

A Comparative Study of Traditional OCR Models and Large Language Models for Optical Character Recognition

Bhupesh Kurra, Harshita Panwar, Neeraj Rithaliya, Nabhi Jain, Prof. Nidhi Nigam, Dr. Shilpa Bhalerao

Department of Computer Science and Information Technology

Acropolis Institute of Technology and Research Bypass Road, Manglia Square, 453771

[<bhupshkurra95@gmail.com>](mailto:bhupshkurra95@gmail.com), [<harshitapanwar0910@gmail.com>](mailto:harshitapanwar0910@gmail.com),

[<neerajrithaliya221020@acropolis.in>](mailto:neerajrithaliya221020@acropolis.in),

[<jainnabhi4@gmail.com>](mailto:jainnabhi4@gmail.com) [<nidhinigam@acropolis.in>](mailto:nidhinigam@acropolis.in), [<shilpabhalerao@acropolis.in>](mailto:shilpabhalerao@acropolis.in),

Abstract

Optical character recognition (OCR) has come a long way. Earlier system relied heavily on template matching and manual feature extraction, but modern OCR now uses advanced deep learning models and methodologies. Recently, large language models, or alms. Especially multimodal ones that can process both images and text have opened a new approach to extracting text from images.

This paper compares traditional OCR tools such as Tesseract and Easy OCR with newer ALM based OCR solutions like Mistral OCR, GPT-4o, and Gemini. We evaluate their performance by examining accuracy across different types of documents and image conditions (such as resolution and complexity), along with speed, cost, layout-handling capability, and overall reliability. We also consider important risks such as hallucinations, using evidence from model testing studies, including those conducted by RoboFlow.

Our results show that LLM-based OCR systems are generally stronger at understanding context and handling complex layouts, while traditional OCR tools often remain more cost efficient and can even be more accurate. In certain scenarios, especially when dealing with low resolution images or text that does not require contextual interpretation.

Keyword: Optical Character Recognition (OCR), Large Language Models (LLMs), Multimodal LLMs, Text Recognition, Performance Evaluation, Comparative Study.

I. INTRODUCTION

Optical character recognition (OCR) is an essential technology used to convert printed or handwritten text from scanned documents, PDFs, or camera captured images into digital texts that can be searched, edited, and stored efficiently [1]. For many years, traditional OCR systems have been widely used, relying on methods such as pattern recognition and feature extraction to identify characters and words.

In recent years, however, the rapid development of Large Language Models (LLMs) has introduced a new direction for OCR, especially with the emergence of multimodal LLMs, models capable of processing both visual and textual information at the same time [2], [3], [4]. These systems offer stronger contextual understanding and greater adaptability when dealing with complex document layouts, making them promising tools for advanced document digitalization and analysis [4], [5].

This paper focuses on comparing traditional OCR systems with modern LLM-based OCR approaches. We examine how each method works, evaluate their performance in terms of accuracy, speed, and cost. And explore which types of OCR tasks they are best suited for. Our analysis is supported by recent benchmarks and studies from sources such as

Mistral AI [7], Klippa [5], Cradl AI [4], academic research papers [2], and industry evaluations like RoboFlow [6]. Key factors considered include image quality, document complexity, language support, and potential risks or limitations associated with each technology, such as the potential for LLM hallucinations [5], [9].

II. BACKGROUND AND RELATED WORK

A. Traditional OCR Models

Traditional OCR systems usually follow a step-by-step pipeline. First, an image is captured or uploaded, and then preprocessing techniques are applied to improve readability, such as describing the image, removing noise, and converting it into a clean black-and-white format (binarization). After this, the system performs text recognition using pattern matching or feature extraction methods, followed by post processing to refine the final output [1].

Some of the most widely used open-source OCR engines include Tesseract (often accessed through tools like PyTesseract) and EasyOCR. Other OCR frameworks such as Surya and DocTR are also considered part of this traditional category [6]. These systems are often trained for specific recognition tasks and in many cases may rely on predefined templates when working with structured documents. While they can deliver excellent results on clean and standardized inputs, their performance often drops when dealing with poor image quality, distorted text, or complex layouts [1].

B. LLMs for OCR

Large Language Models (LLMs), especially multimodal models such as GPT 4-o [3] and Gemini [8], along with specialized tools like Mistral OCR [7], take a very different approach to OCR. Instead of relying on strict recognition pipelines, these models use transformer-based architectures trained on massive datasets to understand both the visual structure and the text content of a document. This allows them to intertrade context and handle divorce or unstructured layouts without needing predefined templates [4], [5].

