PureCertificate: A Privacy-Preserving Small and Local VLM Framework for Secure Certificate Analysis

Odinachi Udemezuo Nwankwo, Heejae-Shin, Victor Ikenna Kanu,
Chigozie Athanasius Nnadiekwe, Ihunanya Udodiri Ajakwe, Kalibbala Jonathan Mukisa,
Nanteza Adah Lubwama, Dong-Seong Kim, Jae Min Lee,
Department of IT Convergence Engineering,
Kumoh National Institute of Technology, Gumi, South Korea.
{odinachifoot@gmail.com, shinheejae@kumoh.ac.kr, kanuxavier@gmail.com, cnnadiekwe01@gmail.com,
ihunanya.ajakwe@gmail.coml.com
dskim@kumoh.ac.kr, ljmpaul@kumoh.ac.kr}

***Abstract*—This work introduces PureCertificate, a privacy-preserving certificate analysis framework that uses locally deployed vision–language models instead of cloud services to meet military-grade confidentiality requirements. PureCertificate employs lightweight fine-tuning, dataset preprocessing, and GPT-Generated Unified Format (GGUF) conversion for efficient on-device inference through OLLAMA. Among the tested models, qwen3-vl:8b-instruct-q8_0 delivered the most reliable performance with a model size of about 9.8 GB. The framework supports similarity checking, information extraction, discrepancy detection, content recognition, and certificate comparison through a Flutter-connected inferencing module. Experiments show fluctuating desktop performance. Overall, PureCertificate is a promising solution to offline multimodal certificate analysis.**

***Index Terms*—Small Language Models (SLM), Vision–Language Models (VLM), Privacy-Preserving Document Analysis, Certificate Analysis**

I. Introduction

Secure and reliable document analysis is increasingly essential across governmental, industrial, and defense domains, where unauthorized alteration or forgery of certificates poses significant operational risks. Conventional cloud-based document analysis pipelines, while effective, introduce confidentiality and compliance concerns that are incompatible with sensitive or classified environments. To address these limitations, this work introduces PureCertificate, a locally deployable framework built on Small Language Models (SLMs) and Vision–Language Models (VLMs) [1] [2] [3] [4] to enable multimodal certificate understanding without external data transmission. By combining lightweight model fine-tuning, structured preprocessing, and mobile-edge inferencing through the OLLAMA runtime, the system provides a practical solution for similarity evaluation, content extraction, and discrepancy detection under strict privacy requirements. This approach advances document intelligence toward secure, decentralized, and operationally adaptable analysis architectures suited for military-grade privacy expectations.

The following list summarizes the primary contributions of this research:

1) A multimodal (image and text) analysis engine capable of similarity checking, information extraction, discrepancy detection, content recognition, and certificate comparison within a unified workflow.
2) A privacy-preserving certificate analysis framework that performs all document analysis locally using SLM- and VLM-based inferencing, eliminating dependence on cloud services and meeting military-grade confidentiality requirements.
3) A practical integration into real-world applications through a Flutter-based interface and local inferencing module, validated through desktop and device-level experiments that demonstrate stable performance.

The remaining parts of the paper are divided into the following sections. Section II describes the literature review. The proposed system workflow is explained in Section III. Section IV presents the experimental setup and results, while Section V discusses the conclusion and future work.

II. Literature Review

A research conducted in [5] developed a proof-of-concept system that verifies the authenticity of news screenshots by applying Optical Character Recognition (OCR) using the Google Cloud Vision Application Programming Interface (API) to extract text from an image, and then comparing that text against the New York Times (NYT) article database through the NYT Developers Application Programming Interface (API) to determine whether the screenshot corresponds to a real published article. Tested on a small curated dataset of 52 screenshots, the system achieves approximately 84.5% accuracy. The key research gap is that the method relies on exact or near-exact text matching from a single news source and therefore does not address multi-source analysis, paraphrased content, subtle manipulations, multilingual news, or noisy real-world social media layouts. The artificial intelligence (AI)

