Do VLMs Read or Rewrite?

Gwang Gook Lee, Jay Mohta, Kenan E. Ak, Dimitrios Dimitriadis, Yan Xu
Amazon.com
Seattle, WA, USA
{gglee, jaymoht, kenanea, dbdim, yanxuml}@amazon.com

Abstract

Vision Language Models (VLMs) are increasingly adopted for document understanding tasks, often replacing traditional OCR systems. However, VLMs exhibit a fundamental difference: they frequently correct or rewrite imperfect text rather than transcribe it literally, a behavior that remains largely underexplored. We present a systematic investigation through controlled experiments with intentionally perturbed text across seven models and 1,706 documents. Our evaluation reveals three distinct behavioral patterns: pipeline OCR systems maintain near-zero degradation ($\sim$ 1%) through character-by-character processing; general purpose VLMs show substantial performance drops (up to 14% F1) due to semantic-level processing that prioritizes plausibility over literal transcription; and OCR-specialized VLMs achieve intermediate performance with minimal degradation (0.2–8%). These findings expose fundamental differences in how models process text, ranging from character-level transcription to holistic semantic understanding, with important implications for model selection in applications requiring literal transcription.

1. Introduction

Vision Language Models (VLMs) [1, 5, 11], which integrate vision encoders with Large Language Models (LLMs), have demonstrated strong capabilities in document understanding, with their Optical Character Recognition (OCR) performance often matching or surpassing traditional systems. Driven by this success, VLMs are becoming convenient alternatives to traditional OCR models for various document processing applications.

However, behind this success lies a subtle yet consequential behavior: VLMs exhibit a systematic tendency to "rewrite" text rather than transcribe it literally. When encountering imperfect text in images containing typos, visual artifacts, or other ambiguities, these models often output plausible corrections rather than faithful transcriptions. For instance, a VLM presented with the typo "prbolem" might confidently return "problem". This behavior resembles the *typoglycemia* phenomenon in human reading, where people comprehend text despite scrambled letters. Unlike traditional OCR systems [7, 23], which output exactly what appears in the image, VLMs process text at a holistic level, extracting semantic meaning from imperfect inputs.

While this correction behavior can be beneficial when encountering genuine errors, it raises concerns for applications requiring literal transcription, such as legal document analysis, medical record digitization, and historical manuscript preservation. Despite its practical importance, this behavior has been largely overlooked in prior VLM research. Existing OCR benchmarks [4, 19] focus on clean text recognition and are therefore insufficient for investigating how models handle imperfect text.

In this work, we conduct a controlled perturbation study to systematically characterize text correction hallucinations in VLMs, uncovering their tendency to prioritize semantic correction over literal transcription. Through empirical analysis across major VLM architectures, we reveal important model-specific behaviors that inform practical considerations for model selection in OCR applications requiring literal accuracy.

2. Related Work

Our work draws on three areas of research: vision-language models, optical character recognition, and hallucination.

2.1. Vision Language Models

VLMs extend the capabilities of LLMs by processing visual inputs alongside text, typically comprising a pretrained vision encoder, a language backbone, and a connector module that projects visual features into the language embedding space [28]. Building on this foundation, recent architectures including LLaVA [11], InternVL [5, 25], Qwen-VL [1, 2], and commercial models such as GPT-4V [16] have shown strong capabilities across diverse visual reasoning and understanding tasks, establishing VLMs as powerful multimodal systems with broad applicability. However, while these general-purpose models can handle document-

related tasks, they prioritize broad visual understanding over domain-specific OCR capabilities.

2.2. Optical Character Recognition

OCR has evolved from traditional systems like Tesseract [22] with engineered pipelines to deep learning approaches including CRNN [21] and transformer-based architectures [3]. More recently, OCR-specialized VLMs [6, 15, 26, 27] combine visual encoders with language models, representing the current state-of-the-art for text-rich document processing. While existing benchmarks such as DocVQA [13], OCR-Bench [8, 12], and OmniDocBench [18] evaluate document understanding and text recognition, they are not designed to assess whether models faithfully transcribe text as written or silently introduce corrections.

