pause between two consecutive tokens.
- **Throughput:** Throughput describes how much work an LLM can do within a given period. In this pepr, we focus on Output Tokens per Second (TPS), which provides a finer-grained view of throughput by measuring how many tokens are processed every second across all active requests.

9.2. Analysis of the Number of Register Tokens

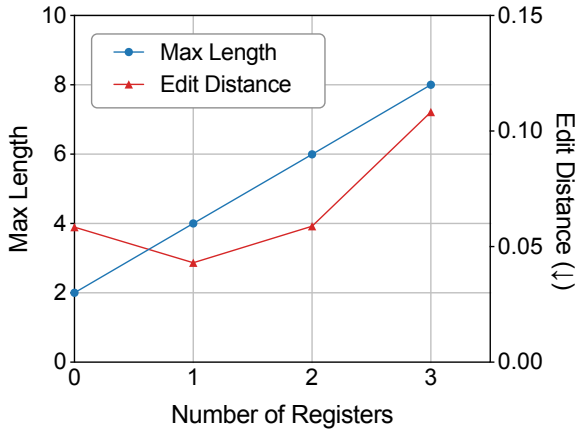

Figure 7. Analysis of the umber of register tokens.

In this section, we investigative the effect of the register tokens numbers during training and inference phases. While register tokens enable future token prediction, indiscriminately increasing their number is counterproductive. As illustrated in Fig. 7, increasing register tokens proportionally expands the input sequence length during training, incurring significant computational overhead. Furthermore, excessively distant predictions suffer from error propagation and increased prediction complexity, ultimately degrading model accuracy. In our experiments, we set the number of register tokens to 2, which achieves an optimal trade-off between inference efficiency and model accuracy.

9.3. Details of Hallucination Evaluation

As illustrated in Fig. 6, we construct a OCR hallucination test set by randomly deleting or substituting characters in words to interfere with the VLM. We ensure consistency in

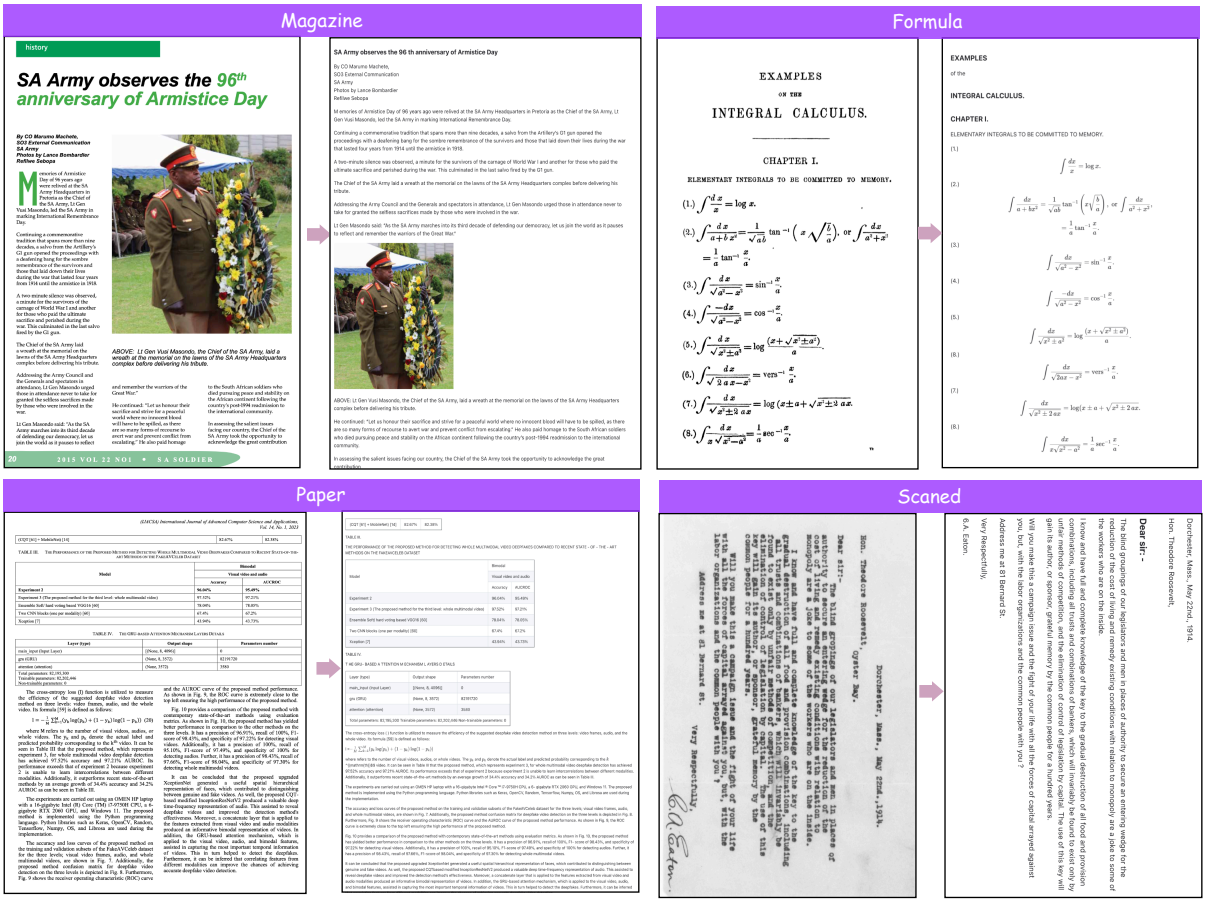

Figure 8. The rendered markdown output for various types of documents of olmOCR-bench.

background color, font, and rendering style between normal and hallucination test sets, with variations limited to text content and corresponding labels. As shown in Fig. 4 (Middle), PTP consistently outperforms NTP on both normal and hallucination data, while demonstrating superior robustness to hallucination-inducing perturbations, thereby validating its effectiveness in hallucination mitigation.

10. Qualitative Examples

We present several qualitative examples in 8.