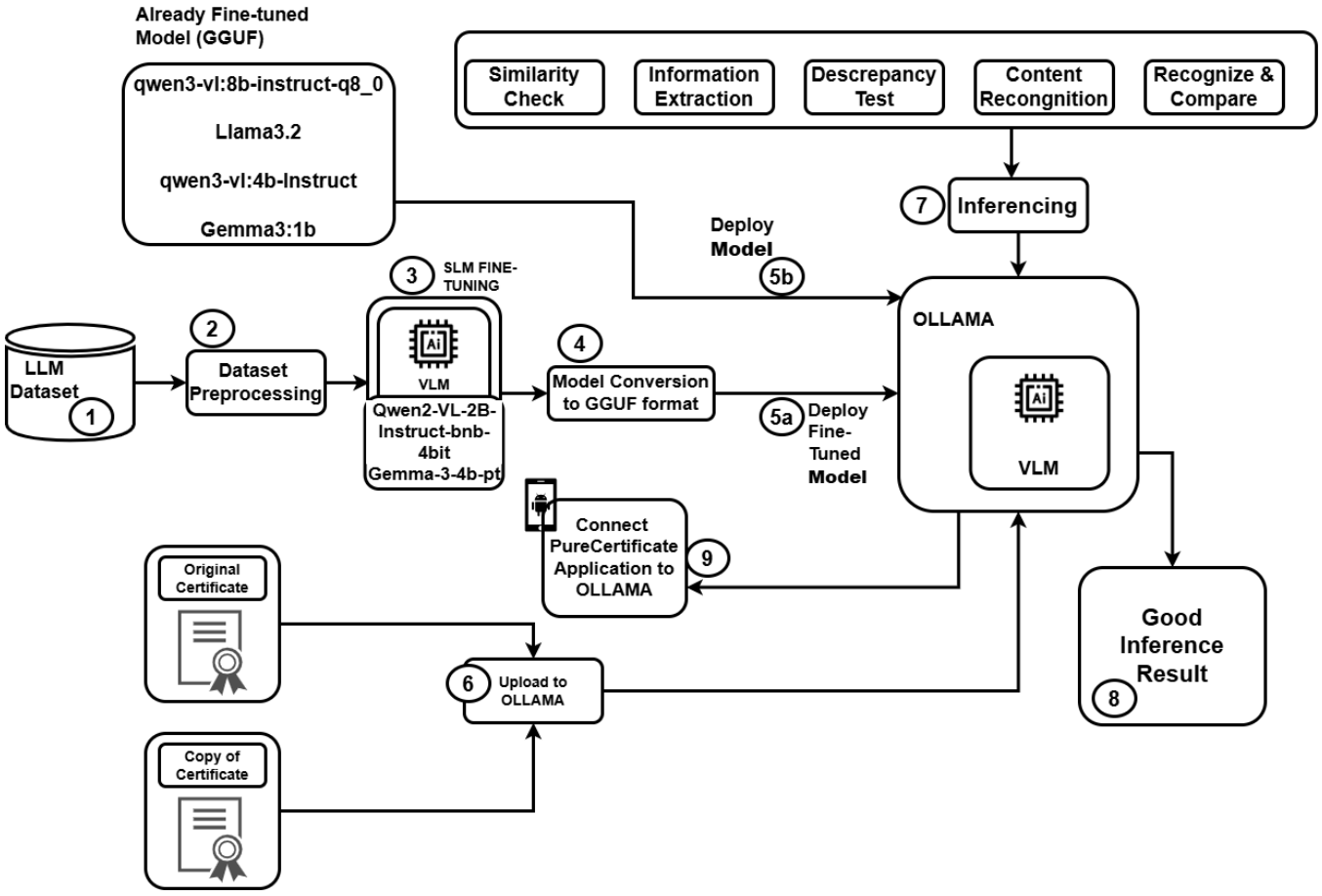

Fig. 1. System workflow showing local model tuning, deployment, and on-device certificate analysis within the PureCertificate framework..

limitation is that the system does not use advanced semantic language models or multimodal reasoning; instead, it depends purely on text matching, meaning it cannot understand article meaning, detect nuanced alterations, or generalize beyond the constrained New York Times-only setting.

