SCIENCE CHINA
Information Sciences

• Supplementary File •

MonkeyOCR: document parsing with a structure-recognition-relation triplet paradigm

Zhang LI[1], Yuliang LIU[1*], Qiang LIU[2], Zhiyin MA[1], Ziyang ZHANG[1], Shuo ZHANG[1], Biao YANG[1], Zidun GUO[1], Jiarui ZHANG[2], Xinyu WANG[1] & Xiang BAI[3*]

[1]*School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China*
[2]*Kingsoft Office, Wuhan 430075, China*
[3]*School of Software Engineering, Huazhong University of Science and Technology, Wuhan 430074, China*

Appendix A MonkeyDoc dataset

MonkeyDoc is designed to cover a complete range of document parsing tasks, including layout detection, reading order prediction, text recognition, table recognition, formula recognition, and code block recognition. As shown in Fig. A1, MonkeyDoc spans more than ten diverse document domains and provides high-quality annotations in both Chinese and English, making it a comprehensive and versatile dataset for document parsing. In comparison, previous datasets are often limited to a subset of tasks, a single document type, or monolingual settings. MonkeyDoc uniquely enables multi-task, multi-domain, and bilingual training and evaluation, supporting both fine-grained and holistic document understanding. In the following sections, we provide a detailed description of the data construction process for each stage.

Appendix A.1 Structure detection

The task of structure detection involves detecting elements such as text, tables, and images within a document and returning their corresponding categories and bounding box coordinates. To construct the structure detection dataset, we focus on two main aspects:

Filtering from open-source datasets. We collect existing open-source structure detection datasets, including M^{6}Doc [1], DocLayNet [2], D4LA [3], and CDLA[1)], covering both Chinese and English documents, with a total of 153k pages. We process and filter the original annotations by removing nested bounding boxes and retaining only the largest bounding box along with its category and coordinates. Additionally, we compute the proportion of each bounding box's area relative to the total page area and remove low-information instances where this ratio is less than 35%. To address the inconsistency in category annotations across different datasets (e.g., M^{6}Doc [1] defines 74 categories while D4LA [3] uses 27), we map all category labels to a unified set of classes.

Synthesizing Chinese data. Given the scarcity of Chinese samples in existing open-source datasets, we further collected over 300k pages of Chinese documents spanning financial reports, textbooks, academic papers, and more. These pages were first pre-annotated using an existing structure detection model, followed by rule-based post-processing, such as removing nested bounding boxes and filtering low-information instances, and subsequent manual correction and verification to ensure annotation accuracy and consistency. In total, these steps yield 150k high-quality samples.

Appendix A.2 Content recognition

Content recognition includes text block recognition, table recognition, formula recognition, and code block recognition. To construct data for these tasks, we focus on four key components:

Cropping document elements. Leveraging the layouts obtained from structure detection, we extracted various types of document elements from the images, yielding a total of 2.5 million cropped samples. Text blocks, formula blocks, and tables were annotated using the advanced model Qwen2.5-VL-72B [4] as well as the commercial tool Gemini 2.5 Pro [5].

Filtering from open-source datasets. For table recognition, we filter data from PubTabNet [6] through a series of quality checks, including verifying HTML tag closure, ensuring table structure completeness (e.g., presence of headers), validating the correctness of merged cells, checking header-body alignment, detecting abnormal encoded characters, and performing overall syntax validation. This results in a clean dataset of 470k tables. For formula recognition, we use the UniMER-1M [7] dataset as the training set, which is constructed from diverse sources, including public datasets such as Pix2tex[2)], CROHME [8–10] and HME100K [11], as well as LaTeX expressions collected from ArXiv, Wikipedia and StackExchange. The dataset includes printed, handwritten and screen-captured formulas, ensuring broad coverage of expression styles and complexities. In total, it contains approximately 1.5 million samples.

