Ultra-Low-Cost Hybrid OCR–LLM Architecture for Production Grade E-KTP Extraction

Anjar Tiyo Saputro[1]*, Bambang Agus Herlambang[2], Mega Novita[3]
[1,2,3]Department of Informatics, Faculty of Technology and Informatics, Universitas PGRI Semarang, Indonesia

Abstract.
Purpose: The purpose of this study is to be able to avoid limitations of inexpensive ID card data extraction services and preserve privacy, which can simultaneously achieve reliable operation even under an environment with minimum infrastructure, in particular if no dependency on GPU-based servers are required.
Method: The proposed approach is a microservice pipeline with three stages: (1) local lightweight pre-processing on devices, (2) Tesseract CPU-based OCR. js, (3) fast text tokenization through a small premature external LLM. The system is developed as TypeScript backend utilizing the Hono framework with all image processing taking place locally in order to keeping user data private.
Result: The result of the experimental evaluations with real ID card samples is that the system can run stably in low-performance VPS (1 vCPU, 1 GB RAM) with operation cost approximately IDR 2.5047 per extraction process and its accuracy level is acceptable for use in a production environment. Moreover, the results indicate that system latency is dominated by LLM inference at the cloud.
Novelty: The main contribution and novelty of this study is that we demonstrate, for the first time, a cost-effective (privacy-preserving) OCR-LLM hybrid pipeline without demanding expensive GPU models at large scale which makes our system suitable under limited storage and resource constraints on-premises or edge environments in small organizations including micro-SaaS services.

Keywords: e-KTP Extraction, Optical character recognition, Large language model, Microservices, Privacy-Preserving System
Received December 2025 / **Revised** December 2025 / **Accepted** December 2025

This work is licensed under a Creative Commons Attribution 4.0 International License.

INTRODUCTION

From another perspective, given the exponential growth of digital onboarding systems in Indonesia, the need for a trustworthy and automated e-KTP data extraction system has drastically increased [1], [2]. In actual production settings, for example, small organizations and startups/micro-SaaS services, accuracy requirements are imposed not only by the precision of the system but also by the number of resources and the level of operational cost, latency, or data privacy budget. These limitations are particularly stringent when dealing with e-KTP data since they contain sensitive personal information that should adhere to strong data-sovereignty and privacy requirements.

In the case of e-KTP and document text extraction, early research has, in most cases, assumed classical OCR pipelines. 2 Traditional OCR-Based Techniques The rule-based image preprocessing and recognition have been explored extensively, for it is computationally inexpensive and easy to implement [3]. Some previous work has shown improvements when applying Tesseract OCR to adaptive preprocessing, such as binarisation, noise reduction, and skew correction, in order to improve recognition accuracy of scanned or photographed documents [4], [5]. Despite that, such methods work well under control environment, their performance significantly drops when handling real webcam or mobile capture scenarios in the presence of lighting variations, motion blur, and perspective distortion.

More recent research has investigated deep learning-based OCR improvements, such as object-detection–aided pipelines like YOLOv5–OCR hybrids implemented on embedded devices [6]. It is even higher than the computational cost of previous methods, which have enhanced text localization and recognition

*Anjar Tiyo Saputro.
Email addresses: 23676001@upgris.ac.id, (Saputro), bambangherlambang@upgris.ac.id (Herlambang), novita@upgris.ac.id (Novita)
DOI: 10.15294/sji.v12i4.38200

robustness as well, but are not deployable on inexpensive servers or basic VPS. Mobile OCR verification tools Mobile-based OCR validation tools [7], have been suggested for enhancing document verification workflows; however, they are often designed using cloud-based backends or proprietary APIs which increases operational expenses and decrease transparency/control.

Although classical OCR has its semantic and language superficiality, making it unable to cater for advanced document understanding tasks, a new family of multimodal Large Language Models (LLMs) is recently introduced. Such models show promising results in extracting structured information from incorrect OCR results, especially for more complex and diverse fields like addresses, occupations, or administrative codes [8]. However, despite their impressive semantic abilities, LLM-based solutions suffer from several key limitations to be used for industrial use. Existing works found that end-to-end latency is even higher since the OCR needs to be performed remotely [9]–[11], there are dangers of hallucinated or fabricated predictions when OCR input is not complete or ambiguous [12], and inference costs can increase dramatically as throughput grows poorly for high-volume demand [13]. Beyond that, sending raw e-KTP images and extracted text away to third-party LLM endpoints implies very serious privacy and data sovereignty issues in some jurisdictions that mandate local processing and storage [14].