Another study in [6] developed a multi-stage method for extracting very small text from document and chart images by combining single-image super-resolution with character-level segmentation before recognition using a Convolutional Recurrent Neural Network with Connectionist Temporal Classification loss. This approach improves accuracy by 18% over a standard baseline on small-text datasets, demonstrating that enhancing resolution and isolating characters helps overcome low-visibility text. However, a key research gap remains: the system struggles with multi-oriented text, tightly spaced characters, mixed font sizes, and complex real-world chart layouts, limiting its robustness. The artificial intelligence limitation is that the framework relies on traditional convolutional–recurrent models without modern transformer or attention-based architectures, preventing it from leveraging contextual semantic cues and reducing its generalization ability across diverse and noisy image conditions.

The study in [7] built a text-extraction system that combines image preprocessing, convolutional neural network–based text detection, character segmentation, and Tesseract Optical Character Recognition within a Streamlit web interface, enabling users to upload images and retrieve extracted text. The work demonstrates effective extraction under clean, horizontally aligned conditions but leaves a research gap in handling skewed, low-quality, multilingual, handwritten, and complex-background text that occurs in real-world scenarios. Its artificial intelligence limitation is reliance on traditional models and Tesseract OCR rather than modern transformer-based or context-aware recognition systems, reducing accuracy, robustness, and semantic understanding.

The study in [8] evaluates eight combinations of speech-to-text, large language models, and text-to-speech systems for controlling a ROSbot XL through spoken commands, demonstrating that the Whisper + GPT-4.0 + Google Text-to-Speech pipeline yields the most accurate and efficient robot responses, while all LLaMA 3.2 configurations perform considerably worse. The work was not intended as a comprehensive or certifiable analysis of autonomous robotic control; rather, it serves as a targeted performance comparison within a controlled simulation environment. The work was not meant for certificate analysis or formal certification of robotic systems, but rather for exploratory performance evaluation within a controlled simulation environment.

TABLE I
SMALL LANGUAGE MODELS (VISION-LANGUAGE AND TEXT-ONLY) WITH REASONING CLASSIFICATION

Model Full Name	Modality	Vision Capability	Parameters (B)	Classification
qwen3-vl:8b-instruct-q8_0	Vision-Language	Yes	8.0	Reasoning
Qwen3-VL-3B	Vision-Language	Yes	3.0	Reasoning
Phi-3.5-Vision	Vision-Language	Yes	≈4.0	Reasoning
DeepSeek-VL-1.3B	Vision-Language	Yes	1.3	Reasoning
DeepSeek-OCR-3B	Vision-Language	Yes (OCR-focused)	≈3.0	Non-Reasoning
LLaVA-7B (LLaMA/Vicuna-based)	Vision-Language	Yes	7.0–7.2	Reasoning
Llama 3.2-3B-Instruct	Text-Only	No	3.0	Reasoning
Llama 3.1-8B-Instruct	Text-Only	No	8.0	Reasoning
Llama 3.2 Vision 11B	Vision-Language	Yes	11.0	Reasoning

TABLE II
GGUF/GGML INTEGER QUANTIZATION FORMATS

Format	Full Meaning of Abbreviation	Bit-Width	Fidelity Level	Typical Use-Case
Q8_0	Quantized 8-bit, Scheme 0 (symmetric zero-point)	8	High	General local inference; minimal quality loss
Q6_K	Quantized 6-bit, K-type block quantization	6	High	Efficient CPU/GPU inference with strong accuracy
Q5_K	Quantized 5-bit, K-type block quantization	5	Medium-high	Low-VRAM GPUs; faster CPU inference
Q5_1	Quantized 5-bit, Scheme 1 (with bias correction)	5	Medium-high	Lightweight inference needing higher precision than Q5_0
Q5_0	Quantized 5-bit, Scheme 0 (baseline 5-bit)	5	Medium	General reduction with slightly more compression
Q4_K	Quantized 4-bit, K-type block quantization	4	Medium-high	Best 4-bit fidelity; widely used for laptops/CPUs
Q4_1	Quantized 4-bit, Scheme 1 (bias-aware)	4	Medium	Balanced compression + accuracy
Q4_0	Quantized 4-bit, Scheme 0	4	Medium	Most common lightweight 4-bit baseline
Q3_K	Quantized 3-bit, K-type block quantization	3	Low-medium	Very compressed CPU/edge inference
Q2_K	Quantized 2-bit, K-type block quantization	2	Low	Extreme compression for tiny devices
Q1_K	Quantized 1-bit, K-type block quantization	1	Very low	Research on binary LLMs only

III. PROPOSED SYSTEM WORKFLOW

The system workflow in Fig. 1 follows a structured sequence that ensures secure, local, and efficient certificate analysis using SLM and VLM models.