Text-only LLMs can work effectively when documents already contain selectable text, while multimodal LLMs can directly analyze images to extract and interpret text content [5], [9]. However, because these models often combine recognition with text generation, they may sometimes hide character level mistakes by “guessing” the correct word based on context rather than reading it accurately [2]. In addition, concerns remain about privacy, especially when using closed source cloud models. In the risk of hallucinations where the model generates text that is not actually present in the image [4], [5].

III. METHODOLOGY

Evaluating OCR systems requires consistent testing methods, reliable datasets, and standardized performance metrics. Since OCR tools vary widely in how they function, different research studies often follow different evaluation approaches depending on the domain and use case.

A. Traditional OCR Testing

The Roboflow study evaluated multiple traditional OCR models, mainly focusing on industrial and non-document OCR applications rather than standard scanned documents.

The study tested the following models, Tesseract (through PyTesseract), EasyOCR, and Surya, all of which were run locally. It also included DocTR, which was tested using the Roboflow Hosted API.

For evaluation, the data set consisted of cropped images taken from ten public datasets available on Roboflow Universe. These datasets cover different industrial OCR domains. From each domain, 10 images were randomly selected. Many of these images were cropped using existing object detection annotations to isolate the exact region containing text.

Accuracy was measured using the Levenshtein distance ratio, which compares the OCR output to manually written

ground truth text. This study mainly focused on comparing 3 major factors across domains, accuracy, speed, medium, prediction time, and cost efficiency.

B. Standard OCR Accuracy Metrics

OCR performance is commonly evaluated using well-established error rate metrics:

- **Character Error Rate (CER):**

CER measures how many characters were incorrectly recognized compared to the ground-truth. It accounts for substitutions, deletions, and insertions. A lower CER means better accuracy.

$$CER = \frac{S_c + D_c + I_c}{N_c}$$

There SC, DC and IC represent character substitutions, deletions and insertions, respectively and NC is the total number of ground truth characters.

- **Word Error Rate (WER):**

WER evaluates OCR performance at the word level, measuring how many words were predicted incorrectly. Like CER, a lower WER indicates higher accuracy.

$$WER = \frac{S_w + D_w + I_w}{N_w}$$

where Sw, Dw, and Iw represent wrd susbtitutions, deletions, and insertions, respectively, and Nw is the total number of ground-truth words.

In most OCR evaluations, the standard procedure involves comparing the row OCR output with manually transcribed and verified ground truth text across a representative set of documents. Preprocessing steps such as binarization and deskewing are often applied before recognition to improve results.

C. LLM-Based OCR Testing

Testing OCR performance for large language models (LLMs) often follows a different strategy compared to traditional OCR evaluation. Many LLM-based OCR studies rely on academic benchmarks or specially designed datasets that highlight the unique strengths and weaknesses of these models.

The datasets used often include standard OCR benchmark collections containing diverse materials such as multilingual text, scanned documents, tables, and mathematical notation [7]. Some studies also introduce custom datasets designed to text context independent character recognition under different image resolutions [2].

The models evaluated in these studies include Mistral OCR, Google Document AI, Azure OCR, different versions of Gemini, and GPT-4o [2], [7].

Performance is usually measured using accuracy scores, fuzzy string-matching metrics [7], and degradation analysis under varying conditions such as changing resolution levels (measured in pixels per inch, ppi) [2]. These evaluations typically focus on overall accuracy, performance across specialized domains (such as mathematical expressions or tabular data), robustness to low quality images, speed, and direct comparisons with both other elements

and traditional OCR systems.

IV. RESULTS AND DISCUSSION

A. Accuracy and Performance

OCR performance can vary widely depending on the type of model being used, the specific task, and the quality of the input data.

Findings from the Roboflow Study [6]:

Among the traditional OCR models tested for non document and industrial OCR tasks, EasyOCR showed consistently strong performance. Across most of the ten evaluated domains. In that comparison, it outperformed several other specialized OCR packages, suggesting that it is a reliable and robust option for certain industrial OCR applications.

Findings from LLM Benchmarks [2], [7]:

Recent benchmark evaluations show that LLM-based OCR systems, especially Mistral OCR, have achieved state-of-the-art performance on large-scale document understanding benchmarks. In multiple categories, such as overall accuracy, mathematical content, multilingual text, scanned documents, and table extraction, Mistral OCR outperformed established OCR services like Azure OCR and Google Document AI (as shown in Table I and Table II) [7].