Hybrid OCR–LLM pipelines have been investigated in the wider literature on document intelligence, as an attempt to find a compromise between efficiency and semantic robustness [15], [16]. These cases show that the use of fast, lightweight OCR followed by processing with language models has positive effects in terms of extraction quality. However, current mixed solutions are primarily aimed at general-purpose document computation and do not fully exploit the strict limitations of e-KTP extraction; i.e., ultra-low cost of operation, extremely tokenized usage of LLM, strongest privacy guarantees, and deployability even on low-end CPU-only machines. Consequently, there is still an outstanding research gap that no existing work has offered that provides a practical cost-bounded privacy-preserving e-KTP extraction architecture feasible with limited resources in real production.

This study fills that gap by presenting a novel, production-oriented architecture that combines lightweight Tesseract-based OCR with an efficient (in terms of token counts) refinement via LLM stage, delivered as an ultra-low-resource microservice in the Hono framework. Unlike existing methods, we process all the images and OCR operations locally to protect privacy, and impose tight bounds on LLM usage for text-level refinement only by consuming a fraction of tokens in a one-shot fashion. To the best of our knowledge, this is the first documented e-KTP extraction system that achieves production-level accuracy for lower than IDR 3 per extraction using a low-end VPS (1 vCPU, 1 GB RAM). The architecture that we propose, with its modularity, replicability, and opportunity of on-premise or edge deployment, is particularly relevant for small institutions and cost-saving digital identity services.

METHODS

The proposed approach includes a three-stage pipeline, which consists of dataset preparation to containerized deployment. We created a representative e-KTP data set by combining both the synthetic and real samples collected on Ro-boflow with anonymized production data from an e-KTP application of the authors. This combined dataset covers a variety of variations present in the real-world, such as lighting differences, compression noise, and device deformations, thereby guaranteeing robustness in varying capture conditions.

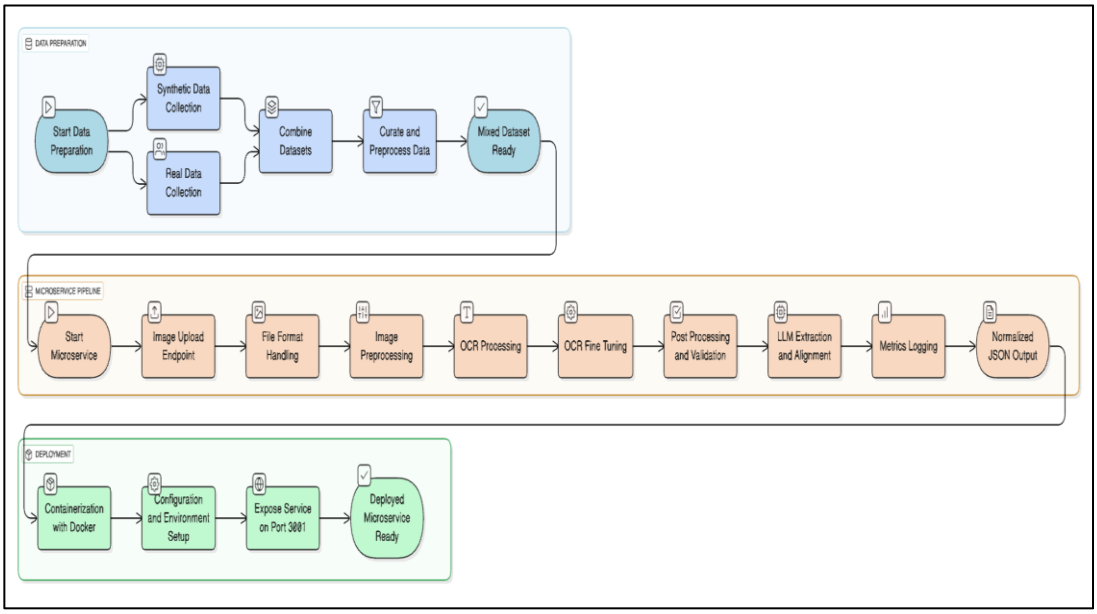

Figure 1. Pipeline Ultra-Low-Cost e-KTP Extraction

Data Preparation

For robustness in the real-world scenarios, a mixed e-KTP dataset which hybridized synthetic and real samples was employed. Roboflow curated and generated synthetic e-KTP data representing layout variation, resolution variety, lighting range, and compression noise in the real-world production pipeline. Additionally, we employed anonymized e-KTP real images that have been collected from authors' production app to maintain privacy and ethical consideration by blurring the text on them so PII is not recognizable.