A. Dataset Description

The dataset is sourced from Hugging Face (unsloth/LaTeX_OCR) and contains images of mathematical expressions paired with their ground-truth LaTeX representations. It is divided into a training set (68,686 samples) and a test set (7,632 samples). Each sample consists of an image and a corresponding LaTeX text label, making the dataset suitable for vision-to-text OCR tasks, specifically mathematical formula recognition using vision–language models.(Step 1), followed by dataset preprocessing (Step 2).

B. Dataset Preprocessing

During preprocessing, each dataset sample is converted into a conversation-based instruction format required for vision fine-tuning. A fixed instruction, defined as `"Write the LaTeX representation for this image."`, is used for all samples. For each sample, the `convert_to_conversation(sample)` function creates a structured message sequence consisting of a user role and an assistant role. The user message contains two content elements: a text instruction (`{"type": "text", "text": instruction}`) and the corresponding image (`{"type": "image", "image": sample["image"]}`). The assistant message contains the ground-truth LaTeX label (`{"type": "text", "text": sample["text"]}`). The function returns the formatted output as `{"messages": conversation}`, ensuring that all samples follow a consistent structure compatible with supervised vision–language fine-tuning.

C. Model Fine-Tuning

The pretrained **Gemma-3 Vision–Language Model** is fine-tuned using supervised fine-tuning with `SFTTrainer` and the `UnslothVisionDataCollator` to support vision–language learning. Fine-tuning is enabled using `FastVisionModel.for_training(model)` and performed on the converted conversation dataset. The configuration uses `per_device_train_batch_size = 1` with `gradient_accumulation_steps`

= 4 and gradient_checkpointing = True (with use_reentrant = False) to reduce memory usage. Optimization is carried out using adamw_torch_fused with learning_rate = 2e-4, weight_decay = 0.01, max_grad_norm = 0.3, warmup_ratio = 0.03, and a cosine learning rate scheduler. Fine-tuning is limited to max_steps = 30 for faster execution, with logging_steps = 1, save_strategy = "steps", and seed = 3407. Vision fine-tuning requirements are satisfied by setting remove_unused_columns = False, dataset_text_field = "", dataset_kwargs = {"skip_prepare_dataset": True}, and max_length = 2048. A selected lightweight vision–language model, such as Qwen2-VL-2B or Gemma-3-4b, is then fine-tuned (Step 3) using a parameter-efficient SLM tuning strategy.

D. Model Inference

After fine-tuning, the pretrained **Gemma-3 Vision–Language Model** is switched to inference mode using FastVisionModel.for_inference(model). A test image is selected from the dataset, and a user-defined instruction is provided while leaving the output empty. The input is formatted using the Gemma-3 instruction chat template and processed jointly with the image to construct the model input tensors.

Text generation is performed using optimized inference hyperparameters, namely temperature = 1.0, top_p = 0.95, and top_k = 64, which balance output diversity and generation stability. The model generates responses token by token using caching and streaming, producing either a textual description or a LaTeX representation of the input image. This demonstrates the model's ability to generalize and generate meaningful outputs from previously unseen visual data.

The tuned model is subsequently converted into the GGUF format (Step 4) to enable optimized execution within the OLLAMA runtime. Deployment takes place through two paths: either a pre-fine-tuned model is directly loaded (Step 5b) or the newly fine-tuned model is deployed (Step 5a).

In parallel, users upload an original certificate and its comparison copy to the PureCertificate application (Step 6). The application, connected to the OLLAMA inference server (Step 9), forwards the processed visual inputs to the local VLM engine. During inferencing (Step 7), the system performs a sequence of analytical tasks, including similarity checking, information extraction, discrepancy testing, content recognition, and certificate comparison using targeted prompts routed to the appropriate VLM. The inference outputs are then evaluated for consistency and quality, producing a structured "Good Inference Result" (Step 8) that is returned to the application interface. Through this workflow, all document data remains local, meeting stringent privacy requirements while enabling accurate multimodal certificate analysis.

