WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Paddle OCR vs. Tesseract: A Comparative Performance Analysis on the Gujarati Script

Vedant Sharma, Shyamal Jani, Riya Jain, Kartik Hajela, Abhi Shah, Prof. Janki Tejas Patel

Department Of Engineering, SAL College Of Engineering, Ahmedabad - 380060

ABSTRACT:

The digitization of documents is essential for preserving cultural heritage and improving information access, yet many of India's languages remain digitally **under-represented**, lacking the large-scale datasets required for robust machine learning models (Joshi P., 2020). This paper investigates effective OCR strategies to bridge this gap, using **Gujarati as a representative case study**. A quantitative comparison was conducted between the widely-used Tesseract and the modern deep-learning-based Paddle OCR on a custom dataset of 250 printed text images. Performance was measured using Character Error Rate (CER), Word Accuracy, and word-level F1-Score (Souza J. & Kumar A., 2018). The results demonstrate a significant performance disparity, with **Paddle OCR achieving a remarkably low CER of 4.5% compared to Tesseract's 18.2%**. These findings suggest that modern deep-learning architectures are a powerful tool for improving OCR accuracy and digital accessibility for under-represented languages (Nagdev K. & Sharma V., 2022).

Keywords: OCR, Deep Learning, Under-Represented Languages, Gujarati, Paddle OCR

Introduction:

Making information accessible to everyone in the digital age means we need to digitize texts from all languages. For many of India's diverse languages, however, this is a huge challenge. There simply isn't enough high-quality digital data for them, which holds back the technology meant to help. In this paper, we tackle this problem head-on. We explore how well modern tools can handle the complexities of Indian scripts, using Gujarati as our main case study. By comparing an established tool with a newer, deep-learning-based one, we aim to find a clear path forward for digitizing our rich linguistic heritage.

The idea of teaching computers to read isn't new. For years, the go-to open-source tool for Optical Character Recognition (OCR) has been Tesseract.[1] It's a powerful engine, the result of decades of development, and it has become a standard benchmark for many languages. It represents a mature and reliable approach to a very difficult problem. But as with any technology, the question is always: what comes next?

The recent revolution in deep learning has completely changed the game. Instead of relying on older methods, modern OCR systems like Paddle OCR use architectures that learn to read in a way that's more intuitive. They use neural networks to automatically recognize visual patterns in text, much like a person does, and then figure out the sequence of characters. This represents a fundamental shift in approach, one that promises to be more flexible and powerful, especially for scripts with complex visual rules.

To see how these two approaches stack up in the real world, we put them to the test. We created a custom dataset of printed Gujarati text and ran both Tesseract and Paddle OCR on it. We then measured their performance using standard, objective metrics like Character Error Rate (CER) and Word Accuracy to get a clear picture of which engine is more accurate and reliable for this kind of task.

Ultimately, this comparison is about more than just technology. Getting OCR right is crucial for preserving the cultural heritage found in millions of books and documents here in Gujarat and across India. The success of these projects depends entirely on a machine's ability to correctly interpret the unique features of our scripts, from complex characters to intricate vowel *Matras*. This study offers a clear, practical answer to which technology is better equipped for that vital mission.

What is Optical Character Recognition (OCR)?

Optical Character Recognition is the use of science and technology to convert images of typed, handwritten, or printed text into machine-readable text data.[2] While humans have been interpreting text for millennia, the effort to automate this process has been a systematic goal since the dawn of the computer age. Early OCR systems were developed for specific tasks like sorting postal mail and digitizing books for the visually impaired.[3]

Today, OCR is predominantly based on computer models that take into account the complex visual features of characters and words.[4] This is a departure from older methods that relied on manually matching pixel patterns to a known library of characters (template matching). Modern OCR is powered by artificial intelligence, specifically deep learning models that analyse an image, identify lines of text, segment words, and recognize the individual characters within them.[5] The accuracy of these models depends heavily on the quality of their training data and their underlying architecture.

The inherent difficulty of "reading" contributes to OCR's inaccuracy. The chaotic nature of real-world documents—including complex layouts, unusual fonts, poor image quality from scanning, and the inherent complexity of scripts like Gujarati—all pose significant challenges. An incomplete understanding of how to model all these variations means that OCR predictions can contain errors. Consequently, as the quality of the source document decreases, the accuracy of the extracted text often becomes less reliable. Using advanced models and high-quality images helps to reduce this error and increase confidence in the output.

Why is High-Accuracy OCR Important?

The purpose of OCR is to provide information that people and organizations can use to unlock the value trapped in non-digital documents.[6] This improves societal advantages, including the preservation of life and property, advancements in public health and education, and overall economic prosperity and quality of life.[7] High accuracy is the single most critical factor in achieving these goals.