The aggregated dataset spans a variety of acquisition settings, including webcam capture, and mobile phone photo along with compressed upload, representing realistic onboarding workflows. Such diversity is necessary for characterizing system stability, rather than just peak OCR accuracy under optimal conditions. No inference-time data augmentation other than and none of heavy image processing like intensity normalization or semantic refinement in later pipeline stages were employed.

Microservices Pipeline

The architecture presented in this article is implemented as a lightweight, production-ready microservice with the Hono web framework on the Bun JavaScript runtime and written using TypeScript to provide static type safety and maintainability. The service has one REST endpoint, which is POST /upload-ktp accepts file uploads in multipart/form-data. It supports image formats like JPEG, PNG, WebP, and HEIC. Stage in the microservices pipeline section, which are: (1) Image Preprocessing: Images undergo processing locally once uploaded by using the Sharp library. Preprocessings also consist of grayscale transformation, contrast normalization, sharpening, and rescaling to a maximum side length of 2500 pixels [17], [18]. These operations are designed to be lightweight in order to maintain OCR readability, as well as low CPU and memory usage. No image patterns of the sensitive e-KTP are sent out from the system; all processing images are done on-premise. (2) OCR Processing: Tesseract is used for the extraction of text. Js operating in CPU-only mode. Several targeted optimizations are performed to tune the OCR engine for Indonesian e-KTP characteristics, such as defining a custom character whitelist, a custom dictionary, and post-processing rules for handling common digit–letter confusions (e.g., “0/O”, “1/I”). Some of the field-level constraints (e.g., NIK length, gender values, and location name pattern) are further enforced by additional validation routines. The result of this stage is a raw, structured e-KTP text [19]. (3) Token-Efficient LLM Refinement: The OCR result is further refined with a text-only Large Language Model (LLM) [20] for semantic normalization into a predetermined JSON format. The pipeline is model-agnostic, but here we used DeepSeek because of its cheap inference. Importantly, the LLM operates only on OCR text data, and no image information is communicated to ensure strong privacy protection. Design embodiment is optimal in the sense that the fewest number of tokens are used to capture schema matching and OCR errors. The performance is reported in the number of tokens consumed and per-stage latency.

Deployment

The entire microservice runs in Docker container and a docker file, docker compose is used to define how it should be deployed. yml [21]–[23]. Environment values (like LLM API key) also added at runtime securely. The service runs on the default 3001 port and is deployable uniformly in disparate deployments. Experiments were conducted on a small VPS (1 vCPU, 1 GB RAM) without GPU support. The design of such a setup is a mockery of real-world constraints for small organisations and smaller SaaS vendors [24]. Container-level resource monitoring was used to capture the CPU and memory usage for both Kubernetes and JAR, in addition to end-to-end extracted latency over repeated extraction requests. Thanks to the containerized architecture, 3D-VQ-VAE enables easy production deployment portability, reproducibility and scalability with consistent run-time performance across diverse deployment setups.

RESULT AND DISCUSSION

The objective of the test was to verify if was possible to run a hybrid OCR–LLM microservice with real production limits, notably in environment where CPU resources are low. Such evaluation is critically important in deployment scenarios, like public-sector services, fintech onboarding platforms or micro-SaaS applications, where maintaining large-scale GPU infrastructure is uneconomical and/or operationally difficult.

Performance and Resource Overhead

The experiment of evaluating the system-by-system performance, namely latency distribution, token usage and resource consumption was the first test. A set of 1,000 unique e-KTP extraction requests were deployed on a containerized environment in CPU-only Virtual Private Server (VPS) (1 vCPU, 1 GB RAM).

