Beyond Atomic Characters: Glyph-Aware Sub-character Alignment for Low-Resource Multilingual OCR

Mengxiao Zhu[1,2], Haixu Chen[1,2], Jiu Sha[3], Jie Liu[1,2], Ge Shi[4]*

[1]School of Artifical Intelligence and Computer Science, North China University of Technology
[2]Beijing Key Laboratory of Key Technologies for AI+ Domain Applications
[3]School of Information Engineering and the Key Laboratory of Ethnic Language Intelligent Analysis and Security Governance of MOE, Minzu University of China
[4]School of Computer Science & Technology, Beijing Institute of Technology

Abstract

Low-resource multilingual OCR faces a dual challenge: complex script structures and severe data scarcity. In such settings, existing OCR models often struggle, as coarse visual representations combined with weak linguistic priors lead to frequent errors among visually similar characters. To address this, we present BASA (Beyond Atomic Sub-character Alignment), a OCR framework built upon high-resolution visual and language backbones with a novel glyph-aware interface. The core technical contribution is the Glyph-Aware Fine-grained Adapter (GAFA). Unlike standard linear projectors, GAFA employs learnable glyph prototypes to actively align sub-character structural primitives (e.g., strokes and radicals) with visual features, explicitly resolving topological ambiguities during vision–language alignment. To complement this, we introduce a two-stage curriculum learning strategy supported by a Glyph-Aware Reverse Synthesis pipeline, which generates large-scale multilingual training corpora with automatic, zero-cost component labels. Furthermore, we construct BASA-Bench, a representative benchmark spanning 11 languages with diverse script structures and 23 authentic scenarios. Experiments demonstrate that BASA achieves consistent improvements over strong OCR baselines, particularly on scripts with complex compositions. Our model and benchmark will be available at https://github.com/NcutLLM/BASA.

1 Introduction

Optical Character Recognition (OCR) is a critical bridge between physical documents and the digital world, enabling information accessibility and the preservation of cultural heritage (Jain et al., 2021; Naiemi et al., 2022; Neudecker et al., 2021; Agarwal and Anastasopoulos, 2024; Anuradha et al., 2021). While OCR systems have achieved remarkable success for high-resource languages such as English and Chinese, effective OCR for low-resource languages remains a persistent challenge, leaving large volumes of historical and regional documents undigitized.

The difficulty of low-resource OCR arises from two intertwined factors: script complexity and modeling limitations. Many low-resource scripts exhibit intricate glyph compositions (such as stacked characters in Tibetan, cursive connections in Mongolian, or dense diacritics in Vietnamese) where character identity depends on subtle topological differences. Traditional pipeline-based OCR systems (Feng et al., 2025; Wang et al., 2024) decompose the task into detection and recognition stages and rely heavily on large-scale bounding-box annotations. Although effective in controlled settings, they are costly to scale to low-resource languages and brittle under complex layouts due to error propagation. Recent Multimodal Large Language Models (MLLMs) (Comanici et al., 2025; Bai et al., 2025) offer end-to-end transcription without box supervision, but they face a different bottleneck: coarse visual grounding. Standard vision–language projectors treat image patches as holistic tokens, failing to capture fine-grained glyph structures. In low-resource settings where linguistic priors are weak, this often leads to visual ambiguity, hallucinations, and character substitution errors.

These modeling challenges are further exacerbated by severe data scarcity. Unlike high-resource languages, low-resource OCR lacks both large-scale image-text pairs for learning robust visual alignments and rich linguistic corpora for semantic correction. As a result, models cannot reliably disambiguate visually similar characters through either visual evidence or contextual cues. We argue that addressing this deadlock requires moving beyond coarse-grained atomic alignment toward glyph-aware fine-grained grounding, sup-

*Corresponding author, tinkersxy@gmail.com.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 30169–30185

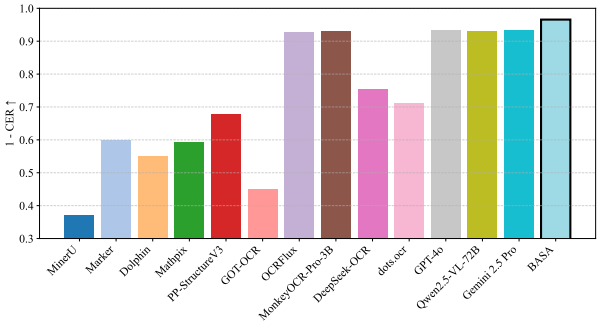


Figure 1: Overall OCR performance on BASA-Bench.

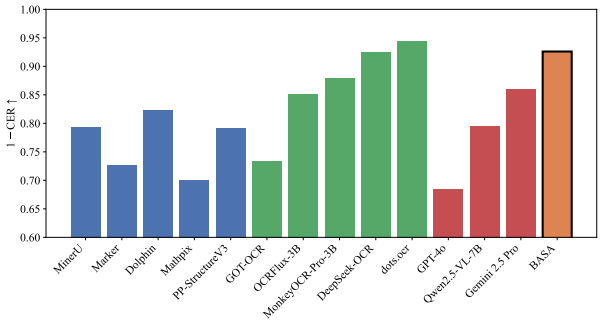


Figure 2: Overall OCR performance on OmniDocBench.

ported by a data ecosystem that explicitly exposes sub-character structure.

Motivated by this insight, we propose BASA (**B**eyond **A**tomic **S**ub-character **A**lignment), a unified framework that integrates architectural innovation with data-centric design. At the model level, we introduce the Glyph-Aware Fine-grained Adapter (GAFA), a lightweight alignment module that injects sub-character structural cues (such as strokes, radicals, and spatial topology) into the vision-language interface. GAFA employs learnable glyph prototypes to actively query and cluster local visual features, enabling the model to distinguish topologically similar characters that are often conflated by standard projectors. At the data level, we construct a dual-source training ecosystem, combining a Glyph-Aware Reverse Synthesis pipeline that provides zero-cost component-level supervision with a curated collection of authentic documents. This design allows the model to learn robust structural representations before adapting to real-world noise and layout variability.

To evaluate low-resource OCR under realistic conditions, we introduce BASA-Bench, a held-out benchmark spanning 11 low-resource languages and 23 real scenarios. Extensive experiments show that BASA consistently outperforms strong pipeline-based systems and state-of-the-art MLLMs, particularly on scripts with complex glyph compositions, underscoring the importance of glyph-aware modeling. Figure 1 and 2 provides an overview of the comparative performance of BASA against strong baselines on both BASA-Bench and OmniDocBench, highlighting its consistent gains under complex and low-resource conditions. In summary, our contributions are threefold:

(1) We propose BASA, a glyph-aware OCR framework that introduces the Glyph-Aware Fine-grained Adapter (GAFA) to explicitly align sub-character structures with visual features, effectively resolving visual ambiguity in low-resource scripts.

(2) We construct a large-scale multilingual data ecosystem that unifies glyph-aware synthetic training corpora and a rigorously curated evaluation benchmark (BASA-Bench), covering 11 low-resource languages and 23 real scenarios.

(3) Through extensive experiments and ablation studies, we demonstrate that BASA consistently outperforms strong OCR methods, with particularly strong gains on scripts featuring complex glyph compositions.

2 Related Work

2.1 OCR Paradigms

Traditional OCR systems adopt *pipeline-based architectures*, such as PP-StructureV3 (Cui et al., 2025), MinerU (Wang et al., 2024), Marker[1], dolphin (Feng et al., 2025) and Mathpix[2], achieve strong performance on structured documents in high-resource settings. However, these methods rely heavily on large-scale bounding-box annotations and handcrafted heuristics, which are difficult to adapt to diverse low-resource scripts.

Recent advances in MLLMs have enabled *end-to-end OCR*. Generalist models, including GPT-4o (Hurst et al., 2024), Gemini (Comanici et al., 2025), and Qwen-VL (Team, 2024b), leverage massive pre-training to perform zero-shot OCR. Expert OCR-oriented VLMs, such as GOT-OCR (Wei et al., 2024), MonkeyOCR-Pro (Li et al., 2025), and dots.ocr (Humane Intelligence Lab (HiLab), Xiaohongshu, 2024), and OCRFlux[3], introduce high-resolution encoders or crop-based processing. However, these methods rely on coarse-grained visual tokens, failing to capture the fine-grained topology required to resolve visual ambiguity in low-resource scripts.