OCR has a vast range of applications.[8] For institutions like the Gujarat Vidhya Sabha or local university libraries here in Ahmedabad, high-accuracy OCR is essential for preserving priceless historical manuscripts and books, making them searchable and accessible to scholars and the public worldwide. In business, it's the engine behind automating invoice processing, digitizing legal contracts, and processing bank cheques, which saves millions of hours of manual labour. Governments use it to digitize public records, making governance more transparent and efficient. On a daily basis, many people use mobile apps with OCR to instantly translate a sign in a foreign language or capture text from a business card.[9] In all these cases, the usefulness of the final output is directly tied to the accuracy of the initial recognition.

Methodology:

As larger digital datasets become available and machine learning technology advances, the science of Optical Character Recognition improves. The data for OCR comes from a wide array of sources, including high-resolution library scanners, office flatbed scanners, mobile phone cameras, PDFs, and even real-world images from vehicle cameras. There are two primary philosophies for text recognition used by computer scientists: **template-based** and **learning-based**, both of which have various sub-methods. A template-based prediction is deterministic, attempting to find an exact match for a character from a known library, like trying to fit a puzzle piece.

A learning-based prediction, in contrast, is probabilistic. It analyses the features of a character or word and predicts the most likely output based on what it has learned from vast amounts of data. This is the foundation of all modern AI-powered OCR. For this particular study, we adopted this modern, learning-based approach and designed a direct comparative experiment. The specific methodology is as follows:

Dataset:

A custom dataset was curated to serve as a representative sample for this study. It consists of **250 digital images** containing printed Gujarati text sourced from modern books, newspapers, and online articles. Each image was paired with a manually verified ground truth text, stored in a ground_truth.csv file.

Experimental Setup:

The experiment was conducted within a Google Colab environment. The key software components were:

- **Tesseract:** Version 5.3.3, utilizing the tesseract-ocr-guj language pack.
- **Paddle OCR:** Version 2.7.3, using the pre-trained gu (Gujarati) language model.
- **Libraries:** Pytesseract, Jiwer for metric calculation, and Pandas for data manipulation.

Evaluation Metrics:

We used a set of standard metrics to evaluate OCR quality from different perspectives:

- **Character Error Rate (CER):** Measures the raw character-level mistakes. **A lower CER is better.**
- **Word Accuracy:** Calculated as 1 - Word Error Rate (WER). **A higher Word Accuracy is better.**
- **Word-Level Recall, Precision, and F1-Score:** The F1-Score provides the most robust single measure of performance, balancing the ability to find correct words (Recall) with not introducing incorrect ones (Precision). **A higher F1-Score is better.**

Background: A Look at Different OCR Methods

To understand the significance of the tools used in this experiment, it is helpful to review the different methods of OCR that have been developed over time.

Matrix Matching Method

The matrix matching approach is one of the earliest and simplest methods. It relies on a library of pre-stored templates for every character. For example, a captured image of the letter "A" is compared pixel-by-pixel against the stored templates for "A", "B", "C", and so on. While simple, this method is not robust against variations in fonts, sizes, and styles.

Feature Extraction Method

This approach is more sophisticated, identifying key geometric features of a character, such as loops, lines, intersections, and curves. Instead of matching a whole shape, the system would be programmed to look for a pattern of two parallel vertical lines connected by a horizontal line to identify an "H". This is more flexible than matrix matching but can be brittle if the features vary even slightly.

Deep Learning Prediction Method

This is the modern, state-of-the-art approach used by Paddle OCR and, to an extent, in the newer versions of Tesseract. These systems use deep neural networks to generate text predictions based on visual parameters learned directly from images. The CRNN (Convolutional Recurrent Neural Network) model is a prime example. The Convolutional (CNN) part acts like a visual cortex, automatically learning important features from raw pixels. The Recurrent (RNN) part then reads these features in sequence to predict the word. While not perfect, this method provides the best overall accuracy for most OCR tasks today.

Objective:

1. To research a variety of OCR strategies for recognizing text from images.
2. To accurately predict the text content of documents written in an under-represented Indian language.
3. To provide a clear, evidence-based platform for comparing OCR engine performance.

Results

The quantitative evaluation conducted on the custom dataset of 250 Gujarati text images reveals a significant and consistent performance advantage for Paddle OCR over the Tesseract engine. The aggregated results from the experiment are presented in **Table 1**, which summarizes the performance across five key metrics. These findings provide a clear picture of each engine's capabilities when faced with the complexities of a printed, under-represented Indian language.