Summary of the respective image preprocessing time (in ms} using Sharp is presented in Tab. 1. The image pre-processing step performed by the Sharp composite operation leads to a very low amount of processing time on all tested devices with an average execution time of 153 ms, showing that applied preprocessing sequences do not lead to a significant computational overhead. The OCR process is powered by Tesseract. js, leads to a higher average processing latency of 3,209 ms because character recognition in high-resolution identity documents takes longer.

Table 1. System Performance and Resource Utilization (Breakdown) for 1000 cycles

Component (unit)	Mean	Std	Min	Max
Total Latency (ms)	**8,665**	-	-	-
Pre-Processing (ms)	153	44	55	350
OCR (ms)	3,209	862	1,498	4,580
LLM Parsing (ms)	5,225	636	4,221	10,029
Total Tokens	**974**			
Prompt Tokens	855	36	805	893
Completion Tokens	119	14	103	141
Server Usage				
CPU Usage (%)	0.14	0.01	0.13	0.15
Memory Usage (MB)	42.08	6.27	36.41	63.77

The LLM-based parsing stage is the latency bottleneck, taking an average of 5225 ms to complete with some spikes over 10 s. This result confirms that the bottleneck is not in local computation but in system latency dominated by remote LLM encoding. But end-to-end latency time of 8,665 ms is not ideal for asynchronous onboarding streams and batch identity verification.

It's a system that's downright unobtrusive for how many resources it gobbles. Average CPU utilization remains at 0.14% per one vCPU, and memory usage has increased to around 42.08 MB, which is approximately 4 % of the RAM allocated. These findings confirm that the proposed framework is suitable for running on low-end systems without causing resource depletion.

Token Efficiency and Cost Analysis

In Figure 2 which illustrates the illusion of stable usage when issuing repeated requests. On 1,000 extraction tasks, the system consistently maintains a small token footprint: Avg of 855 prompt tokens and 119 completion tokens total to about 974 tokens per request.

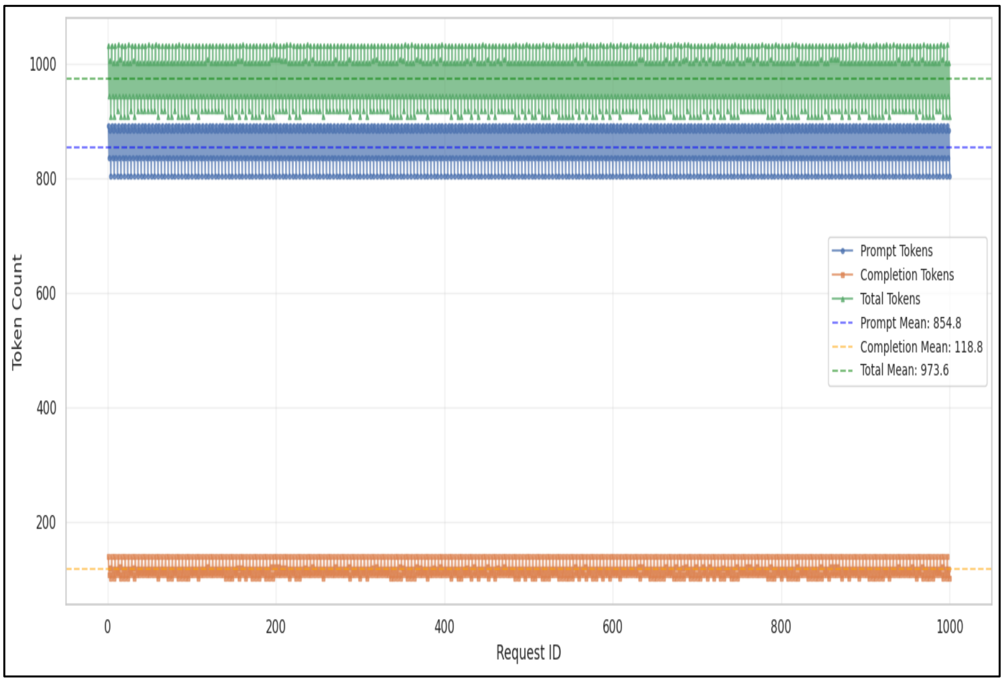

Figure 2. Token Usage Stability Across 1000 Requests

At DeepSeek-Chat rates, this token consumption is equivalent to an average inference cost of approximately that is the accumulated funds to use the system (USD 1.5 × 10^{-4} or IDR 2.50) per retrieval. This shows that semantic sophistication from a small LLM incurs very little financial cost, while significantly improving the translation quality and encouraging schema alignment. This study provide the cost fraction and the cost per request in Eqs (1) and (2).