Table I consolidates the candidate small-scale models assessed for PureCertificate by contrasting (i) modality (vision–language versus text-only), (ii) native visual perception availability, (iii) parameter count in billions P (spanning approximately $P \in [1.3, 11]$B), and (iv) a coarse "reasoning" versus "non-reasoning" designation. This characterization is operationally salient for certificate analysis because the target functions—including similarity checking, discrepancy detection, and certificate-to-certificate comparison—are intrinsically multimodal, such that models with explicit vision capability are structurally advantaged over text-only baselines that would otherwise require an external OCR front-end. The predominance of vision–language entries marked as reasoning-capable suggests suitability for multi-step, context-dependent judgments beyond transcription, whereas the inclusion of an OCR-focused non-reasoning variant motivates a division of labor in which perception-centric extraction can be decoupled from deliberative verification. Finally, the spread in model sizes reflects the deployment trade-off in local settings, where resource consumption increases with P and numeric precision; in practice, on-device feasibility is governed by an approximate scaling relationship $M \propto P \cdot b$, where M denotes model weight memory and b denotes effective bit-width.

Table II specifies the GGUF/GGML integer quantization design space used to enable local deployment by enumerating formats ({Q8_0, Q6_K, Q5_K, Q5_1, Q5_0, Q4_K, Q4_1, Q4_0, Q3_K, Q2_K, Q1_K}) and mapping each scheme to bit-width $b \in \{8, 6, 5, 4, 3, 2, 1\}$, an expected fidelity tier, and a typical deployment niche. The table thereby operationalizes the monotone compression–quality frontier: higher precision (e.g., $Q8\_0$) is positioned for high-fidelity inference with minimal quality loss, while intermediate 5-bit and 4-bit schemes—including block-quantized K-type variants—target practical constraints such as low-VRAM GPUs and laptop/CPU inference while preserving usable accuracy. Conversely, aggressive compression (e.g., $Q3\_K$ and $Q2\_K$) is aligned with edge CPU contexts at the cost of lower fidelity, and the 1-bit setting is framed as primarily research-oriented due to severe representational loss. Conceptually, reducing b lowers storage and bandwidth demands approximately linearly in b, but introduces quantization distortion that can degrade extraction and reasoning if pushed beyond the robustness envelope of the deployed vision–language model.

IV. Experimental Setup and Result

The experiment was conducted using a locally hosted OLLAMA server configured on a desktop environment equipped with GPU acceleration, enabling real-time execution of SLM and VLM models in gguf format. Multiple lightweight models were tested, including Qwen2 VL 2B, Gemma 3 4b, and Qwen3 VL variants, to assess inference stability, accuracy, and resource efficiency within the PureCertificate workflow. Among these, qwen3-vl-8b-instruct-q8_0 demonstrated the highest reliability and most consistent outputs while maintaining a manageable 9.8 GB model size, making it a promising candidate for on-device certificate analysis. The Flutter client connected seamlessly to the local server on desktop and localhost configurations, producing successful inference re-

sponses across all analysis functions. In contrast, real mobile-device testing revealed connectivity limitations for externally accessed local servers, highlighting the need for secure tunneling approaches such as Virtual Private Network (VPN) or Zero Trust Network Access (ZTNA). Overall, the results demonstrate the feasibility of implementing local and privacy-preserving document analysis using small vision–language models.

The fine-tuning l oss exhibits a clear downward trend across the 30 fine-tuning steps, indicating effective convergence of the model. While the loss fluctuates during the initial fine-tuning phase, reaching a peak of 2.74 at step 2, such behavior is expected as the model adapts to the vision–language task. As fine-tuning progresses, the loss steadily decreases, dropping below 1.0 by step 7 and stabilizing around 0.3 in later steps. The final loss value of 0.264 at step 30 demonstrates that the model successfully learned the task-specific representations within a limited number of fine-tuning steps, confirming the stability and effectiveness of the fine-tuning setup.

Fig. 2 presents the graphical user interface of the PureCertificate application and illustrates how end users interact with the local vision–language inference pipeline. The AI Analysis in the Pure Certificate application provides a straightforward way to compare and process documents. Users select a reference file and a comparison file, then choose from several automated tasks such as checking for similarity, recognizing content, or extracting specific information. By handling these complex AI tasks on a local server rather than the cloud, the application allows for advanced document analysis while keeping the user's data private and the interface easy to navigate.