Synthesizing Chinese data. Due to the limited availability of Chinese data for table and formula recognition, we manually synthesize additional datasets. For tables, we collect numerous existing table structures featuring diverse multi-row, multi-column, and

* Corresponding author (email: ylliu@hust.edu.cn, xbai@hust.edu.cn)
1) https://github.com/buptlihang/CDLA
2) https://github.com/lukas-blecher/LaTeX-OCR

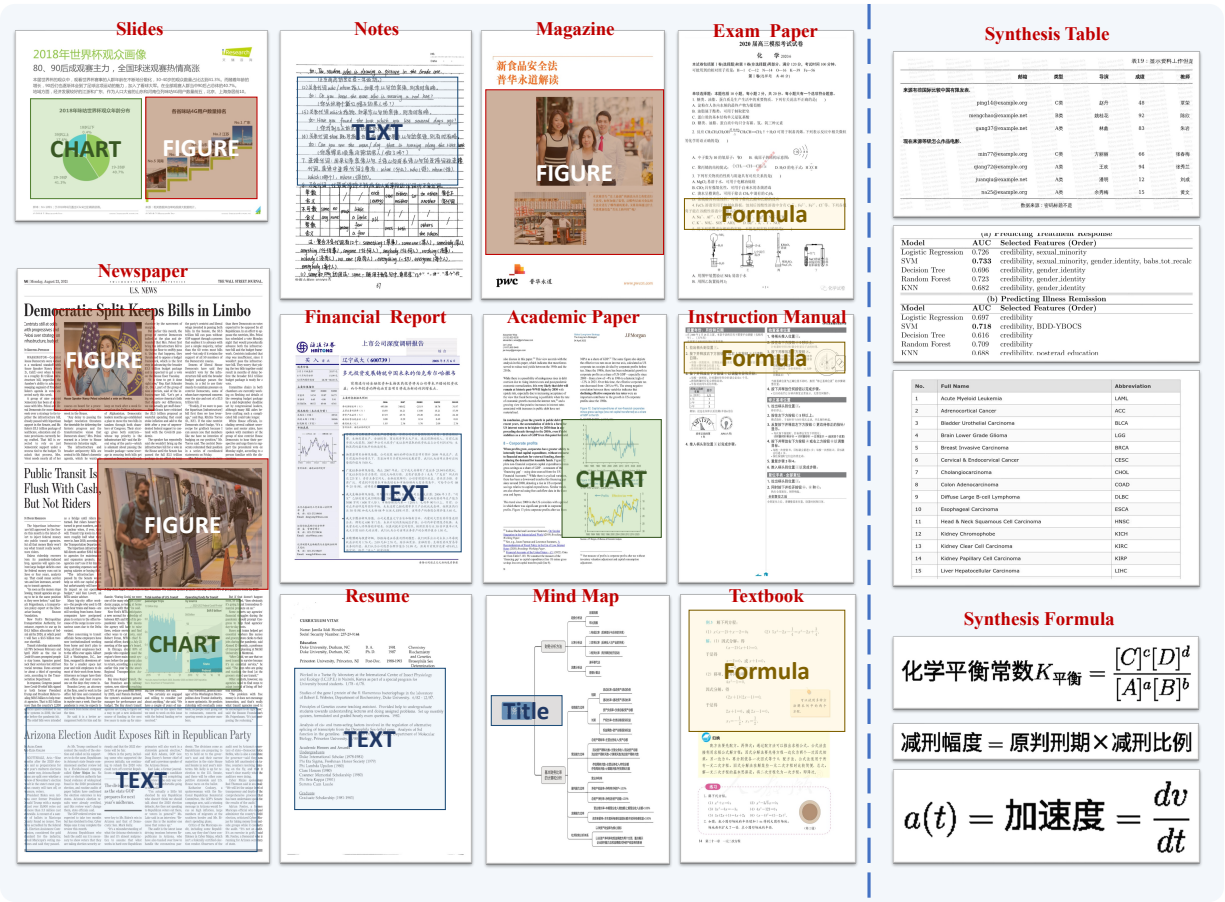


Figure A1 Visualization of the MonkeyDoc dataset. MonkeyDoc encompasses more than ten document types and includes our synthesized tables and formula data.