$$Cost\ fraction\ =\left(\frac{CPU\ Quota\ Used}{CPU\ Quota\ Allcoated}\right)\times\left(\frac{Memory\ Consumption}{Memory\ Limit}\right) \tag{1}$$

$$Cost\ per\ Request\ =\ Monthly\ Cost\ \times\ Cost\ Fraction\ \times\frac{latency}{2{,}592{,}000} \tag{2}$$

To understand operational efficiency in a more detailed way, cloud resource utilization was analyzed at the container level by aggregating the CPU and memory-time cost of requests. Using average latency of 8.59 seconds, CPU utilization of 0.144%, and memory usage of 42 MB, we estimate the server-side cost to process an extraction is around IDR0.0047 per extraction, with a VPS pricing at IDR265K/month as a reference point. This value is low in comparison to the LLM inference cost, which shows that infrastructure is not a bottleneck of scalability for this system.

Comparative Analysis with Existing Approaches

Comparative study with available e-KTP extractiontechnologies are given in Table 2. Vision-based pipelines that use object detection models like YOLOv5 with OCR engines can achieve the highest extraction accuracy, but these systems require GPU acceleration, resulting in high latency and infrastructure costs, making them unsuitable for cost-constrained deployments [6]. State-of-the-art semantic comprehension can be provided by multimodal LLM-based solutions at the expense of prohibitively high per-request costs and full image transmission to external cloud services, eliciting serious privacy and data-sovereignty issues [25].

Table 2. Comparative Analysis with the existing method

Method	Metrics			
	Accuracy	Latency	Cost per Request	Deployment Req.
YOLO V5 + OCR [6]	High	High	Medium	GPU
Multimodal LLM [25]	Very High	Very High	High	Cloud GPU, external data transfer
Classical Tesseract Only [4]	Low	Low	Ultra Low	CPU
Hybrid Tesseract + Lightweight LLM	High	Medium	Ultra Low	CPU, Fully local inference pipeline

By contrast, classical OCR-only pipeline like runing on Tesseract could only achieve ultra-low cost with low robustness in real e-KTP images captured under an uncontrolled setting [4]. Such a hybrid architecture gives an optimal trade-off between high accuracy, timely response rate and ultra-low cost, working on CPU-only servers without causing any data to leave the local network.

This difference emphasizes the main advantage of our approach: we succeed in bridging between cheaper classical OCRs and expensive but effective multimodal LLM systems. Restricting LLM usages to token-efficient text-only fine-tuning and processing images on premise, the design achieves production-level operable performance free of privacy vulnerabilities or economic cost.

Discussion and Implications

The experimental results offer several significant implications for designing a more cost-effective, privacy-preserving identity extraction system. First, the observed distribution of latencies gives evidence that local OCR and preprocessing are no longer performance bottlenecks in nowadays document extraction pipelines. Preprocessing and OCR stages account for less than 40% end-to-end latency, even when we run purely on CPU. This result runs contrary to the common intuition that GPU acceleration should be required for quality document extraction, and shows that when teamed with a targeted post-processing pipeline, a well-optimized classical OCR engine is very much fit for production use.

Second, the dominance of the LLM inference stage over total latency can be indicative of an important design choice. However, LLMs do enhance semantic robustness, particularly for complex KTP fields such as an address or administrative attributes, and their remote inference will result in (inevitable) network delays and computation latencies. Nevertheless, by limiting LLM utilization to only text-level fine-tuning, the proposed model reduces token usage and circumvents transmitting confidential image information. It balances semantic accuracy against privacy and operational cost with this design choice, putting the system between pure OCR and a completely multimodal LLM.

There are several important implications based on the experimental results for us to design a cheaper and privacy-preserving identity extraction system. First, the distribution of latencies strongly indicates that local OCR as well as preprocessing do not circumscribe web-scale document extraction pipelines anymore.