[1]https://github.com/datalab-to/marker
[2]https://github.com/Mathpix
[3]https://github.com/chatdoc-com/OCRFlux

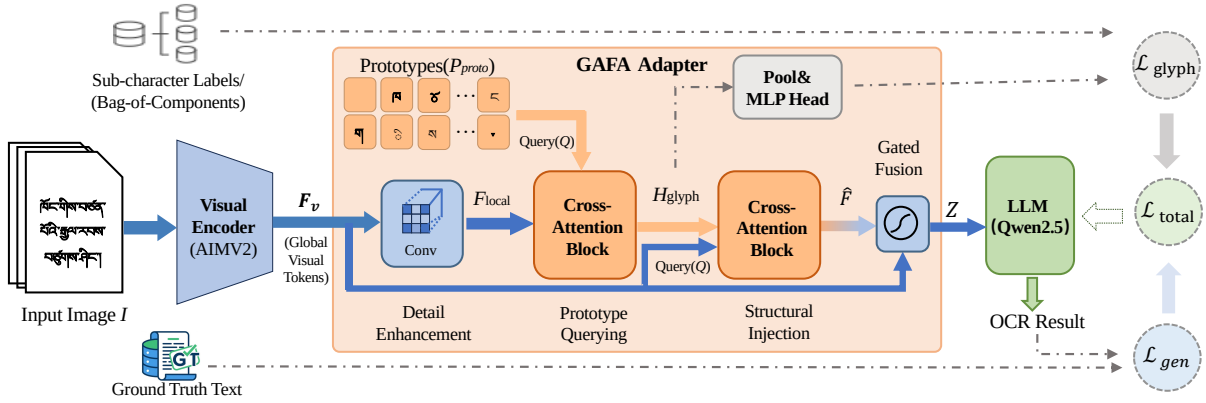


Figure 3: Framework of BASA

2.2 Low-Resource OCR and Data Scarcity

Low-resource OCR faces the dual challenge of script complexity and data scarcity (Agarwal and Anastasopoulos, 2024). While model adaptations like few-shot transfer (Lim et al., 2024) enhance scalability, they struggle to distinguish similar glyphs without explicit sub-character modeling. Similarly, synthetic engines (e.g., SynthTIGER (Yim et al., 2021)) lack the component-level annotations required for fine-grained grounding. Furthermore, benchmarks like OmniDocBench (Ouyang et al., 2025) remain biased toward high-resource languages, leaving low-resource robustness underexplored.

3 Methodology

3.1 Overview

We propose BASA, a framework designed to bridge high-resolution visual perception with fine-grained linguistic priors. As shown in Figure 3, BASA comprises a visual encoder (AIMV2 (Fini et al., 2025)), our novel Glyph-Aware Fine-grained Adapter (GAFA), and a LLM (Qwen2.5). Formally, given image I, the encoder extracts visual tokens $F_v \in \mathbb{R}^{N\times D}$. Unlike standard linear projectors, BASA reformulates alignment as a *prototype-based query process*, grounding predictions in sub-character structure. We adopt a linguistic decomposition rule where a character c is represented by a component set $\mathcal{S}_c = \{s_1, \ldots, s_m\}$ ($s_i \in \mathcal{V}_{\text{comp}}$). This provides data-efficient supervision for low-resource scripts (see Appendix A.1 for script-specific rules).

3.2 GAFA

Current OCR systems primarily rely on globally aggregated visual representations, which tend to over-smooth high-frequency stroke patterns crucial for distinguishing visually similar characters. This limitation is especially severe for low-resource scripts with dense or stacked compositions. To address this issue, we introduce the GAFA, a lightweight plug-and-play module inserted between the visual encoder and the language decoder.

3.2.1 Local Glyph Detail Extraction

Although AIMV2 supports high-resolution inputs and is further fine-tuned with native-resolution adaptation, its global self-attention still tends to favor semantic abstraction and may attenuate stroke-level cues. To enhance local glyph topology, we introduce a detail enhancement layer that selectively amplifies high-frequency patterns.

Given visual tokens $F_v \in \mathbb{R}^{N\times D}$, we rasterize them into a padded 2D feature map $\tilde{F}_v$ using the patch spatial metadata returned by the encoder, accompanied by a validity mask. A depth-wise convolution is applied on the padded map, followed by masking and flattening back to the token sequence:

$$F_{\text{local}} = \text{Flatten}\Big(\text{Conv}_{\text{dw}}(\tilde{F}_v)\Big) + F_v. \quad (1)$$

This operation sharpens edges and stroke intersections while preserving global context. The output $F_{\text{local}} \in \mathbb{R}^{N\times D}$ serves as the source of fine-grained visual details.

3.2.2 Prototype-based Alignment

A key challenge in OCR inference is that the target character identity is unknown, making it infeasible to condition directly on ground-truth component embeddings. GAFA addresses this by introducing a set of learnable glyph prototypes $P_{\text{proto}} \in \mathbb{R}^{K\times D}$, where K is a hyperparameter independent of the component vocabulary size. These prototypes are latent vectors that jointly discover recurring sub-character patterns during training.

Step 1: Prototype Querying. The prototypes act as active queries attending to local visual features to aggregate structural evidence:

$$H_{\text{glyph}} = \text{Softmax}\left(\frac{P_{\text{proto}}(F_{\text{local}}W_K)^\top}{\sqrt{D}}\right)(F_{\text{local}}W_V), \tag{2}$$

where W_K and W_V are learned projections. Through training, the prototypes specialize to recurring glyph primitives, yielding $H_{\text{glyph}} \in \mathbb{R}^{K\times D}$ as a compact structural representation.

Step 2: Structural Injection. To inject prototype-level evidence back into the token space required by the decoder, we perform a reverse attention step. We use the global tokens F_v as queries to allow contextualized tokens to selectively retrieve structural primitives from H_{glyph}:

$$\hat{F} = \text{Softmax}\left(\frac{(F_vW_Q)(H_{\text{glyph}}W'_K)^\top}{\sqrt{D}}\right)(H_{\text{glyph}}W'_V). \tag{3}$$

This produces glyph-enhanced features $\hat{F} \in \mathbb{R}^{N\times D}$ aligned with the original visual tokens.

Step 3: Gated Fusion. Finally, $\hat{F}$ is fused with the original visual tokens via a gated connection:

$$Z = \sigma(W_g[F_v;\hat{F}]) \odot \hat{F} + F_v, \tag{4}$$

where $\sigma(\cdot)$ denotes the sigmoid function and $[\cdot;\cdot]$ denotes concatenation. The fused embeddings $Z \in \mathbb{R}^{N\times D}$ are used for decoding.

3.3 Optimization and Curriculum

3.3.1 Multi-task Objectives

We employ a multi-task objective $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{gen}} + \lambda\mathcal{L}_{\text{glyph}}$. Here, $\mathcal{L}_{\text{gen}}$ is the standard autoregressive loss for vision-language generation. $\mathcal{L}_{\text{glyph}}$ is an auxiliary multi-label binary cross-entropy loss that enforces structural consistency:

$$\mathcal{L}_{\text{glyph}} = -\sum_{s\in\mathcal{V}_{\text{comp}}} \big[y_s \log \hat{p}_s + (1-y_s)\log(1-\hat{p}_s)\big]. \tag{5}$$

where $y_s = 1$ if component s appears in the target text. $\hat{p}_s$ is predicted by a lightweight head on the pooled H_{glyph}. This acts as a structural regularizer without requiring box annotations. Although computed on pooled features, gradients back-propagate to all prototypes, driving specialization.

3.3.2 Two-Stage Curriculum.

Since GAFA is randomly initialized while the visual encoder and LLM are pre-trained, end-to-end optimization may induce shortcut learning, where linguistic priors overshadow fine-grained visual grounding. To disentangle structural perception from semantic reasoning, we adopt a two-stage curriculum learning strategy.

Stage 1 (Structure Pre-training): We freeze the LLM and train GAFA on synthetic, semantically uninformative but structurally valid data constructed in Section 4.1. We set λ high to prioritize $\mathcal{L}_{\text{glyph}}$, anchoring prototypes to visual primitives. $\mathcal{L}_{\text{gen}}$ is retained with a small weight to maintain modality alignment.

Stage 2 (End-to-End Tuning): We introduce semantic complexity using authentic and semantically coherent corpora, and jointly optimize GAFA and the LLM via LoRA. λ is reduced to focus on fluent generation while preserving structural grounding.

Script Category	Language (ISO)	Training Samples *Synthetic*	Training Samples *Real*	Benchmark *BASA-Bench*
Logographic	Chinese (zh)	700k	15k	-
	Cantonese (yue)	400k	7k	1000
Latin / Linear	English (en)	600k	13k	-
	Vietnamese (vi)	400k	7k	1000
	Malay (ms)	300k	8k	1000
	Zhuang (za)	200k	5k	1000
Block / Featural	Korean (ko)	400k	8k	1000
Stacked (2D)	Tibetan (bo)	500k	15k	1000
	Burmese (my)	300k	6k	1000
Connected / Cursive	Mongolian (mn)	300k	6k	1000
	Uyghur (ug)	300k	4k	1000
	Kazakh (kk)	300k	3k	1000
	Kyrgyz (ky)	300k	3k	1000
	Total	**5.0M**	**100k**	**11000**

Table 1: Statistics of constructed data. Languages are grouped by visual topology.