Fig. 3 illustrates the response time behavior of the PureCertificate application over fifteen consecutive requests. Based on the bar chart, the response time behavior is characterized by an initial peak followed by frequent fluctuations rather than a steady-state. The first request requires the most time at 67.0 seconds, likely due to the system's initialization. Once this is complete, the subsequent 14 requests show significant variability, with times ranging from a minimum of 21.6 seconds (Request 3) to a maximum of 46.0 seconds (Request 13). While there is no clear trend of performance degradation, the response times do not settle into a consistent range, oscillating instead between 21 and 46 seconds throughout the test period.

As summarized in the table III, Prior work primarily targets narrow OCR accuracy or cloud-based, domain-specific verification, whereas PureCertificate delivers a fully local, multimodal framework that integrates perception and semantic reasoning. By enforcing data locality and enabling higher-level operations such as similarity and discrepancy analysis, it moves beyond OCR toward secure document intelligence suitable for sensitive deployments.

V. Conclusion and Future Work

It appears that PureCertificate could serve as a promising, privacy-preserving solution for certificate analysis by leveraging locally deployed Vision-Language Models. By utilizing GGUF-quantized models like qwen3-vl:8b-instruct-q8_0 within the OLLAMA runtime, the system suggests a viable path for performing complex tasks such as similarity checking and discrepancy detection, without external data transmission.

Future work should pursue adaptive, context-aware quantization methods and hybrid precision strategies that further reduce response time, resource usage as well as evaluate performance of other promising AI models. Future work will also focus on clearly distinguishing OCR functionality from higher-level VLM-based semantic reasoning, and on incorpo-

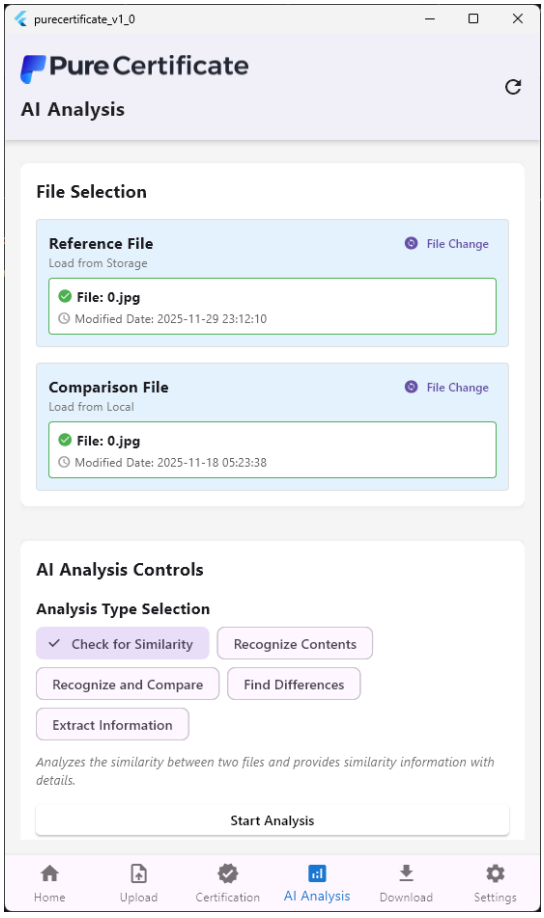

Fig. 2. PureCertificate application user interface displaying local vision–language model inference commands.

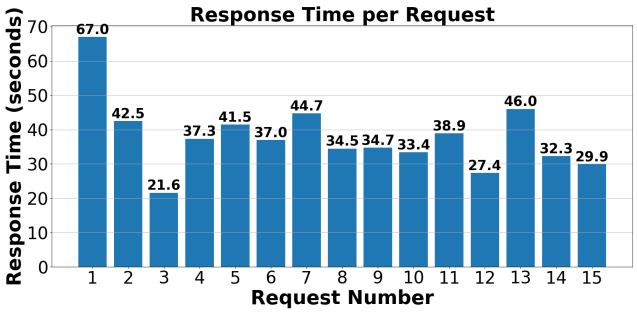

Fig. 3. Response Time per Request

TABLE III
COMPARATIVE ANALYSIS WITH EXISTING LITERATURE ACROSS APPLICATION DOMAIN, TECHNIQUE, DEPLOYMENT, PRIVACY HANDLING, ANALYTICAL CAPABILITY, AND LIMITATIONS.