The comparison results demonstrate the engineering significance of the proposed structure. GPU based vision pipelines are providing huge accuracy, but with overhead in deployment and maintenance. Despite its powerfulness, the multimodal LLM-based approaches introduce big per-request overheads and privacy issues due to the inherent cloud image processing. Contrastingly, the hybrid arrangement we present here illustrates that production-level accuracy can be achieved without compromising data sovereignty and hence may have a specific place in regulatory domains such as are imposed by regulations restricting cross-border data movement and dependence upon cloud services.

The other one is for the stability of the system. Steady Use of Tokens: Steady token consumption for 1,000 requests demonstrates that the semantic fine-tuning phase behaves predictably, hopefully reducing concerns about unexpected cost or noisy output associated with unconstrained LLM prompts. This predictability is important not just for production environments, but also because budget and SLAs depend on known system behavior including monitoring alerts.

However, the proposed model also has several limitations that should be discussed. First, the latency remains acceptable for asynchronous and batch checks, but may be unacceptable if a real-time or interactive authenticity verification is needed. Second, relying on an external LLM provider results in dependence to ongoing third-party availability and pricing maintenance. While this dependency is weaker than for multimodal pipelines, it remains problematic for the purposes of long-term deployment.

These limitations point to specific avenues for future work. By replacing the cloud-based LLM inference with a lighter local language model, its latency may be reduced while being independent of off-device resources at the cost of increased on-device resource consumption. Furthermore, adaptive prompt compression and caching techniques can reduce token consumption on similar extraction patterns. Lastly, comprehensive stress testing with workloads in parallel is required to estimate throughput bounds and find optimal scaling approaches.

In conclusion, the extended discussion demonstrates that the hybrid OCR–LLM microservice is not a technical and theoretical proof of concept but an implementation-ready solution that can be deployed within real-world budget, privacy, and infrastructure concerns. The results add to a growing body of evidence that selecting and token-efficiently combining LLMs (as opposed to performing full end-to-end multimodal inference) is a viable path forward for document intelligence systems under low-resource settings.

CONCLUSION

In this study, we have shown that the hybrid OCR–LLM microservice can be operated in production quality with high-reliability levels and very low computational overhead—CPU-only execution—and an ultra-low operational cost of around IDR 2.50 per request (the price is not stated if created with AWS), making it suitable for e-KTP extraction at scale and respecting privacy. The results also show that total latency is mainly bounded by the DeepSeek LLM stage, which indicates that deployments in the future could benefit from faster providers such as the Azure GPT-4o Mini or other Lightweight inference APIs. For security, the design can be hardened by moving to a fully on-premise setup with smaller local LLMs in order to open up no cloud needs for semantic parsing. Being a deployable microservice, the system already has production-like characteristics and can be deployed on small VPS instances in the micro-SaaS niche with just some commodity features (such as authentication and load-balancing). Although we will need to conduct load testing to measure maximal throughput, the analysis indicates that the bottleneck for scalability will be cloud LLM inference (not local computation).

REFERENCES

[1] R. A. Prasojo, T. Yuniningsih, and R. Hidayati, “Innovation of Digital Population Identity Application at the Population and Civil Registration Service of Semarang City,” *PERSPEKTIF*, vol. 14, no. 2, pp. 297–306, 2025.

[2] I. Zulkarnaen, “Service Innovation for Electronic Identity Cards Based on Digital Population Identity Applications at The Cirebon Regency Population and Civil Registration Office,” *Int. J. Bus. Appl. Econ.*, vol. 4, no. 3, pp. 1023–1036, 2025.

[3] Y. Afifah, A. Sujono, and C. H. B. Apribowo, “The line segmentation algorithm of Indonesian electronic identity card (e-KTP) for data digitization,” in *AIP Conference Proceedings*, 2020, vol. 2217, no. 1, p. 30138.

[4] K. Nisha, T. Wahyuni, and M. A. M. Hayat, “Pemeriksaan KTP Menggunakan Optical Character Recognition (OCR) dan Pengenalan Background serta Komponen KTP,” *Arus J. Sains dan Teknol.*, vol. 2, no. 2, pp. 490–495, 2024.

[5] H. Holila, A. R. Pratama, S. A. P. Lestari, and J. Indra, “Introduction National Identification Number and Name on Id Card Using Ocr (Optical Character Recognition) Method,” *J. Tek. Inform.*, vol. 5, no. 4, pp. 1191–1196, 2024.