4 Data Construction and Benchmark

To support the proposed structure-first curriculum and enable rigorous evaluation, we construct a dual-purpose data ecosystem covering 13 languages. Our construction pipeline emphasizes two core principles: (1) **Controllable Synthesis**, where we explicitly regulate semantic cues to support curriculum learning; and (2) **Rigorous Quality Control**, ensuring that both synthetic and authentic data meet high-quality standards. Table 1 provides detailed statistics for all data splits. The distributions of languages and fine-grained document categories in the training data are detailed in Appendix B.4.

4.1 Glyph-Aware Reverse Synthesis

We develop a reverse synthesis pipeline to generate large-scale training data with "free" component annotations. Unlike standard methods, our pipeline ensures both visual diversity and textual quality.

Source Corpus Preparation. To ensure the linguistic quality of the training data, we curate raw text from Wikipedia and digitized literature. We apply a rigorous data cleaning pipeline: (1) removing HTML tags and non-target script characters; (2) filtering out low-quality sequences using perplexity scores; and (3) performing strict deduplication to prevent data leakage. This yields a clean corpus of 50M sentences.

The Rendering Pipeline. Given the cleaned text, the engine decomposes characters into sub-character units (Section A.1) and renders them into document images. We simulate diverse visual conditions using 130+ fonts, varying layout strategies (vertical/horizontal), and physics-based noise (blur, occlusion). Crucially, this process automatically generates precise component labels. Based on this pipeline, we construct two subsets:

(1) Semantically-Agnostic Corpora: Comprising 2M images of random character strings and isolated words. These samples are visually valid but linguistically incoherent, forcing the model to learn pure glyph perception without semantic shortcuts.

(2) Semantically-Coherent Corpora: Comprising 2M images of natural paragraphs and complex layouts (e.g., double column). This subset reintroduces semantic context to support fine tuning.

4.2 Authentic Data Construction

To bridge the sim-to-real gap, we collect and annotate a set of authentic document images. This process involves strict quality assurance to ensure benchmark reliability.

Collection and Diversity. To ensure coverage of multiple languages, styles, and scenarios, we collect data from online sources (websites, PDFs, digital media), printed materials (books, newspapers, reports, ancient texts), and real-world scenes (store signs, banners, handwritten notes, posters). Additional structured documents such as PPTs and vertical-script files are also included. Images are captured with varied devices under diverse environmental conditions, targeting in excess of 100K samples of effective samples across 23 authentic scene categories.

Annotation and Quality Control. To ensure high-quality ground truth, we implement a Human-in-the-Loop annotation pipeline. First, initial transcriptions are generated by a strong baseline model to assist annotators. Second, native speakers of the target languages correct the pre-labels using a standardized annotation tool, strictly following OCR guidelines. Finally, we perform a double-blind quality check: 10% of the samples are randomly sampled and reviewed by a senior linguist. Batches with an error rate exceeding 1% are returned for complete re-annotation. This rigorous protocol ensures that our authentic dataset and the BASA-Bench benchmark are highly reliable.

4.3 The Benchmark

We use the same method stated above to construct the benchmark BASA-Bench. For each language, *500* instances collected from the authentic corpus and *500* instances constructed with the synthetic rendering, forming a multilingual evaluation set across 11 low-resource languages and 23 real-world scenarios; all sampled items are strictly excluded from training to avoid data contamination.

5 Experiments

5.1 Evaluation Datasets.

We conduct comprehensive evaluations on four benchmarks: (1) **BASA-Bench (Ours)**: The evaluation set described in Section 4, spanning 11 low-resource languages across 23 diverse scenarios; (2) **OmniDocBench** (Ouyang et al., 2025): A mainstream benchmark for general English and Chinese document understanding under realistic layouts; (3) **OCRFlux-bench**[4]: A human-verified PDF-to-Markdown dataset (2k pages) stressing long-form parsing and formatting fidelity; (4) **OCRFlux-pubtabnet**[5]: A fine-grained table recognition benchmark (9k samples) derived from PubTabNet, evaluating structural parsing across simple and complex table topologies.

5.2 Baseline Models

To ensure a comprehensive evaluation, we compare BASA against state-of-the-art methods across three distinct paradigms: (1) **Pipeline Tools**: We select `MinerU` (Wang et al., 2024), `Marker`[6], `Mathpix`[7], dolphin (Feng et al., 2025) and `PP-StructureV3` (Cui et al., 2025). These represent traditional modular systems widely used for OCR. (2) **Expert VLMs**: We include `GOT-OCR` (Wei et al., 2024), `OCRFlux`[8],

[4]https://huggingface.co/datasets/ChatDOC/OCRFlux-bench-single

[5]https://huggingface.co/datasets/ChatDOC/OCRFlux-pubtabnet-single

[6]https://github.com/datalab-to/marker

[7]https://github.com/Mathpix

[8]https://github.com/chatdoc-com/OCRFlux

Model Type	Methods	CER↓	WER↓	BLEU↑	ANLS↑	METEOR↑
Pipeline Tools	MinerU	0.6299	0.6251	0.3979	0.3983	0.3435
	Marker	0.4009	0.4424	0.7693	0.7749	0.7687
	Dolphin	0.4502	0.4842	0.6478	0.6875	0.6308
	Mathpix	0.4074	0.4169	0.6739	0.7065	0.6897
	PP-StructureV3	0.3220	0.3410	0.7317	0.7780	0.7582
Expert VLMs	GOT-OCR	0.5495	0.5702	0.5123	0.5218	0.4626
	OCRFlux	0.0731	0.1644	0.7224	0.9289	0.8267
	MonkeyOCR-Pro-3B	0.0690	0.1283	0.8044	0.9353	0.8684
	DeepSeek-OCR	0.2461	0.2444	0.7770	0.7546	0.7737
	dots.ocr	0.2878	0.3089	0.8012	0.7993	0.7633
General VLMs	GPT-4o	0.0660	0.1128	0.8373	0.9383	0.8852
	Qwen2.5-VL-72B	0.0685	0.1279	0.8047	0.9357	0.8688
	Gemini 2.5 Pro	0.0656	0.1124	0.8376	0.9387	0.8857
Ours	**BASA**	**0.0343**	**0.0510**	**0.9298**	**0.9688**	**0.9501**

Table 2: Overall results on BASA-Bench (Best results are highlighted in **bold**).

`dots.ocr` (Humane Intelligence Lab (HiLab), Xiaohongshu, 2024), and `MonkeyOCR-Pro-3B` (Li et al., 2025). These are specialized vision-language models fine-tuned specifically for high-resolution optical character recognition. (3) **Generalist VLMs**: We benchmark against frontier multimodal foundation models including `GPT-4o` (Hurst et al., 2024), `Qwen2.5-VL-72B` (Team, 2024b), and `Gemini 2.5 Pro` (Comanici et al., 2025), aimed at assessing the zero-shot capabilities of large-scale generalists on low-resource scripts.

5.3 Evaluation Metrics

To assess recognition accuracy, semantic consistency, and structural fidelity, we employ a comprehensive suite of eight metrics. (1) **Recognition Accuracy**: We report CER and WER (Nazeem et al., 2024), and EDIT (Li et al., 2025) to measure strict transcription precision, alongside ANLS (Biten et al., 2019) for thresholded soft-matching in noisy scenarios. (2) **Semantic Fluency**: We use BLEU (Papineni et al., 2002) and METEOR (Banerjee and Lavie, 2005) to evaluate the lexical overlap and semantic alignment of the generated text. (3) **Structural Fidelity**: To assess layout preservation, we employ TEDS (Zhong et al., 2020) for tabular structure and EDS (Normalized Edit Distance) (Team, 2024a) for Markdown formatting consistency. Detailed description are provided in Appendix C.

For reproducibility, all implementation details are reported in Appendix E.

5.4 Main Results

Results on BASA-Bench. The overall experimental results on the proposed `BASA-Bench` are summarized in Table 2. All reported scores correspond to the macro-average over 11 low-resource languages spanning diverse writing systems and real-world document scenarios. Detailed per-language results are provided in Appendix F. BASA consistently outperforms all baseline methods across all five evaluation metrics. In particular, our model achieves the lowest error rates in terms of CER (0.0343) and WER (0.0510), indicating substantially improved character-level recognition accuracy under low-resource conditions. Compared with strong general-purpose MLLMs such as GPT-4o and Gemini 2.5 Pro, BASA reduces character-level errors by a clear margin, demonstrating more reliable fine-grained visual grounding. From a modeling perspective, traditional pipeline-based OCR systems suffer from severe performance degradation due to error propagation and limited adaptability to complex scripts. Expert OCR-oriented VLMs partially alleviate this issue via high-resolution modeling, yet remain sensitive to visual ambiguity and weak linguistic priors. In contrast, BASA benefits from glyph-aware sub-character alignment and a structure-first training strategy, enabling robust recognition across heterogeneous scripts and layouts. These results validate the effectiveness of explicitly modeling glyph-level structure for low-resource multilingual OCR.