nested combinations, automatically populate table cells with Chinese characters, and render the results into Chinese table images paired with HTML annotations. For formulas, we synthesize the dataset through two approaches: first, by using large multimodal models to generate commonly used formulas in Chinese across various industries; second, by translating English formulas from the UniMER-1M [7] dataset into Chinese using large language models and rendering them into Chinese formula images. In total, the synthesized Chinese tables and formulas comprise 135k samples.

ArXiv data extraction. To further enrich the dataset, we crawl LaTeX data for tables and formulas from ArXiv. The crawled data is processed using the advanced open-source large language model Qwen2.5-72B [12] to filter out unnecessary elements such as citations and comments. Additionally, table LaTeX data is converted into HTML using these models and then rendered into images. This ensures the inclusion of high-quality academic data into the dataset, resulting in a total of 411k collected samples.

By combining these four steps, we build a comprehensive and diverse dataset for content recognition, supporting text block, table and formula identification in both Chinese and English.

Appendix A.3 Relation prediction

Based on the completed structure detection and content recognition, we further predict the reading order relationships between elements. Existing structure detection models output region-level bounding boxes, which are not directly compatible with line-level reading order prediction models. To bridge this gap, we construct a region-level reading order dataset to train models specifically for this granularity. We collect three types of data for reading order construction: open-source datasets with region-level reading order annotations, unlabeled Chinese data with diverse document types, and documents with region-level layout annotations, and design dedicated strategies for each.

Filtering from open-source datasets. To our knowledge, DocGenome [13] is the only open-source training dataset that provides region-level reading order annotations. These annotations are generated through an automated labeling process, which may introduce a certain degree of inaccuracy. To address potential inaccuracies or omissions in the reading order annotations for images and tables, we refine the annotations by strictly associating each image and table with its corresponding caption. We then filter out low-quality data, including pages with unannotated regions or excessive blank areas. After completing the correction and filtering steps, we score samples based on the diversity of element types on each page, incorporating high-scoring pages into our training set. As a result, we obtain 951k samples with high-quality reading order annotations.

Manually annotating Chinese data. Given the scarcity of Chinese documents in existing datasets, we collected a wide range of Chinese document types, including research reports, academic papers, user manuals, books, test papers, slides, official documents,