However, other studies highlight an important limitation. Multimodal alms such as GPT 4-o, and Gemini can experience significant drops in performance when image resolution is low, for example, below 150 ppi. Under these conditions, their accuracy may fall behind traditional systems like Azure Computer Vision. Essentially and tasks involving context independent characters where their model cannot rely on surrounding meaning [2].

Overall, LM based OCR tends to perform best in scenarios where contextual reasoning improves recognition or where document layouts are complex and inconsistent. Additionally, Table III supports the trend by showing strong fuzzy Max scores for Mistral OCR in generation-based OCR tasks [7].

B. Speed and Cost

Traditional OCR engines such as Tesseract, DocTR and Surya are generally much cheaper to operate than large multimodal LLM based systems, especially when the task is simple text extraction. The RoboFlow analysis highlights this cost advantage, clearly showing that traditional OCR tools can deliver acceptable results at a far lower computational and financial cost. Among the models tested in that study, EasyOCR offered the best overall cost efficiency, making it a strong choice for practical industrial use cases where budget and scale matter [6].

TABLE I

BENCHMARK COMPARISION OF LLM/OCR MODELS (SCORES)

Model	Overall	Math	Multilingual	Scanned	Tables
Google Document AI	83.42	80.29	86.42	92.77	78.16
Azure OCR	89.52	85.72	87.52	94.65	89.52
Gemini-1.5-Flash-002	90.23	89.11	86.76	94.87	90.48
Gemini-1.5-Pro-002	89.92	88.48	86.33	96.15	89.71
Gemini-2.0-Flash-001	88.69	84.18	85.80	95.11	91.46
GPT-4o-2024-11-20	89.77	87.55	86.00	94.58	91.70
Mistral OCR 2503	94.89	94.29	89.55	98.96	96.12

Source : Adapted from [7]

TABLE II

MULTILINGUAL PERFORMANCE COMPARISION (SCORES)

Language	Azure OCR	Google Doc AI	Gemini-2.0-Flash-001	Mistral OCR 2503
ru	97.35	95.56	96.58	99.09
fr	97.50	96.36	97.06	99.20
hi	96.45	95.65	94.99	97.55
zh	91.40	90.89	91.85	97.11
pt	97.96	96.24	97.25	99.42
de	98.39	97.09	97.19	99.51
es	98.54	97.52	97.75	99.54
tr	95.91	93.85	94.66	97.00
uk	97.81	96.24	96.70	99.29
it	98.31	97.69	97.68	99.42
ro	96.45	95.14	95.88	98.79

Source: Adapted from [7]

In contrast, LLM-based OCR systems often come with higher and less predictable costs. Models accessed through APIs typically charged based on usage, meaning costs can rise quickly when processing large volumes of documents. Even when open source LLMs are self-hosted, the required infrastructure, such as GPU servers, storage, scaling systems, and maintenance adds significant overhead. That said, specialized OCR focused L&M services like Mistral OCR claim very high-performance processing speeds which could make them attractive for large scale document workflows if cost is justifiable [7].

C. Strengths and Weaknesses Traditional OCR

Traditional OCR systems remain widely used because they are stable, well tested, and easy to deploy. They are often highly cost effective, particularly for simpler recognition tasks or offline/local processing. In addition, they can achieve very strong accuracy on clean and structured documents such as printed forms or scanned PDFs, and they generally provide consistent and predictable output formats [1], [6].

However, traditional OCR models tend to struggle when document layouts vary significantly or when the format is unfamiliar. Since they lack deep contextual understanding, they may confuse visually similar characters such as “O” and “0” or “i” and “1”. Their performance is also highly sensitive to image quality, meaning noise, blur, skew and poor

lighting can cause major drops in accuracy. In structured extraction tasks, they may also require templates or manual configuration to correctly identify key fields [1], [5].

LLM-Based OCR

LLM-based OCR approaches offer a different set of advantages. They are highly flexible and can handle complex or unsealed layouts more effectively than traditional OCR. Because they incorporate language understanding, they are better at interpreting context, resolving ambiguous characters, and even inferring missing information when the text is partially unclear. These models are also easy to interact with through prompts, making them convenient for tasks such as summarization, field extraction, and unstructured document understanding. In many cases, they also perform well across diverse languages, scripts, and even handwritten texts [4], [5], [7].