Results on OmniDocBench. Table 3 reports the performance on OmniDocBench, which primarily targets high-resource document understanding scenarios in Chinese and English. Despite being designed for low-resource OCR, BASA achieves competitive or superior results across most evaluation metrics, demonstrating strong generalization beyond its original target setting.

On error-sensitive edit-based metrics, including Overall Edit, Text Edit, and Formula Edit, BASA consistently maintains low error rates in both English and Chinese. Moreover, on structure- and order-sensitive metrics such as Text Edit (ZH), Table Edit, and Read Order Edit, BASA matches

Model Type	Methods	$Overall^{Edit}\downarrow$		$Text^{Edit}\downarrow$		$Formula^{Edit}\downarrow$		$Table^{TEDS}\uparrow$		$Table^{Edit}\downarrow$		$ReadOrder^{Edit}\downarrow$	
		EN	ZH	EN	ZH	EN	ZH	EN	ZH	EN	ZH	EN	ZH
Pipeline Tools	MinerU	0.150	0.357	0.061	0.215	0.278	0.577	78.60	62.10	0.180	0.344	0.079	0.292
	Marker	0.336	0.556	0.080	0.315	0.530	0.883	67.60	49.20	0.619	0.685	0.114	0.340
	Dolphin	0.206	0.306	0.107	0.197	0.447	0.580	77.30	67.20	0.180	0.285	0.091	0.162
	Mathpix	0.191	0.365	0.105	0.384	0.306	0.454	77.00	67.10	0.243	0.320	0.108	0.304
	PP-StructureV3	0.145	0.206	0.058	0.088	0.295	0.535	-	-	0.159	0.109	0.069	0.091
Expert VLMs	GOT-OCR	0.287	0.411	0.189	0.315	0.360	0.528	53.20	47.20	0.459	0.520	0.141	0.280
	OCRFlux	0.195	0.281	0.064	0.183	0.379	0.613	71.60	81.30	0.253	0.139	0.086	0.187
	MonkeyOCR-Pro-3B	0.138	0.206	0.067	0.107	0.246	0.421	81.50	87.50	0.139	0.111	0.100	0.185
	DeepSeek-OCR	**0.123**	0.157	0.049	0.087	**0.242**	**0.377**	86.30	88.40	0.147	0.080	0.056	0.085
	dots.ocr	0.125	0.160	**0.032**	0.066	0.329	0.416	88.60	89.00	0.099	0.092	0.040	0.067
General VLMs	GPT-4o	0.233	0.399	0.144	0.409	0.425	0.606	72.00	62.90	0.234	0.329	0.128	0.251
	Qwen2.5-VL-72B	0.214	0.261	0.092	0.180	0.315	0.434	82.90	83.90	0.341	0.262	0.106	0.168
	Gemini 2.5 Pro	0.148	0.212	0.055	0.168	0.356	0.439	85.80	86.40	0.130	0.119	0.049	0.121
Ours	**BASA**	0.126	**0.132**	0.035	**0.061**	0.328	0.389	**88.90**	**90.20**	**0.097**	**0.091**	**0.042**	**0.066**

Table 3: Evaluation results on OmniDocBench (Best results are highlighted in **bold**).

Model Type	Methods	$OCRFlux-bench-single^{AvgEDS}\uparrow$			$OCRFlux-pubtabnet-single^{AvgTEDS}\uparrow$		
		EN	ZH	Total	Simple	Complex	Total
Pipeline Tools	MinerU	0.840	0.813	0.825	0.905	0.893	0.899
	Marker	0.893	0.869	0.881	0.902	0.868	0.885
	Dolphin	0.874	0.810	0.842	0.895	0.878	0.886
	Mathpix	0.856	0.823	0.839	0.897	0.877	0.887
	PP-StructureV3	0.893	0.793	0.843	0.877	0.821	0.849
Expert VLMs	GOT-OCR	0.802	0.742	0.772	0.791	0.739	0.765
	OCRFlux-3B	0.971	0.962	0.967	0.912	0.807	0.861
	MonkeyOCR-Pro-3B	0.828	0.731	0.780	0.880	0.826	0.853
	DeepSeek-OCR	0.899	0.848	0.874	0.946	0.924	0.935
	dots.ocr	0.942	0.928	0.935	0.784	0.771	0.778
General VLMs	GPT-4o	0.998	0.997	0.999	0.804	0.676	0.742
	Qwen2.5-VL-72B	0.769	0.764	0.796	0.560	0.470	0.510
	Gemini 2.5 Pro	0.996	0.998	0.991	00.803	0.668	0.737
Ours	**BASA**	0.909	0.932	0.921	0.912	0.868	0.891

Table 4: Evaluation results on `OCRFlux-bench-single` and `OCRFlux-pubtabnet-single`.

or outperforms strong general-purpose VLMs and OCR-oriented baselines, indicating improved robustness to complex layouts and visual ambiguity.

For structure understanding tasks, BASA achieves the best performance on Table TEDS in both languages, outperforming models such as GPT-4o, Gemini 2.5 Pro, and dots.ocr. This advantage highlights the effectiveness of GAFA in jointly modeling fine-grained visual cues and global layout structure, leading to more stable parsing behavior in complex document scenarios.

Overall, these results show that the proposed structure-first, sub-character-aware modeling approach not only benefits low-resource OCR, but also generalizes to high-resource languages and diverse document understanding tasks without catastrophic forgetting. Detailed OmniDocBench results are provided in Appendix G.

Results on OCRFlux-bench. Table 4 reports results on OCRFlux-bench-single and OCRFlux-pubtabnet-single, which emphasize structure-sensitive edit metrics under complex document layouts. Although BASA is not explicitly designed for document-level structure modeling, it demonstrates stable and competitive performance.

On OCRFlux-bench-single, BASA achieves strong AvgEDS scores with balanced performance in English and Chinese, indicating robust recognition under visually complex settings. On OCRFlux-pubtabnet-single, BASA performs well on dense and irregular table layouts, outperforming general-purpose VLMs and approaching expert OCR systems. We attribute these gains primarily to improved discrimination of fine-grained glyph details,

Category	Variant	CER↓	WER↓	BLEU↑	ANLS↑	METEOR↑
Architecture	BASA (Full)	**0.0343**	**0.0510**	**0.9298**	**0.9688**	**0.9501**
	w/o GAFA (Linear Projector)	0.0480	0.0570	0.5470	0.8960	0.8710
	w/o GAFA (Cross-Attention)	0.0400	0.0530	0.6830	0.9020	0.9010
Supervision	No Glyph Supervision	0.0498	0.0614	0.5382	0.8724	0.8410
Training	One-stage Training	0.0463	0.0544	0.8503	0.8257	0.7871
	No Structure-first Stage	0.0431	0.0538	0.8757	0.8689	0.8442
Data Source	Only Real Data	0.0536	0.0674	0.7744	0.7369	0.7140
	Only Synthetic Data	0.0368	0.0523	0.8585	0.8642	0.9284

Table 5: Ablation studies on the BASA-Bench benchmark.

which helps reduce local visual ambiguities that frequently propagate into structural edit errors.

While BASA does not consistently surpass highly specialized OCR systems in visually simpler or high-resource scenarios, its performance remains stable across all evaluated settings, without signs of catastrophic forgetting. Compared with the internal baseline Qwen2.5-VL-7B, BASA yields consistent improvements on both benchmarks, suggesting that the observed gains stem from the glyph-aware alignment mechanism rather than increased reliance on large-scale language priors.

5.5 Ablation Studies

Effect of the Glyph-Aware Adapter. Table 5 (Rows 1-3) evaluates the architectural contribution of GAFA. Replacing GAFA with a Linear Projector causes a large degradation, especially on BLEU, indicating that simple projection fails to preserve fine-grained glyph evidence. Cross-Attention improves over linear mapping but remains consistently inferior to the full model, suggesting that GAFA's prototype-based querying and write-back provide a stronger inductive bias for aggregating local glyph cues and disambiguating visually ambiguity.

Effect of Glyph-level Supervision. Table 5 (Row 4) shows that removing $\mathcal{L}_{\text{glyph}}$ degrades all metrics, highlighting that architecture alone is insufficient to stabilize prototype specialization. The drop on ANLS/METEOR suggests unreliable grounding under noisy layouts, consistent with prototype degeneration without structural constraints.

Effect of the Two-stage Curriculum. Table 5 (Rows 5-6) validates the structure-first curriculum. One-stage training underperforms on both accuracy and quality metrics, while skipping the structure-first stage yields partial recovery but still lags behind the full setting. This indicates that early-stage glyph-focused optimization is critical to prevent semantic shortcuts and to establish stable structure-aware alignments before instruction tuning.