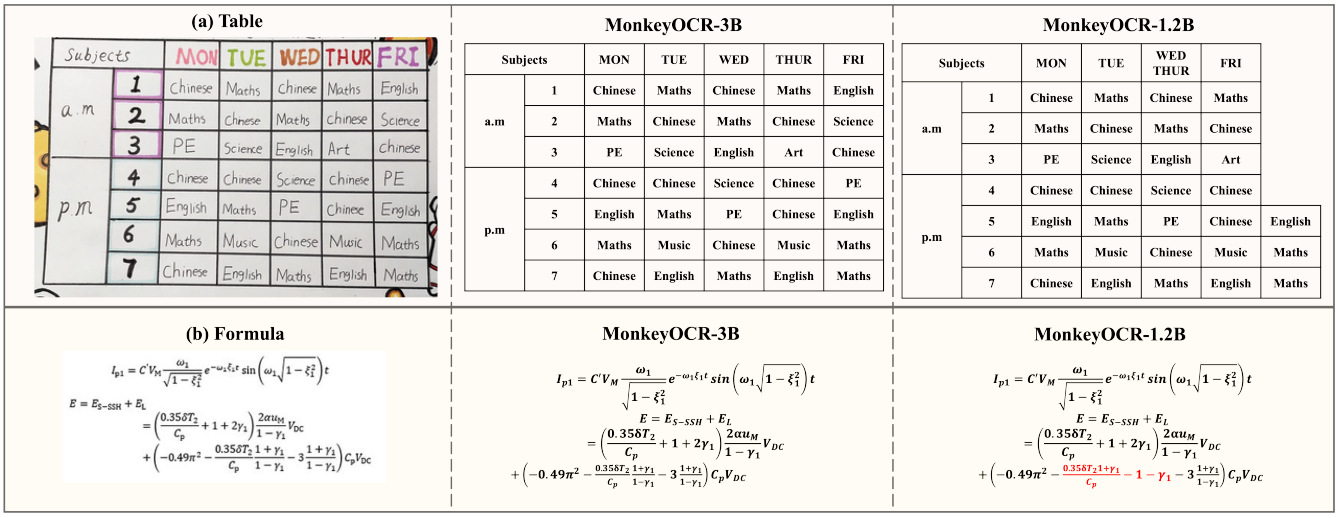


Figure C1 Typical failure cases of CPD.

newspapers and journals, as well as contracts. We manually annotated the region-level reading order of these documents, resulting in 154k samples, to enhance the coverage and diversity of Chinese scenarios in the dataset.

Automatic annotation with expert models. Existing layout analysis datasets typically provide region-level bounding box annotations but lack corresponding reading order information. To address this limitation, we leverage the bounding box annotations from these datasets and the line-level LayoutReader [14] model to automatically generate region-level reading order labels. Specifically, we first utilize PPOCR [15] to perform line-wise text recognition within each bounding box, obtaining the positional information of all text lines. Subsequently, LayoutReader predicts the reading order of these text lines. Finally, the region-level reading order is determined by calculating the average predicted order of all text lines within the region. This process produces 228k region-level reading order annotations, effectively enhancing existing layout datasets.

Table A1 The end-to-end evaluation results of different backbones on OmniDocBench.

Vision Encoder	LLM	Overall$^{\text{Edit}}\downarrow$		Text$^{\text{Edit}}\downarrow$		Formula$^{\text{Edit}}\downarrow$		Table$^{\text{TEDS}}\uparrow$		Table$^{\text{Edit}}\downarrow$		Read Order$^{\text{Edit}}\downarrow$	
		EN	*ZH*	*EN*	*ZH*	*EN*	*ZH*	*EN*	*ZH*	*EN*	*ZH*	*EN*	*ZH*
QwenViT	Qwen2.5-3B	0.118	0.151	0.043	0.077	0.247	0.321	84.2	89.5	0.118	0.093	0.066	0.113
QwenViT	Qwen3-1.7B	0.131	0.149	0.044	0.074	0.293	0.326	84.7	87.9	0.125	0.087	0.062	0.110
InternViT	Qwen3-0.6B	0.144	0.176	0.116	0.154	0.263	0.328	83.8	87.9	0.128	0.112	0.070	0.110

Appendix B MonkeyOCR with different backbones

We further analyze the performance of MonkeyOCR under different backbone configurations, as shown in Tab. A1. Specifically, we evaluate multiple encoders (such as QwenViT [4] and InternViT [16]) and LLMs of varying scales (Qwen2.5-3B [12], Qwen3-1.7B, and Qwen3-0.6B) trained on MonkeyDoc. Experimental results show that the proposed SRR paradigm and the MonkeyDoc dataset exhibit strong robustness across all backbone settings. Moreover, backbone configurations with larger parameter sizes often achieve better overall performance.

Appendix C Comparison on OmniDocBench v1.5

We further conduct comparisons with other models on OmniDocBench v1.5, which contains 1,355 document images with a more balanced distribution of document and element types. In this updated benchmark, formula recognition is evaluated using the CDM metric. As shown in Tab. C1, our model demonstrates strong performance.

Table C1 Comprehensive evaluation of document parsing on OmniDocBench v1.5.