TABLE III

FUZZY MATCH IN GENERATION COMPARISON (SCORES)

Model	Fuzzy Match Score
Google-Document-AI	95.88
Gemini-2.0-Flash-001	96.53
Azure OCR	97.31
Mistral OCR 2503	99.02

Source: Adapted from [7]

Despite these trends, LLM-based OCR systems have serious limitations. A key concern is the risk of hallucination, where the model generates text that appears plausible but is not actually present in the image. Output formatting may also be inconsistent unless carefully controlled through prompting or structured constraints. The research also shows that performance can degrade sharply under low resolution conditions, particularly when the task requires precise recognition of context independent characters. Additionally, LLM's often do not provide standardized confidence scores like traditional OCR engines. Privacy is another concern, especially when documents must be sent to third party cloud APIs. Finally, these systems can be computationally expensive and may involve high usage costs at scale [4], [5], [9], [10].

Table III further supports the competitive performance of LLM based OCR in generation tasks, showing that Mistral OCR achieved the highest fuzzy match score among the compared models [7].

D. Graphs and Visualizations

Visual representations play an important role in comparing OCR systems because they allow performance differences to be understood more clearly. The Roboflow study [6] included multiple types of plots, such as bar charts comparing accuracy across different domains, scattered plots showing the relationship between speed (latency) and accuracy, and additionally graphs displaying median prediction time and estimated cost. Request for the tested traditional OCR models.

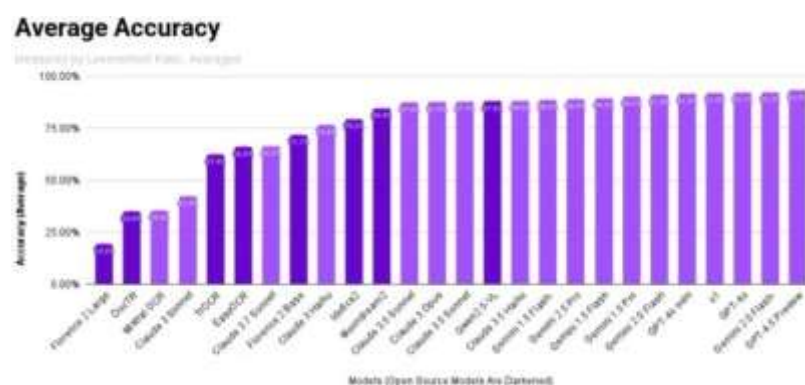


Fig. 1: Example: Comparison of OCR accuracy across different domains (Illustrative, based on descriptions in [6]).

Similarly, the arXiv study by K. Hiraoka [2] presented line graphs illustrating how recognition accuracy changes as image resolution decreases. These plots compared multimodal Alms with baseline OCR services, showing performance degradation trends across different pixels per inch (ppi) levels. Figures such as Fig.1 and Fig. 2, serve as useful visual summaries of these findings. Can help highlight the trade-offs between OCR approaches under varying real-world conditions.

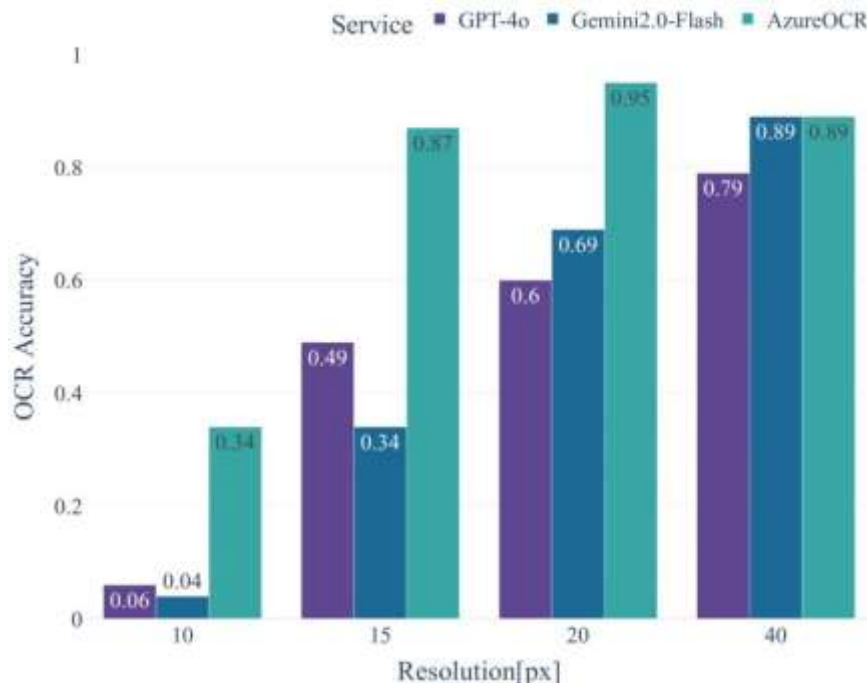


Fig. 2: Example: OCR accuracy vs. image resolution for different models (Illustrative, based on descriptions in [2]).