Effect of Data Source (Synthetic vs. Real). Table 5 (Rows 7-8) studies data composition. Training with only real data performs worst, reflecting limited coverage in low-resource settings. Only synthetic data approaches the full model on CER/WER but still trails on semantic metrics, implying a realism gap. Combining both yields the best overall results, confirming that reverse-synthesized data provides scalable structural supervision while authentic data anchors real-world language usage and layouts.

We further include qualitative OCR case studies in Appendix H.

6 Conclusion

In this work, we revisit low-resource multilingual OCR from a structural perspective and show that its core difficulty lies not merely in data scarcity, but in the lack of explicit *glyph-level grounding*. Through systematic analysis and experimentation, we demonstrate that treating characters as atomic visual units is insufficient for scripts with complex internal composition, especially when linguistic priors are weak. Our findings suggest that effective low-resource OCR requires a tight coupling between *fine-grained visual structure* and *appropriate supervision*. By introducing a glyph-aware alignment mechanism and supporting it with sub-character-aware training data, BASA provides a practical instantiation of this principle. The consistent gains observed across diverse scripts and document conditions indicate that structural grounding plays a more critical role than increased model capacity alone. Beyond the proposed model, this work contributes a scalable data construction pipeline and a challenging benchmark that together expose failure modes overlooked by existing evaluations. We hope these resources will encourage future research to move beyond coarse-grained recog-

nition toward structurally grounded, data-efficient OCR systems for underrepresented languages.

Limitations

While BASA demonstrates consistent improvements for low-resource multilingual OCR, several limitations remain.

First, the effectiveness of glyph-aware alignment depends on the availability of linguistically meaningful sub-character decompositions. In this work, we rely on established linguistic rules and Unicode standards to define component inventories. Although this design avoids manual annotation, it may not generalize seamlessly to scripts with poorly standardized decompositions or highly cursive writing styles where sub-character boundaries are ambiguous. Extending glyph-aware supervision to such scripts remains an open challenge.

Second, the auxiliary glyph-consistency loss provides image-level supervision rather than precise spatial alignment. Our approach intentionally avoids bounding-box annotations to reduce annotation cost, but this also limits the granularity of structural supervision. While experiments show that global component supervision is sufficient to guide prototype specialization, finer spatial supervision could further improve robustness for extremely dense layouts or severely degraded documents.

Third, BASA introduces additional computational overhead during training. The prototype-based querying and auxiliary loss slightly increase training complexity compared to standard projection-based adapters. Although these components are discarded or simplified at inference time, the training cost may still pose challenges for practitioners with limited computational resources.

Finally, our benchmark, while diverse, does not exhaustively cover all low-resource writing systems. BASA-Bench focuses on 11 representative languages and 23 real scenarios, but many scripts and real-world conditions remain unexplored. Future benchmarks could further expand language coverage and incorporate handwritten or historical documents with extreme stylistic variation.

Acknowledgment

This work was supported by the Youth Research Special Project of NCUT (110051360025XN077-13), Provincial-Level Research Platform Operations Support-Beijing Key Laboratory of Key Technologies for AI+ Domain Applications, and the Joint Fund Key Program of the National Natural Science Foundation of China(U23B2029).

References

Milind Agarwal and Antonios Anastasopoulos. 2024. A concise survey of ocr for low-resource languages. In *Proceedings of the 4th Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP 2024)*, pages 88–102.

Isuri Anuradha, Chamila Liyanage, and Ruvan Weerasinghe. 2021. Estimating the effects of text genre, image resolution and algorithmic complexity needed for sinhala optical character recognition. *International Journal on Advances in ICT for Emerging Regions (ICTer)*, 14(3).

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluis Gomez, Marçal Rusinol, Ernest Valveny, CV Jawahar, and Dimosthenis Karatzas. 2019. Scene text visual question answering. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4291–4301.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Cheng Cui, Ting Sun, Manhui Lin, Tingquan Gao, Yubo Zhang, Jiaxuan Liu, Xueqing Wang, Zelun Zhang, Changda Zhou, Hongen Liu, and 1 others. 2025. Paddleocr 3.0 technical report. *arXiv preprint arXiv:2507.05595*.

Hao Feng, Shu Wei, Xiang Fei, Wei Shi, Yingdong Han, Lei Liao, Jinghui Lu, Binghong Wu, Qi Liu, Chunhui Lin, and 1 others. 2025. Dolphin: Document image parsing via heterogeneous anchor prompting. *arXiv preprint arXiv:2505.14059*.

Enrico Fini, Mustafa Shukor, Xiujun Li, Philipp Dufter, Michal Klein, David Haldimann, Sai Aitharaju, Victor G. Turrisi da Costa, Louis Béthune, Zhe Gan, Alexander Toshev, Marcin Eichner, Moin Nabi, Yinfei Yang, Joshua M. Susskind, and Alaaeldin

El-Nouby. 2025. Multimodal autoregressive pre-training of large vision encoders. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition,*, pages 9641–9654.

Humane Intelligence Lab (HiLab), Xiaohongshu. 2024. dots.ocr: A multilingual document ocr system. https://dotsocr.xiaohongshu.com. Code available at https://github.com/rednote-hilab/dots.ocr.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Pooja Jain, Kavita Taneja, and Harmunish Taneja. 2021. Which ocr toolset is good and why: A comparative study. *Kuwait Journal of Science*, 48(2).

Zhang Li, Yuliang Liu, Qiang Liu, Zhiyin Ma, Ziyang Zhang, Shuo Zhang, Zidun Guo, Jiarui Zhang, Xinyu Wang, and Xiang Bai. 2025. Monkeyocr: Document parsing with a structure-recognition-relation triplet paradigm. *arXiv preprint arXiv:2506.05218*.

Jit Yan Lim, Kian Ming Lim, Chin Poo Lee, and Yong Xuan Tan. 2024. Ssl-protonet: Self-supervised learning prototypical networks for few-shot learning. *Expert Systems with Applications*, 238:122173.

Fatemeh Naiemi, Vahid Ghods, and Hassan Khalesi. 2022. Scene text detection and recognition: a survey. *Multimedia Tools and Applications*, 81(14):20255–20290.

Meharuniza Nazeem, R Anitha, S Navaneeth, and 1 others. 2024. Open-source ocr libraries: A comprehensive study for low resource language. In *Proceedings of the 21st International Conference on Natural Language Processing (ICON)*, pages 416–421.

Clemens Neudecker, Konstantin Baierer, Mike Gerber, Christian Clausner, Apostolos Antonacopoulos, and Stefan Pletschacher. 2021. A survey of ocr evaluation tools and metrics. In *Proceedings of the 6th International Workshop on Historical Document Imaging and Processing*, pages 13–18.

Linke Ouyang, Yuan Qu, Hongbin Zhou, Jiawei Zhu, Rui Zhang, Qunshu Lin, Bin Wang, Zhiyuan Zhao, Man Jiang, Xiaomeng Zhao, and 1 others. 2025. Omnidocbench: Benchmarking diverse pdf document parsing with comprehensive annotations. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24838–24848.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

ChatDOC Team. 2024a. Ocrflux. https://ocrflux.pdfparser.io/#/blog. GitHub repository.

Qwen Team. 2024b. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2.

Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, and 1 others. 2024. Mineru: An open-source solution for precise document content extraction. *arXiv preprint arXiv:2409.18839*.

Haoran Wei, Chenglong Liu, Jinyue Chen, Jia Wang, Lingyu Kong, Yanming Xu, Zheng Ge, Liang Zhao, Jianjian Sun, Yuang Peng, and 1 others. 2024. General ocr theory: Towards ocr-2.0 via a unified end-to-end model. *arXiv preprint arXiv:2409.01704*.

Moonbin Yim, Yoonsik Kim, Hancheol Cho, and Sungrae Park. 2021. Synthtiger: Synthetic text image generator towards better text recognition models. In *16th International Conference on Document Analysis and Recognition*, volume 12824, pages 109–124.

Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. 2020. Image-based table recognition: data, model, and evaluation. In *European conference on computer vision*, pages 564–580. Springer.

Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. 2019. Publaynet: largest dataset ever for document layout analysis. In *2019 International conference on document analysis and recognition (ICDAR)*, pages 1015–1022. IEEE.

A Detailed Methodology Settings

A.1 Linguistic Sub-character Decomposition

Many low-resource writing systems encode discriminative visual information at the sub-character level, such as radicals, diacritics, or stacked components. However, existing OCR methods typically treat characters as atomic symbols, ignoring their internal structure, which leads to systematic confusions among visually similar glyphs under limited supervision.