Work	Primary Application Domain	Core Technique	Deployment Model	Privacy Handling	Analytical Capability	Key Limitations
Kamal et al. [5]	News screenshot verification	Cloud-based OCR with exact text matching against NYT database	Fully cloud-dependent	Low (external APIs and databases)	Text extraction and direct matching	Dependent on exact text matching; cannot understand article meaning.
Busa et al. [6]	Small text extraction from documents and charts	Super-resolution; character-level segmentation; CRNN with CTC	Offline, model-specific	Medium (local processing)	Fine-grained OCR accuracy improvement	Reliance on traditional neural network models rather than modern, context-aware transformer architectures
Rajmod et al. [7]	Generic OCR for images	Image preprocessing; CNN-based detection; Tesseract OCR	Web-based application	Medium (user uploads to web interface)	Basic text extraction	Lack of attention mechanism & context-awareness
Uruj et al. [8]	Human–robot interaction	Speech-to-text; large language models; text-to-speech	Mixed cloud and local	Not a design focus	Spoken command understanding	Not intended for document or certificate analysis
PureCertificate (This Work)	Secure certificate analysis	Local vision–language models with multimodal reasoning	Fully local (OLLAMA-based)	High (no external data transmission)	Similarity checking; information extraction; discrepancy detection; content recognition; certificate comparison	Current latency constrained by on-device inference resources

rating quantitative evaluation metrics such as accuracy and resource utilization to better validate deployment feasibility. The study will also include a comprehensive security and privacy threat analysis addressing potential data leakage and offline verification scenarios. In addition, the related work will be strengthened by incorporating recent advances in VLM-based document understanding, privacy-preserving machine learning, and document forgery and similarity detection.

ACKNOWLEDGEMENT

This work was partly supported by Innovative Human Resource Development for Local Intellectualization program through the IITP grant funded by the Korea government(MSIT) (IITP-2025-RS-2020-II201612, 25%) and by Priority Research Centers Program through the NRF funded by the MEST(2018R1A6A1A03024003, 25%) and by the MSIT, Korea, under the ITRC support program(IITP-2025-RS-2024-00438430, 25%). This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(RS-2025-25431637, 25%).

REFERENCES

[1] Anonymous, "A survey on efficient vision-language models," *preprint*, 2025, survey of compact VLM architectures and efficiency techniques. [Online]. Available: https://github.com/MPSC-UMBC/Efficient-Vision-Language-Models-A-Survey

[2] Z. Yi, T. Xiao, and M. V. Albert, "A survey on multimodal large language models in radiology for report generation and visual question answering," *Information*, vol. 16, no. 2, p. 136, 2025.

[3] J. Lee and J. Rew, "Vision-language model-based local interpretable model-agnostic explanations for intrusion detection," *Sensors*, vol. 25, no. 10, p. 3020, 2025.

[4] H.-T. Ho, L. V. Nguyen, M.-T. Pham, Q.-H. Pham, Q.-D. Tran, D. N. M. Huy, and T.-H. Nguyen, "A review on vision-language-based approaches: Challenges and applications," *Computers, Materials and Continua*, vol. 82, no. 2, pp. 1733–1756, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1546221825001420

[5] A. Kamal, Z. Jamal, G. Rosales, B. Robinson, Z. Sotny, and H. Rathore, "Image to text recognition for detecting human and machine altered news in social media," in *2023 10th International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, 2023, pp. 72–74.

[6] R. Busa, K. C. Shahira, and A. Lijiya, "Small text extraction from documents and chart images," in *2022 IEEE 19th India Council International Conference (INDICON)*, 2022, pp. 1–5.

[7] V. Rajmod, G. Derkar, P. Nagrale, N. Awari, and M. Lokhande, "Text extraction from image using ocr," in *2025 6th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI)*, 2025, pp. 113–116.

[8] S. Uruj, R. Goswami, S. D. Shetty, K. Venkatesan, and K. Ramanujam, "Comparative analysis of gpt-4 and llama 3.2 integration with speech processing models for enhancing human–robot interaction and motion control in real-world applications," *IEEE Access*, vol. 13, pp. 127 170–127 182, 2025.