V. CONCLUSION

Both traditional OCR systems and LLM-based OCR approaches come with their own strengths and limitations. Traditional OCR engines such as EasyOCR and Tesseract are still highly relevant today because they provide reliable and cost-effective performance, particularly well-defined tasks and standardized document formats. They are especially useful in scenarios where local deployment is preferred; costs must remain predictable, and the input data is clean enough that heavy contextual reasoning is not required [6]. In such cases, traditional OCR can deliver strong accuracy with minimal complexity.

On the other hand, Elam based OCR, especially multimodal models like Mistral-OCR represents a major step forward and document understanding. These models offer much greater flexibility when working with diverse, unstructured, or unfamiliar layouts. Their ability to use contextual understanding often allows them to perform better on complex, low quality or degraded documents. Additionally. They make OCR workflows easier to use through natural language prompting, reducing the need for rigid templates or manual rule-based extraction methods [4], [7].

However, LLM-based OCR also introduces important challenges. Users must account for risks such as hallucinations, where the model may generate text that appears believable but is not actually present in the document. Performance can also drop noticeably when image resolution is low, as highlighted in recent studies [2]. Other concerns include difficulty in consistently producing structured outputs without strict prompting and the lack of standardized confidence scores that traditional OCR systems often provide [4], [9].

Ultimately, the best OCR solution depends heavily on the specific use case. The key factors include the type of document (structured versus unstructured). The required accuracy level, Tolerance for different types of errors (misrecognition versus hallucination), cost constraints, volume of processing, latency requirements, and data privacy policies. Future research will likely focus on hybrid systems that combine traditional OCR reliability with LLM-level contextual understanding, along with better fine-tuning methods and stronger verification techniques to improve

trust and consistency in LLM-based OCR outputs.

REFERENCES

- [1] DocuClipper, “What Is OCR Accuracy and How To Measure It,” *DocuClipper Blog*, accessed Apr. 11, 2025. [Online]. Available: <https://www.docuclipper.com/blog/ocr-accuracy/>
- [2] K. Hiraoka, K. Ohsugi, H. Iso, and K. Fujii, “Context-Independent OCR with Multimodal LLMs: Effects of Image Resolution and Visual Complexity,” *arXiv preprint arXiv:2503.23667*, Mar. 2025. [Online]. Available: <https://arxiv.org/abs/2503.23667v1>
- [3] OpenAI, “GPT-4o Model Documentation,” accessed Apr. 11, 2025. [Online]. Available: <https://platform.openai.com/docs/models/gpt-4o>
- [4] Cradl AI, “Using LLMs for Document OCR: What You Need to Know,” *Cradl AI Blog*, Apr. 09, 2025, accessed Apr. 11, 2025. [Online]. Available: <https://www.cradl.ai/post/llm-ocr>
- [5] Klippa, “LLMs vs OCR Software for Data Extraction: Which One Should You Use?,” *Klippa Blog*, Mar. 03, 2025, accessed Apr. 11, 2025. [Online]. Available: <https://www.klippa.com/en/blog/information/llms-vs-ocr-software/>
- [6] J. Nelson, “The Best OCR Models for Text Recognition in 2024,” *Roboflow Blog*, accessed Apr. 11, 2025. [Online]. Available: <https://blog.roboflow.com/best-ocr-models-text-recognition/>
- [7] Mistral AI, “Mistral OCR: Setting a New Standard in Document Understanding,” *Mistral AI News*, accessed Apr. 11, 2025. [Online]. Available: <https://mistral.ai/news/mistral-ocr>
- [8] Google, “Gemini API Documentation,” accessed Apr. 11, 2025. [Online]. Available: <https://cloud.google.com/gemini/docs>
- [9] Klippa, “Are LLMs Good at Data Extraction? A Comparison with OCR Software,” *Medium*, Mar. 17, 2025, accessed Apr. 11, 2025. [Online]. Available: <https://medium.com/@Klippa/are-llms-good-at-data-extraction-a-comparison-with-ocr-software-6f6c3ad89763>
- [10] Microsoft, “Azure Document Intelligence (OCR) Overview,” accessed Apr. 11, 2025. [Online]. Available: <https://learn.microsoft.com/en-us/azure/ai-services/document-intelligence/overview?view=doc-intel-4.0.0>