To expose such fine-grained structure, we adopt a linguistically motivated sub-character decomposition that unifies script-specific units under a common notion of glyph components. For a character c, we define its component set as

$$\mathcal{S}_c = \{s_1, \ldots, s_m\}, \quad s_i \in \mathcal{V}_{\text{comp}}, \tag{6}$$

where $\mathcal{V}_{\text{comp}}$ denotes the global component vocabulary shared across languages. Decomposition is performed at the Unicode character level after decoding, rather than on subword tokens.

We instantiate this decomposition according to script-specific conventions: (1) radicals and strokes for logographic scripts; (2) base forms and subjoined components for stacked abugida scripts (e.g., Tibetan); and (3) base letters and diacritic marks for featural or diacritic-rich alphabets (e.g., Korean and Vietnamese). These decompositions follow established linguistic standards and Unicode specifications, requiring no additional manual annotation.

Component-level supervision is highly shareable across characters, enabling data-efficient learning of glyph primitives when character-level samples are rare. This property provides the linguistic foundation for the glyph-aware alignment module described next.

B Dataset Construction

B.1 Data Sources and Preprocessing

The training and evaluation data used in this work are constructed from two complementary sources. Figure 5 shows the pipeline of data construction. First, we generate large-scale synthetic document images through a glyph-aware reverse synthesis pipeline, which allows explicit control over sub-character composition, layout variation, and script-specific writing rules. This synthetic data provides dense and noise-free structural supervision at scale.

Second, we curate a collection of authentic document images from publicly accessible materials, including books, reports, and newspaper documents. These data are selected to reflect realistic visual noise, layout diversity, and script variability encountered in real-world OCR scenarios.

To ensure responsible data usage, all authentic documents undergo a strict preprocessing and filtering procedure. Any potentially identifying content, such as personal names, addresses, signatures, or contact information, is removed or obscured prior to annotation. As a result, the final dataset contains no personal or private information and is suitable for OCR-focused research.

B.2 Human Annotation Workflow

For the authentic document subset, we adopt a Human-in-the-Loop annotation pipeline to ensure transcription accuracy and consistency. Initial OCR transcriptions are generated using a strong baseline model to reduce annotation overhead. Human annotators then correct these pre-generated outputs by carefully comparing them against the document images.

All annotators are native speakers or professionally proficient users of the target languages, with prior experience in OCR correction or linguistic annotation. This ensures reliable handling of script-specific characteristics such as stacked glyphs, cursive connections, and diacritics. Annotation is performed using a dedicated web-based interface that presents the document image and OCR text side by side, allowing precise visual alignment during correction. Annotators were recruited through academic collaborations and language-specific expert networks. All annotation work was compensated on an hourly basis, with payment rates set above the local minimum wage in the annotators' respective regions.

To maintain annotation quality, we apply a double-layer verification mechanism. A random subset (10%) of annotated samples is independently reviewed by a senior annotator. If the error rate of a batch exceeds a predefined threshold, the entire batch is returned for re-annotation.

B.3 Annotation Guidelines and Consistency Control

To ensure consistency across languages and document types, we provide unified annotation guidelines to all annotators. Annotators are instructed to rely strictly on visual evidence in the document image and avoid introducing semantic guessing or normalization beyond what is explicitly visible.

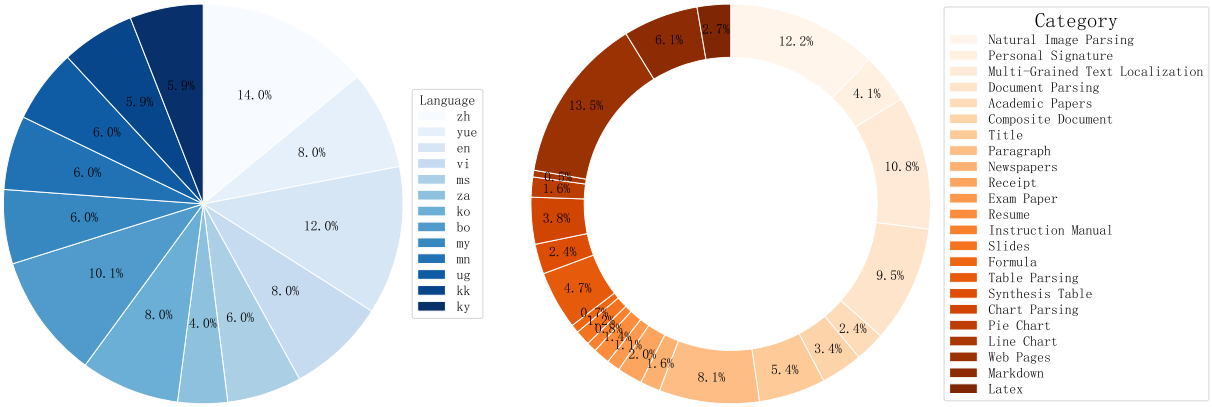


Figure 4: Distributions of languages (left) and fine-grained document categories (right) in the collected dataset.

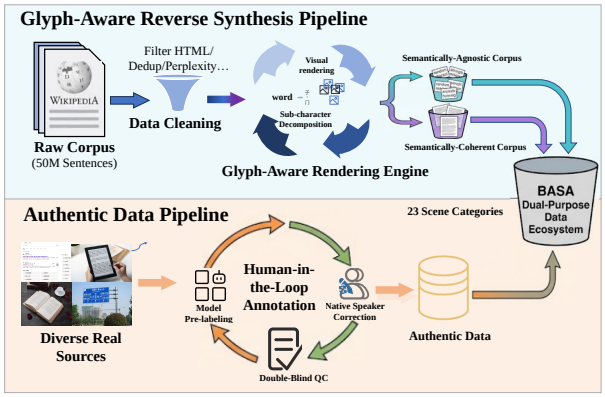


Figure 5: Pipeline of Data Construction.

In particular, annotators are required to: (i) correct recognition errors while preserving the original formatting and reading order; (ii) follow language-specific writing conventions, including script direction, glyph composition, and diacritic placement; (iii) avoid hallucinating characters that are visually ambiguous or partially missing.

Ambiguous or low-quality samples can be flagged during annotation and are subsequently reviewed by senior annotators. This process helps prevent error propagation and ensures that the resulting annotations accurately reflect visual structure rather than linguistic inference.

B.4 Data Statistics

Figure 4 illustrates the data distribution of the collected corpus. The left chart shows the language distribution across the 13 supported languages, indicating a balanced coverage without dominance from any single language. The right chart presents the distribution of fine-grained image categories, spanning diverse real-world scenarios such as natural image parsing, academic papers, forms, receipts, and tables. This diversity in both language and document type ensures that the dataset captures realistic structural and layout variations, supporting robust training under practical OCR conditions.

Figure 6 illustrates some samples in the dataset.

C Evaluation Metrics: Formal Definitions

For completeness, we provide the formal definitions of all evaluation metrics employed in this study.

C.1 Character Error Rate (CER)

Character Error Rate is computed using Levenshtein distance at the character level:

$$CER = \frac{S + D + I}{N} \tag{7}$$

where S, D, and I denote the number of substitutions, deletions, and insertions, respectively, and N is the total number of characters in the reference text.

C.2 Word Error Rate (WER)

Word Error Rate is analogous to CER but computed at the word level:

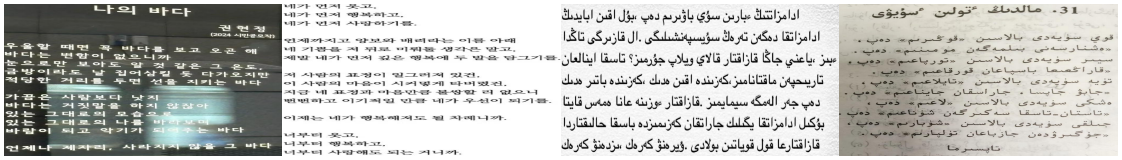
$$WER = \frac{S + D + I}{N} \tag{8}$$

where the operations are measured on word tokens rather than characters.