2.3. Hallucinations in Language Models

Hallucinations, the generation of plausible but incorrect content, have been extensively studied in LLMs [9, 14]. However, text correction hallucinations in VLMs remain largely underexplored. While work on adversarial text [20] and typographic attacks [24] addresses related security vulnerabilities, the tendency of VLMs to silently correct text during transcription has not been systematically investigated, particularly for document processing applications requiring literal accuracy.

3. Experiment Setup

In this section we describe our experimental setup including our dataset construction in Section 3.1, evaluated models in Section 3.2 and finally evaluation protocol in Section 3.3.

3.1. Dataset

We construct our evaluation dataset on top of READoc [10], a benchmark for structured content extraction from documents. READoc provides high-quality ground truth annotations. To enable our controlled study, we modify READoc's Markdown files by applying targeted perturbations, render them as PDF images for model input, and use the modified Markdown as ground truth for evaluation. This approach allows us to measure whether VLMs transcribe perturbed text literally or attempt to correct it.

We introduce three types of perturbations, along with unmodified originals as controls:

- **Original**: Unmodified documents serving as baseline.
- **Random**: Each character is replaced with a random letter (e.g., "standard" $\rightarrow$ "*xkpmewqi*").
- **Scramble**: Internal characters are shuffled while preserving the first and last (e.g., "s*tandar*d" $\rightarrow$ "s*danart*d").
- **Visual**: Characters are replaced with visually similar alternatives (e.g., "st*a*ndard" $\rightarrow$ "st*sn*ard").

Our dataset comprises four variants per document (original, random, scramble, visual). Perturbations are applied only to non-stopwords with length ≥ 4 to ensure meaningful modifications. We limit documents to 10,000 characters for computational efficiency, resulting in 1,706 documents total. READoc provides two subsets (arXiv and GitHub); dataset statistics are summarized in Table 1.

Table 1. Dataset statistics

Dataset	Documents	Total Pages	Words (Avg)	Perturbed (Avg)
arXiv	891	2,016	1,361	167
GitHub	815	1,685	697	56

3.2. Models

Model Selection We evaluate eight models across three categories, enabling comparison spanning different model architectures, from general-purpose VLMs to OCR-specialized VLMs and traditional pipeline approaches.

- **General VLMs**: State-of-the-art VLMs with strong general visual understanding but no explicit OCR specialization (Qwen3-VL-4B and 235B[2]).
- **OCR-Specialized VLM**: explicitly optimized for document understanding (DeepSeek-OCR-v2 [26], GLM-OCR [17], MinerU2.5 [15], PaddleOCR-VL-1.5 [6]).
- **Pipeline Models**: Traditional pipeline-based OCR model (Tesseract [22]).

Text Extraction We employ a consistent text extraction task across all models without explicit instructions about handling perturbed text, allowing us to observe natural model behavior. Each page is rendered as an image without preprocessing and passed to the models for conversion to Markdown format.

Prompt *Please convert this document image to markdown format. Make sure not to add any text other than what is written in the document.*

3.3. Evaluation Metrics

We employ metrics built from the READoc evaluation suite.

- **Edit Distance Similarity (EDS)**: measures character-level similarity based on normalized Levenshtein distance, where 1.0 indicates a perfect match.
- **F1**: provides a balanced measure of word-level accuracy that combines precision and recall measured for individual word.

4. Experimental Results

This section presents our experimental findings on model performance and behavior patterns when encountering perturbed text.

4.1. Baseline Performance

On original, unperturbed documents, all evaluated models achieve strong performance as shown in Table 2. On arXiv documents, for example, Qwen3-VL-235B (96.66% EDS, 92.64% F1) and GLM-OCR (96.41% EDS, 91.66% F1) demonstrate top-tier accuracy. These results confirm that general-purpose VLMs achieve performance comparable to OCR-specialized models on clean text.