Model	Overall$\uparrow$	Text$^{\text{Edit}}\downarrow$	Formula$^{\text{CDM}}\uparrow$	Table$^{\text{TEDS}}\uparrow$	Table$^{\text{TEDS-S}}\uparrow$	Read Order$^{\text{Edit}}\downarrow$
MonkeyOCR-3B	91.02	0.049	89.51	88.43	92.74	0.074
MinerU2.5	90.67	0.047	88.46	88.22	92.38	0.044
dots.ocr	88.41	0.048	83.22	86.78	90.62	0.053
DeepSeek-OCR	87.01	0.073	83.37	84.97	88.80	0.086

Appendix D More analysis of CPD

To further analyze the advantages of CPD, we sampled a subset of the training data to compare CPD with traditional knowledge distillation (KD). Specifically, we initialized the LLM with selected layers from Qwen2.5-3B and used MonkeyOCR-3B as the teacher to train the 1.2B model. Since KD requires additional inference from the 3B model, CPD achieves approximately three times higher training efficiency. Under the same training setup, CPD outperforms KD by 2.17%. We also examined how the smaller 1.2B model performs across different tasks. For simple tasks, its performance drops only slightly compared to the 3B model, whereas more complex tasks such as formula recognition and table recognition suffer a larger decline. This difference is likely because text recognition only requires content transcription, while tables and formulas involve intricate structures that demand deeper understanding. Visualizing typical failure cases after pruning, we found that errors mainly occur in table and formula structures. For instance, the model misreads table cells (Fig. C1 (a)) and interprets fractions incorrectly as left–right sequences (Fig. C1 (b)).

References

1 Cheng H, Zhang P, Wu S, et al. M6doc: 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, 2023. 15138–15147.
2 Pfitzmann B, Auer C, Dolfi M, et al. Doclaynet: A large human-annotated dataset for document-layout segmentation. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2022. 3743–3751.
3 Da C, Luo C, Zheng Q, et al. Vision grid transformer for document layout analysis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023. 19462–19472.
4 Bai S, Chen K, Liu X, et al. Qwen2.5-vl technical report. 2025. ArXiv:2502.13923.
5 Comanici G, Bieber E, Schaekermann M, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. 2025. ArXiv:2507.06261.
6 Zhong X, Shafiei E, Jimeno A. Image-based table recognition: data, model, and evaluation. In: Proceedings of the European Conference on Computer Vision, 2020. 564–580.
7 Gu Z, Liang G, Wang B, et al. Unimernet: A universal network for real-world mathematical expression recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2026.
8 Mouchere H, Viard-Gaudin C, Zanibbi R, et al. Icfhr 2014 competition on recognition of on-line handwritten mathematical expressions (crohme 2014). In: Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition, 2014. 791–796.
9 Mouchère H, Viard-Gaudin C, Zanibbi R, et al. Icfhr2016 crohme: Competition on recognition of online handwritten mathematical expressions. In: Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition, 2016. 607–612.
10 Mahdavi M, Zanibbi R, Mouchere H, et al. Icdar 2019 crohme+ tfd: Competition on recognition of handwritten mathematical expressions and typeset formula detection. In: Proceedings of the International Conference on Document Analysis and Recognition, 2019. 1533–1538.
11 Yuan Y, Liu X, Dikubab W, et al. Syntax-aware network for handwritten mathematical expression recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022. 4553–4562.
12 Yang A, Yang B, Zhang B, et al. Qwen2.5 technical report. 2024. ArXiv:2412.15115.
13 Xia R, Mao S, Yan X, et al. Docgenome: An open large-scale scientific document benchmark for training and testing multi-modal large language models. 2024. ArXiv:2406.11633.
14 Wang Z, Xu Y, Cui L, et al. Layoutreader: Pre-training of text and layout for reading order detection. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, 2021. 4735–4744.
15 Li C, Liu W, Guo R, et al. Pp-ocrv3: More attempts for the improvement of ultra lightweight ocr system. 2022. ArXiv:2206.03001.
16 Chen Z, Wu J, Wang W, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024. 24185–24198.