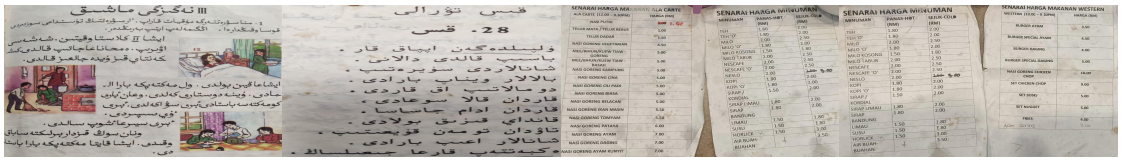
C.3 BLEU

BLEU score evaluates the precision of n-grams with a brevity penalty:

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{9}$$

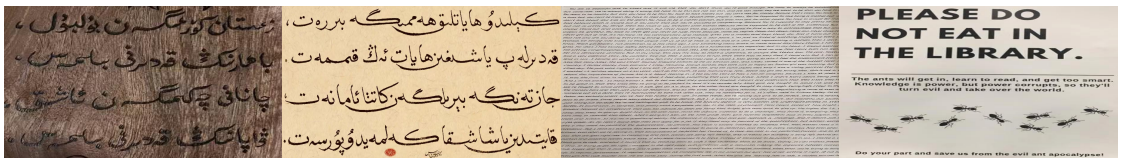


(a) Korean(Geometric Shapes) (b) Korean(Geometric Shapes) (c) Kazakh(Notes) (d) Kazakh(Magazines)

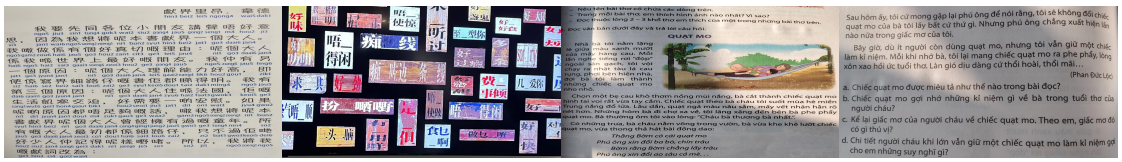


(e) Kyrgyz(Textbooks) (f) Kyrgyz(Exams) (g) Malay(Geometric Shapes) (h) Malay(Geometric Shapes)

(i) Mongolian(Slides) (j) Mongolian(Magazines) (k) Burmese(Geometric Shapes) (l) Burmese(Textbooks)



(m) Uyghur(Geometric Shapes) (n) Uyghur(Religious Scriptures) (o) English(Coord-Based OCR) (p) English(Color-Based OCR)



(q) Cantonese(Tables) (r) Cantonese(Geometric Shapes) (s) Vietnamese(Textbooks) (t) Vietnamese(Textbooks)

(u) Chinese(Color-Based OCR) (v) Chinese(Mind Maps) (w) Zhuang(Textbooks) (x) Zhuang(Mind Maps)

(y) Yi script(Mind Maps) (z) Yi script(Newspapers) () Tibetan(User Manuals) () Tibetan(Religious Scriptures)

Figure 6: Examples of the dataset. It comprises data across 13 languages, organized into 23 detailed subtypes.

where p_n is the modified n-gram precision, w_n is the weight (usually uniform), and $BP = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}$ is the brevity penalty with c the candidate length and r the reference length.

C.4 Average Normalized Levenshtein Similarity (ANLS)

ANLS measures similarity based on normalized Levenshtein distance (NL):

$$ANLS = \frac{1}{N} \sum_{i=0}^{N} \left(\max_j s(a_{ij}, o_{q_i}) \right),$$

$$s(a_{ij}, o_{q_i}) = \begin{cases} 1 - NL(a_{ij}, o_{q_i}), & \text{if } NL(a_{ij}, o_{q_i}) < \tau \\ 0, & \text{if } NL(a_{ij}, o_{q_i}) \geq \tau \end{cases}$$

where a_{ij} is a ground-truth answer, o_{q_i} the model prediction, and NL denotes the normalized Levenshtein distance (Biten et al., 2019).

C.5 METEOR

METEOR integrates unigram precision P, recall R, and a fragmentation penalty F_{pen}:

$$METEOR = F_{mean} \cdot (1 - F_{pen}) \tag{10}$$

where

$$F_{mean} = \frac{10PR}{R + 9P}, \tag{11}$$

and $F_{pen} = \gamma \left(\frac{ch}{m}\right)^{\beta}$, with m the number of matched unigrams, ch the number of chunks, and γ, β empirically set constants.

C.6 Tree Edit Distance Similarity (TEDS)

TEDS evaluates the structural similarity between predicted and reference hierarchical representations (e.g., HTML or XML-based tables). It is computed as:

$$TEDS = 1 - \frac{TED(p, r)}{\max(|p|, |r|)} \tag{12}$$

where $TED(p, r)$ is the tree edit distance between the predicted structure p and ground-truth r, and $|p|$, $|r|$ are the number of nodes in each tree. A higher TEDS indicates closer structural alignment. We adopt the implementation from (Zhong et al., 2019).

Figure 7: Training loss curves of BASA during optimization.

C.7 Edit Distance Similarity (EDS)

EDS measures layout-sensitive string similarity by computing normalized edit distance between Markdown representations of predicted and reference documents:

$$EDS = 1 - \frac{ED(p, r)}{\max(|p|, |r|)} \tag{13}$$

where $ED(p, r)$ is the character-level edit distance, and $|p|$, $|r|$ are the string lengths of the predicted and reference Markdown sequences, respectively. EDS is particularly suited for evaluating structured document OCR results.

C.8 EDIT

EDIT refers to a general-purpose similarity metric based on normalized Levenshtein distance between predicted and reference sequences:

$$EDIT = 1 - \frac{LD(p, r)}{\max(|p|, |r|)} \tag{14}$$

where $LD(p, r)$ is the Levenshtein distance between predicted text p and ground-truth r. Unlike ANLS, EDIT provides a continuous score without thresholding, and is layout-agnostic.

D Training Loss Curves

Figure 7 shows the training loss curves of BASA during optimization. The raw loss exhibits expected stochastic fluctuations due to mini-batch sampling, while the moving-average trend demonstrates a smooth and consistent decrease throughout training. This behavior indicates stable convergence without signs of divergence or oscillation, validating the effectiveness of the proposed optimization strategy and curriculum schedule.

E Implementation Details

Model Configuration. We instantiate the visual backbone with AIMV2-3B, which processes document images using a dynamic patching strategy with a nominal patch size of 14×14. Visual features are projected to match the hidden size of the language model (D). The language decoder is initialized from Qwen2.5-3B.

The GAFA module is inserted between the visual encoder and the LLM. It maintains $K = 1024$ learnable glyph prototypes. The Detail Enhancement Layer uses a depth-wise convolution with kernel size 3×3 and padding 1. Both prototype-query and write-back attention are implemented with multi-head attention using 8 heads.

The auxiliary glyph classification head consists of a LayerNorm, followed by a two-layer MLP with hidden dimension 1024, a GeLU activation, and a final projection to $|\mathcal{V}_{\text{comp}}|$.

During Stage 2 training, we apply LoRA to the LLM. LoRA is injected into the query and value projections of self-attention layers, with rank $r = 32$ and scaling factor $\alpha = 32$.

Optimization Hyperparameters. We use AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and weight decay 0.05. The learning rate follows a cosine decay schedule with a warm-up ratio of 0.03. Mixed-precision training is used throughout.

Stage 1 (Structure Pre-training): Trained for 1 epoch on synthetic, semantically-unpredictable structural data. The LLM and most visual encoder layers are frozen. Learning rate is 2×10^{-4} with a global batch size of 128. The glyph loss weight is set to $\lambda = 10.0$, while the generation loss is weakly weighted for modality alignment. One epoch is sufficient as this stage focuses on prototype initialization rather than full convergence.

Stage 2 (End-to-End Tuning): Trained for 1 epoch on a mixed corpus of synthetic and authentic data. The visual encoder is frozen, and GAFA and the LLM are jointly optimized. Learning rate is reduced to 2×10^{-5} with a global batch size of 64. The glyph loss weight is reduced to $\lambda = 0.2$.

Inference. We use beam search with a beam size of 3 and a length penalty of 1.0. The maximum generation length is set to 2048 tokens. The auxiliary glyph classification head is removed at inference time.

Hardware and Environment. All experiments are conducted on NVIDIA A100 GPUs (80GB) using 8 GPUs with data parallelism. We implement all models using PyTorch.

Lang	Metrics				
	CER↓	WER↓	BLEU↑	ANLS↑	METEOR↑
Kk	0.0017	0.0024	0.9851	0.9879	0.9695
Ko	0.0084	0.0097	0.9849	0.9887	0.9552
Ky	0.0019	0.0025	0.9797	0.9898	0.9530
Mn	0.1067	0.1606	0.7585	0.9588	0.9512
Ms	0.0005	0.0007	0.9955	0.9968	0.9862
My	0.0163	0.0243	0.8466	0.8987	0.8836
Ug	0.1889	0.2822	0.7684	0.9497	0.8991
Vi	0.0043	0.005	0.9571	0.9886	0.9433
Yue	0.0007	0.0014	0.9872	0.9987	0.9951
Za	0.0002	0.0003	0.9973	0.9988	0.9923
Zw	0.0478	0.0719	0.9674	0.8998	0.8987
Avg	**0.0343**	**0.0510**	**0.9298**	**0.9688**	**0.9501**

Table 6: Per-language OCR performance of BASA on the BASA-Bench, reported using CER, WER, BLEU, ANLS, and METEOR.