Table 2. Baseline performance on original documents.

	arXiv		GitHub	
Model	EDS	F1	EDS	F1
Qwen3-VL-4B	95.11	90.67	94.30	93.85
Qwen3-VL-235B	**96.66**	**92.64**	**96.58**	**95.15**
DeepSeek2-OCR	94.64	90.00	93.24	92.29
MinerU 2.5	95.90	89.74	92.86	91.94
PaddleOCR-VL-1.5	94.75	89.73	89.99	91.88
GLM-OCR	96.41	91.66	94.05	92.94
Tesseract	89.74	88.96	92.30	90.65

4.2. Performance Degradation Under Perturbation

Tables 3 and 4 present performance degradation under each perturbation type, measured as the drop in percentage points from baseline shown in Table 2. Figure 1 visualizes these drops across all models and perturbation types. The results reveal that model responses to perturbations vary widely depending on architecture and perturbation type.

Table 3. Performance drop against baseline in arXiv dataset.

	EDS			F1		
Model	Random	Scramble	Visual	Random	Scramble	Visual
Qwen3-VL-4B	−3.06	−4.05	−2.71	−9.88	−11.38	−14.18
Qwen3-VL-235B	−0.47	−2.47	−3.08	−1.91	−6.48	−12.44
DeepSeek2-OCR	−0.52	−1.79	−1.14	−4.71	−8.71	−8.49
MinerU 2.5	−0.25	−0.47	−0.96	−2.20	−3.15	−7.64
PaddleOCR-VL	+0.02	−0.14	−0.39	+0.39	−0.37	−2.90
GLM-OCR	−0.06	−0.05	−0.24	+0.52	+0.23	−1.63
Tesseract	−1.27	0.19	0.04	−0.82	+0.35	−0.18

Table 4. Performance drop against baseline in GitHub dataset.

	EDS			F1		
Model	Random	Scramble	Visual	Random	Scramble	Visual
Qwen3-VL-4B	−1.02	−1.22	−1.15	−4.50	−5.29	−7.04
Qwen3-VL-235B	−0.69	−2.86	−3.42	−1.43	−4.14	−6.96
DeepSeek2-OCR	−0.08	−0.91	−0.32	−2.25	−4.75	−4.44
MinerU 2.5	−0.25	−0.26	−0.53	−1.44	−1.71	−4.01
PaddleOCR-VL	+0.11	−0.18	−0.16	−0.15	−0.86	−2.40
GLM-OCR	−0.06	−0.21	−0.29	+0.07	−0.13	−1.21
Tesseract	−0.30	−0.07	−0.03	−0.62	−0.07	−0.55

Models exhibit three fundamentally different behaviors. **Pipeline models** (Tesseract) demonstrate remarkable robustness with drops about 1% or less, as their character-by-character processing is immune to linguistic interference. **General VLMs** (Qwen3-VL) suffer the most severe degradation, with F1 drops of 12–14% on arXiv visual perturbations, revealing strong tendencies to prioritize semantic plausibility over literal transcription. In contrast, **OCR-specialized VLMs** demonstrate substantially greater robustness. GLM-OCR and PaddleOCR maintain exceptional stability (F1 drop <2.9%) on arXiv, while others such as DeepSeek-OCR exhibit moderate degradation (4.7-8.7%), suggesting that OCR-specialized training substantially mitigates but does not entirely eliminate the hallucination behavior.