[6] A. Izzuddin, R. R. M. Putri, and E. Setiawan, “Pengembangan Sistem Identifikasi Data Pada KTP Menggunakan YOLOv5 dan Tesseract OCR Berbasis Raspberry Pi 4B,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 9, no. 11, 2025.

[7] E. M. Sermana and D. Kurniadi, “Aplikasi Validasi e-KTP Berbasis Mobile Dengan Menerapkan Teknologi Optical Character Recognition,” *J. Algoritm.*, vol. 22, no. 1, pp. 504–516, 2025.

[8] H. P. Zou *et al.*, “Eiven: Efficient implicit attribute value extraction using multimodal llm,” *arXiv Prepr. arXiv2404.08886*, 2024.

[9] S. Wu, H. Fei, L. Qu, W. Ji, and T.-S. Chua, “Next-gpt: Any-to-any multimodal llm,” in *Forty-first International Conference on Machine Learning*, 2024.

[10] C. Mawela, C. Ben Issaid, and M. Bennis, “A web-based solution for federated learning with LLM-based automation,” *IEEE Internet Things J.*, 2025.

[11] H. Zhou *et al.*, “Large language model (llm) for telecommunications: A comprehensive survey on principles, key techniques, and opportunities,” *IEEE Commun. Surv. Tutorials*, vol. 27, no. 3, pp. 1955–2005, 2024.

[12] S. Wu, H. Fei, L. Pan, W. Y. Wang, S. Yan, and T.-S. Chua, “Combating Multimodal LLM Hallucination via Bottom-Up Holistic Reasoning,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025, vol. 39, no. 8, pp. 8460–8468.

[13] B. Li, Y. Jiang, V. Gadepally, and D. Tiwari, “Llm inference serving: Survey of recent advances and opportunities,” in *2024 IEEE High Performance Extreme Computing Conference (HPEC)*, 2024, pp. 1–8.

[14] M. A. Rahman, “A survey on security and privacy of multimodal llms-connected healthcare perspective,” in *2023 IEEE globecom workshops (GC Wkshps)*, 2023, pp. 1807–1812.

[15] F. Borisyuk, A. Gordo, and V. Sivakumar, “Rosetta: Large scale system for text detection and recognition in images,” in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018, pp. 71–79.

[16] Y.-F. Liao, Y.-H. Huang, M. Pleva, D. Hládek, and M.-H. Su, “A Preliminary Study on Taiwanese OCR for Assisting Textual Database Construction from Historical Documents,” in *2022 13th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, 2022, pp. 270–274.

[17] M. T. Shahriar and H. Li, “A study of image pre-processing for faster object recognition,” *arXiv Prepr. arXiv2011.06928*, 2020.

[18] M. P. Pavan Kumar *et al.*, “Image abstraction framework as a pre-processing technique for accurate classification of archaeological monuments using machine learning approaches,” *SN Comput. Sci.*, vol. 3, no. 1, p. 87, 2022.

[19] S. Selvakanmani, T. Chandrashekar, N. D. Federick, and A. M. Jaffar, “Optical character recognition based text analyser: A case study,” *Science (80-.).*, 2020.

[20] A. Lee and H. Tong, “Token-Efficient RL for LLM Reasoning,” *arXiv Prepr. arXiv2504.20834*, 2025.

[21] N. Singh *et al.*, “Load balancing and service discovery using Docker Swarm for microservice based big data applications,” *J. Cloud Comput.*, vol. 12, no. 1, p. 4, 2023.

[22] X. Wan, X. Guan, T. Wang, G. Bai, and B.-Y. Choi, “Application deployment using microservice and docker containers: Framework and optimization,” *J. Netw. Comput. Appl.*, vol. 119, pp. 97–109, 2018.

[23] P. S. Kocher, *Microservices and containers*. Addison-Wesley Professional, 2018.

[24] V. Khoriya, “The Future of SaaS Platforms: A Comprehensive Review,” *Vidhyayana-An Int. Multidiscip. Peer-Reviewed E-Journal-ISSN 2454-8596*, vol. 10, no. si2, pp. 74–86, 2024.

[25] Y. Han *et al.*, “Chartllama: A multimodal llm for chart understanding and generation,” *arXiv Prepr. arXiv2311.16483*, 2023.