F Evaluation Across Languages on BASA-Bench

Table 6 presents detailed performance breakdowns for each of the 11 low-resource languages included in the BASA-Bench. Overall, BASA achieves the best or near-best results across all five metrics, indicating strong robustness across diverse languages and writing systems. A closer inspection reveals that the gains are particularly evident for languages with complex glyph structures or limited semantic redundancy, where baseline models are more prone to character substitution and visual confusion. In contrast, BASA maintains consistently lower CER and WER across these challenging cases, suggesting improved stability in fine-grained character recognition.

In addition to character-level accuracy, BASA also shows consistent advantages on sequence-level semantic metrics, including BLEU, ANLS, and METEOR. This observation indicates that enhanced glyph-level perception contributes not only to more accurate recognition but also to improved semantic consistency of the generated text. Taken together, the per-language results confirm that the performance improvements of BASA generalize well beyond a small subset of languages, and remain stable across scripts with varying structural complexity and real-world document conditions.

Model Type	Models	Book	Slides	Financial Report	Textbook	Exam Paper	Magazine	Academic Papers	Notes	Newspaper	Overall
Pipeline Tools	MinerU	0.055	0.124	0.033	0.102	0.159	0.072	0.025	0.984	0.171	0.206
	Marker	0.074	0.340	0.089	0.319	0.452	0.153	0.059	0.651	0.192	0.274
	Dolphin	0.091	0.131	0.057	0.146	0.231	0.121	0.074	0.363	0.307	0.177
	Mathpix	0.131	0.220	0.202	0.216	0.278	0.147	0.091	0.634	0.690	0.300
	PP-StructureV3	0.092	0.254	0.088	0.165	0.242	0.176	0.177	0.489	0.287	0.208
Expert VLMs	GOT-OCR	0.111	0.222	0.067	0.132	0.204	0.198	0.179	0.388	0.771	0.267
	OCRFlux-3B	0.068	0.125	0.092	0.102	0.119	0.083	0.047	0.223	0.536	0.149
	MonkeyOCR-Pro-3B	0.054	0.203	0.038	0.112	0.138	0.111	0.032	0.194	0.136	0.120
	DeepSeek-OCR	0.052	0.090	0.034	0.091	**0.079**	0.079	0.048	0.100	0.099	0.075
	dots.ocr	0.031	0.047	0.011	**0.082**	**0.079**	**0.028**	0.029	0.109	**0.056**	**0.055**
General VLMs	GPT4o	0.157	0.163	0.348	0.187	0.281	0.173	0.146	0.607	0.751	0.316
	Qwen2.5-VL-7B	0.148	0.053	0.111	0.137	0.189	0.117	0.134	0.204	0.706	0.205
	Gemini 2.5 Pro	0.100	**0.040**	0.080	0.090	0.130	0.08	0.090	0.140	0.490	0.140
Ours	**BASA**	**0.026**	0.120	**0.010**	0.090	0.129	0.086	**0.024**	**0.092**	0.131	0.074

Table 7: The text recognition performance on **OmniDocBench** across 9 PDF page types. Each column represents performance on a specific document domain. **Bold** indicates the best result per types.

G Evaluation Across Document Types on OmniDocBench

Table 7 reports text recognition performance on OmniDocBench across nine representative PDF page types. Although OmniDocBench primarily targets high-resource English and Chinese documents, it provides a useful testbed to examine whether glyph-aware modeling preserves robustness beyond low-resource settings.

Across structure-intensive domains such as Books, Financial Reports, Academic Papers, and Notes, BASA achieves strong or best-in-class performance. These document types are characterized not only by complex layouts, but more importantly by dense typography, mixed fonts, small glyphs, and visually ambiguous character patterns. The consistent gains in these categories indicate that explicit sub-character alignment improves the model's ability to resolve fine-grained visual distinctions, thereby reducing cumulative recognition errors in visually crowded text regions.

On semi-structured domains such as Textbooks and Exam Papers, BASA remains competitive with strong expert OCR systems and general-purpose VLMs. In these cases, character shapes are relatively regular and language priors are strong, which reduces the relative advantage of glyph-level modeling. Nevertheless, BASA does not incur noticeable degradation, suggesting that the proposed glyph-aware mechanism does not interfere with standard semantic decoding.

For visually simpler or highly stylized domains such as Slides, Magazines, and Newspapers, BASA does not consistently outperform all baselines. These domains typically involve large fonts, limited character ambiguity, and strong domain-specific visual regularities, where fine-grained glyph modeling is less critical. We attribute the remaining gap primarily to limited exposure to large-scale high-resource data rather than deficiencies in the proposed alignment mechanism.

Overall, the results on OmniDocBench demonstrate that glyph-aware, structure-first modeling primarily benefits scenarios where character-level visual ambiguity dominates, while remaining robust in settings where such ambiguity is minimal. This confirms that BASA's improvements stem from enhanced fine-grained visual grounding, rather than reliance on layout heuristics or high-resource semantic priors.

H Case Study

H.1 Qualitative Analysis on Multilingual Fine-grained OCR

Figure 8 presents a qualitative case study illustrating BASA's OCR behavior on a real-world multilingual document containing mixed Chinese and Korean text. The input image (left) consists of multiple short paragraphs with dense character distributions, numerals, and punctuation, while the output (right) shows the recognized text produced by BASA.

This example highlights several key advantages of the proposed glyph-aware modeling paradigm. First, BASA accurately preserves character-level details in visually similar glyphs, such as numerals ("7902.3 / 79023000") and punctuation, which are common sources of confusion in low-resource or cross-lingual OCR. In particular, subtle differences in stroke topology and spacing are correctly resolved, indicating effective sub-character grounding rather than reliance on coarse visual patches or language priors.

Second, the model demonstrates strong robust-

Figure 8: A qualitative case study of multilingual OCR results

ness under mixed-script conditions. Chinese and Korean sentences are interleaved within the same document, yet BASA maintains stable recognition boundaries without script interference. This suggests that the glyph prototypes learned by GAFA can generalize across writing systems while remaining sensitive to script-specific sub-character structures.

Third, BASA avoids common hallucination and substitution errors observed in baseline systems, such as merging adjacent characters, dropping modifiers, or normalizing numerals based on semantic bias. The recognized output closely matches the input content at both the character and word levels, even in semantically similar but visually distinct contexts. This behavior supports our claim that explicit glyph-level alignment is critical for resolving fine-grained visual ambiguities, especially when linguistic context alone is insufficient.

Overall, this case study qualitatively confirms the effectiveness of BASA's structure-aware design. By grounding recognition in sub-character visual evidence rather than purely semantic decoding, BASA achieves faithful transcription in challenging multilingual scenarios, complementing the quantitative gains reported in the main experiments.

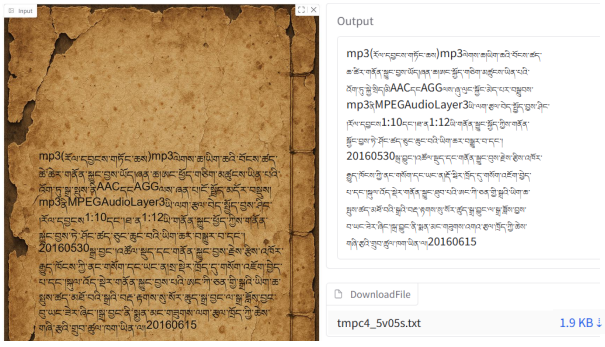


Figure 9: Qualitative OCR case on a degraded Tibetan document

H.2 OCR on Degraded Tibetan Manuscripts

Figure 9 presents a qualitative OCR case on a Tibetan document exhibiting severe degradation, including background stains, uneven illumination, faded ink, and partial character erosion. Such documents pose significant challenges for OCR systems, as Tibetan script features stacked glyph compositions and dense sub-character structures, where subtle visual cues are critical for correct character discrimination.

As shown in the output, BASA successfully preserves the internal structure of Tibetan glyphs and produces a coherent transcription that maintains correct character shapes, stacking order, and line continuity, despite the presence of heavy visual noise. In particular, subjoined consonants and diacritic components are consistently recognized, indicating that the model is able to rely on fine-grained visual evidence rather than coarse patch-level semantics.

This behavior directly reflects the effectiveness of the proposed glyph-aware alignment mechanism. By introducing learnable glyph prototypes and enforcing component-level supervision during training, BASA is encouraged to attend to stable sub-character primitives even when global appearance cues are corrupted. Traditional pipeline-based OCR systems or generic vision–language models often struggle in this setting, as they tend to either over-segment degraded regions or hallucinate visually plausible but structurally incorrect characters.

Overall, this case highlights BASA's robustness to document degradation and its ability to faithfully transcribe complex low-resource scripts under realistic and challenging conditions. It further supports our central claim that explicitly modeling glyph-level structure is essential for reliable OCR in low-resource and historical document scenarios.