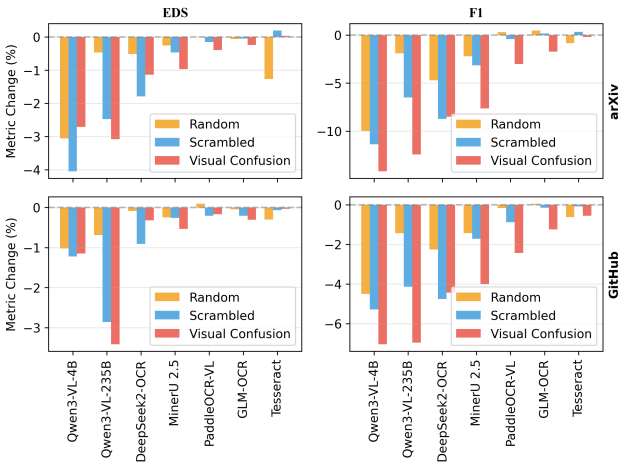

Figure 1. Performance degradation across models and perturbations.

Notably, the divergence between character-level (EDS) and word-level (F1) metrics reveals the nature of model errors. In models that tend to predict original words, F1 drops substantially exceed EDS drops. For example, Qwen3-VL-235B on arXiv visual perturbations shows 3.08% EDS drop but -12.44% F1. This disparity reflects a strong bias toward valid words from training data: when encountering perturbed text that resembles familiar words, these models actively correct toward plausible alternatives rather than transcribe literally, preserving most characters while producing lexically incorrect words and causing larger word-level degradation.

4.3. Word-Level Hallucination Behavior

Table 5 examines error patterns on perturbed words. For cases where the model fails to transcribe the perturbed form literally, we categorize the output as either recovering the original unperturbed word ("Orig.") or producing an output matching neither the original nor perturbed form ("Incorr."). Both cases represent transcription errors in which the model rewrote what appeared in the image to produce a more plausible alternative. Figure 2 visualizes the distribution of these error types across all models.

Model architectures reveal a fundamental distinction be-

Table 5. Error analysis: "Orig." represents where model predicted the original unperturbed word while "Incorr." is for predictions matching neither the original nor the perturbed form (Note that both represent transcription errors).

	arXiv						GitHub					
	Scramble		Visual		Random		Scramble		Visual		Random	
Model	Orig.	Incorr.	Orig.	Incorr.	Orig.	Incorr.	Orig.	Incorr.	Orig.	Incorr.	Orig.	Incorr.
Qwen3-VL-4B	54.14	45.86	64.44	35.56	0.36	99.64	48.92	51.08	58.46	41.54	0.28	99.72
Qwen3-VL-235B	69.04	30.96	80.16	19.84	0.66	99.34	67.57	32.43	74.08	25.92	1.01	98.99
DeepSeek-OCR2	44.93	55.07	37.87	62.13	0.07	99.93	53.23	46.77	43.65	56.35	0.05	99.95
MinerU 2.5	18.42	81.58	49.56	50.44	0.11	99.89	18.12	81.88	47.94	52.06	0.04	99.96
PaddleOCR-VL	33.05	66.95	62.75	37.25	0.43	99.57	37.01	62.99	61.04	38.96	0.24	99.76
GLM-OCR	25.04	74.96	66.33	33.67	0.41	99.59	34.88	65.12	71.61	28.39	1.62	98.38
Tesseract	0.00	100.00	14.72	85.28	0.00	99.99	0.25	99.75	16.15	83.85	0.33	99.67

tween pipeline models and VLMs. As shown in Table 5, **Pipeline model** (Tesseract) always predicts the perturbed words correctly for scrambled and random inputs, confirming that it can transcribe character-by-character without linguistic interpretation. In contrast, **both general and OCR-specialized VLMs** exhibit similar hallucination behaviors despite their different training objectives. Both categories demonstrate a strong tendency to output semantically plausible words rather than literal transcriptions. This suggests that plausibility-based rewriting is a fundamental characteristic of VLM architectures, regardless of OCR-specific training. The key difference lies in the magnitude of performance degradation rather than the behavioral pattern itself, as established in Section 4.2.

Examining the word-level errors patterns across perturbation types in Table 5 reveals a clear ordering in how frequently models hallucinate (predict) the original unperturbed word: **visual** (37–71% for VLMs) > **scramble** (18–54%) > **random** (<4%). For instance, GLM-OCR predicts the original word 66% of the time on arXiv visual perturbations, dropping to 25% with scramble and vanishing to <1% with random. Similarly, Qwen3-VL shows 64% original word prediction for visual, 54% for scramble, and <1% for random. This ordering directly reflects how VLMs process text holistically: visual perturbations (e.g., 'a'→'s') preserve sufficient word shape and contextual cues to trigger this rewriting behavior, scrambled letters partially disrupt word recognizability while maintaining some gestalt, and random character substitutions eliminate all lexical patterns, forcing character-level processing that VLMs struggle to perform accurately.

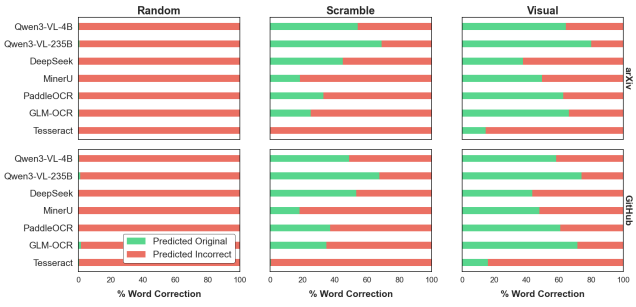

Figure 2. Percentages where model predicted the original word (green) vs. failed to read perturbed word correctly (red). Note that both are OCR failures.

5. Conclusion

This work systematically investigates how VLMs handle OCR tasks involving imperfect text. Across seven models, we find that, unlike traditional OCR systems, VLMs prioritize plausible corrections over literal transcription when encountering imperfect text, even when performing well on standard benchmarks.

These findings have immediate practical implications. For applications requiring literal transcription, such as legal document processing, historical manuscript digitization, and forensic analysis, traditional pipeline systems or carefully selected OCR-specialized VLMs remain the safest choice. General-purpose VLMs, while powerful for semantic understanding, may introduce systematic biases when exact character reproduction is required. This highlights a broader tension in VLM development: the same holistic processing that enables strong semantic understanding can become a liability in tasks requiring literal output.

Our study has several limitations. We focus on English text in document contexts; behavior may differ significantly for other languages, scripts, or specialized domains. The perturbation types evaluated, while informative, represent only a subset of real-world text degradations. An important direction for future research is evaluating how this behavior impacts downstream task performance in real-world applications where literal transcription is critical. While our work establishes that the behavior exists systematically, understanding its practical consequences requires task-specific evaluation. Finally, our analysis focuses on zero-shot behavior with minimal prompting; systematic investigation of strategies to mitigate such behavior remains an important direction for future work.

References

[1] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023. 1

[2] Shuai Bai, Yuxuan Cai, Ruizhe Chen, and *et al*. Qwen3-vl technical report. *ArXiv*, abs/2511.21631, 2025. 1, 2

[3] Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. Nougat: Neural optical understanding for academic documents, 2023. 2

[4] Song Chen, Xinyu Guo, Yadong Li, Tao Zhang, Mingan Lin, Dongdong Kuang, Youwei Zhang, Lingfeng Ming, Fengyu Zhang, Yuran Wang, et al. Ocean-ocr: Towards general ocr application via a vision-language model. *arXiv preprint arXiv:2501.15558*, 2025. 1

[5] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks, 2024. 1

[6] Cheng Cui, Ting Sun, Suyin Liang, Tingquan Gao, Zelun Zhang, Jiaxuan Liu, Xueqing Wang, Changda Zhou, Hongen Liu, Manhui Lin, Yue Zhang, Yubo Zhang, Yi Liu, Dianhai Yu, and Yanjun Ma. Paddleocr-vl-1.5: Towards a multi-task 0.9b vlm for robust in-the-wild document parsing, 2026. 2

[7] Yuning Du, Chenxia Li, Ruoyu Guo, Xiaoting Yin, Weiwei Liu, Jun Zhou, Yifan Bai, Zilin Yu, Yehua Yang, Qingqing Dang, et al. Pp-ocr: A practical ultra lightweight ocr system. *arXiv preprint arXiv:2009.09941*, 2020. 1

[8] Ling Fu, Zhebin Kuang, Jiajun Song, Mingxin Huang, Biao Yang, Yuzhe Li, Linghao Zhu, Qidi Luo, Xinyu Wang, Hao Lu, Zhang Li, Guozhi Tang, Bin Shan, Chunhui Lin, Qi Liu, Binghong Wu, Hao Feng, Hao Liu, Can Huang, Jingqun Tang, Wei Chen, Lianwen Jin, Yuliang Liu, and Xiang Bai. Ocrbench v2: An improved benchmark for evaluating large multimodal models on visual text localization and reasoning, 2025. 2

[9] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023. 2

[10] Zichao Li, Aizier Abulaiti, Yaojie Lu, Xuanang Chen, Jia Zheng, Hongyu Lin, Xianpei Han, and Le Sun. Readoc: A unified benchmark for realistic document structured extraction. *arXiv preprint arXiv:2409.05137*, 2024. 2

[11] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, pages 34892–34916. Curran Associates, Inc., 2023. 1

[12] Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 67(12), 2024. 2

[13] Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. Docvqa: A dataset for vqa on document images, 2021. 2

[14] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization, 2020. 2

[15] Junbo Niu, Zheng Liu, Zhuangcheng Gu, and *et al*. Mineru2.5: A decoupled vision-language model for efficient high-resolution document parsing, 2025. 2

[16] OpenAI, Josh Achiam, Steven Adler, and *et al*. Gpt-4 technical report, 2024. 1

[17] ZAI Organization. Glm-ocr: Open-source ocr tool. https://github.com/zai-org/GLM-OCR, 2024. Accessed: 2026-02-10. 2

[18] Linke Ouyang, Yuan Qu, Hongbin Zhou, Jiawei Zhu, Rui Zhang, Qunshu Lin, Bin Wang, Zhiyuan Zhao, Man Jiang, Xiaomeng Zhao, Jin Shi, Fan Wu, Pei Chu, Minghao Liu, Zhenxiang Li, Chao Xu, Bo Zhang, Botian Shi, Zhongying Tu, and Conghui He. Omnidocbench: Benchmarking diverse pdf document parsing with comprehensive annotations, 2024. 2

[19] Jake Poznanski, Aman Rangapur, Jon Borchardt, Jason Dunkelberger, Regan Huff, Daniel Lin, Christopher Wilhelm, Kyle Lo, and Luca Soldaini. olmocr: Unlocking trillions of tokens in pdfs with vision language models. *arXiv preprint arXiv:2502.18443*, 2025. 1

[20] Erfan Shayegani, Md Abdullah Al Mamun, Yu Fu, Pedram Zaree, Yue Dong, and Nael Abu-Ghazaleh. Survey of vulnerabilities in large language models revealed by adversarial attacks, 2023. 2

[21] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2298–2304, 2016. 2

[22] Ray Smith. An overview of the tesseract ocr engine. In *Ninth international conference on document analysis and recognition (ICDAR 2007)*, pages 629–633. IEEE, 2007. 2

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

[24] Hanzhang Wang and Qingyuan Ma. Textural or textual: How vision-language models read text in images. In *Forty-second International Conference on Machine Learning*, 2025. 2

[25] Weiyun Wang, Zhangwei Gao, Lixin Gu, and *et al*. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency, 2025. 1

[26] Haoran Wei, Yaofeng Sun, and Yukun Li. Deepseek-ocr 2: Visual causal flow, 2026. 2

[27] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, page 1192–1200. ACM, 2020. 2

[28] Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. A survey on multimodal large language models. *National Science Review*, 11(12): nwae403, 2024. 1