Table: Aggregated Performance Metrics Summary

Metric	**Tesseract**	**Paddle OCR**
Character Error Rate (CER)	0.182	0.045
Word Accuracy	0.758	0.921
Precision (Word-Level)	0.810	0.945
F1-Score (Word-Level)	0.797	0.938

The analysis of the data identified several key dimensions of performance. First and foremost, **character-level accuracy** was identified as a crucial factor for the overall usability of the OCR output. In this area, the disparity was stark. Paddle OCR achieved a mean Character Error Rate (CER) of just **4.5%**, indicating a very low frequency of mistakes. This stands in sharp contrast to Tesseract's CER of **18.2%**, which is approximately four times higher. This finding suggests that Tesseract struggled significantly more with the nuanced shapes and diacritics (*matras*) of the Gujarati script. This fundamental advantage for Paddle OCR extends to the word level, where it achieved a Word Accuracy of **92.1%** compared to Tesseract's **75.8%**. **Textually, this represents a performance gap of over 16 percentage points in Word Accuracy, clearly establishing the superior capability of the deep learning model.**

Beyond simple accuracy, the **quality of word recognition**, measured by Precision, Recall, and F1-Score, was also analysed. These metrics provide deeper insight into each engine's reliability. Paddle OCR's high word-level Recall (**93.2%**) shows it was highly effective at finding the majority of the original words in the text. Furthermore, its high Precision (**94.5%**) indicates that it did not frequently "hallucinate" or invent incorrect words. Tesseract was less effective on both fronts. The most telling metric, the F1-Score, which balances Precision and Recall, was **0.938** for Paddle OCR versus **0.797** for Tesseract.

This F1-Score differential of 0.141 confirms a substantial improvement in overall recognition quality, proving that Paddle OCR is not only more accurate but also more reliable in its textual output.

These results strongly suggest that **modern deep learning architectures are a highly effective tool** for improving OCR accuracy for under-represented languages. Paddle OCR's superior performance indicates that its model has learned more generalized representations of character and word structures, allowing it to adapt effectively to the Gujarati script. Its underlying CRNN architecture is purpose-built to extract visual features and interpret them as a sequence, a method that proves far more robust than the older approaches.

The key takeaway is that the architectural advantage of modern OCR engines may be a critical factor for success when dealing with languages that have limited digital footprints. Their ability to **generalize from vast, multilingual datasets** appears to provide a more robust foundation than traditional models that are more tightly coupled to the availability of extensive, language-specific training data. The implications of this are powerful: this approach could significantly lower the barrier to high-quality digitization for hundreds of other under-represented languages across India and the world, helping to preserve cultural heritage and broaden digital access for millions.

Conclusion

This research has successfully demonstrated that for under-represented Indian languages, the choice of OCR technology is critical, with modern deep learning engines significantly outperforming traditional baselines. Using Gujarati as a representative case study, this work highlights a promising path forward for making digital tools more equitable and accessible across diverse linguistic communities. Accurate text digitization plays a vital role in preserving cultural heritage and ensuring equal access to information, and it is a challenge to arrange large-scale projects without effective and reliable tools. While the complexity of Indic scripts and the scarcity of data make this a difficult problem, this study shows that modern architectures provide a clear and effective solution.

In this study, we conducted a direct quantitative comparison of the traditional Tesseract engine against the deep-learning-based Paddle OCR. By testing both on a custom dataset of printed Gujarati text, we established that Paddle OCR’s architecture delivers a substantially lower error rate and higher overall accuracy. Hopefully, this comparative approach can be used to evaluate and validate OCR solutions for other under-represented languages. We demonstrated that the modern architecture facilitates a more robust and reliable digitization process.

Building on these findings, future research should proceed in several key directions to further validate and extend this work. First, **cross-lingual validation** is essential; replicating this study for other Indian languages such as Odia, Assamese, and Punjabi would confirm if the superior performance of the deep learning engine is a consistent trend across different Indic scripts. Second, the **robustness on degraded documents** needs to be tested by evaluating the models on more challenging historical manuscripts and low-quality scans. Finally, exploring **domain-specific fine-tuning** by investigating how a small, specialized dataset could be used to elevate these models to expert-level performance for specific applications like legal or academic texts is a crucial next step.

The model's output was compared to the established baseline in the field, and the suggested deep-learning approach outperforms it significantly in terms of accuracy. This work provides a clear recommendation for institutions like the **Gujarat Vidhya Sabha**, national libraries, universities, and government archives. By adopting modern OCR architectures, these organizations can dramatically reduce the cost and manual effort associated with their digitization initiatives. Ultimately, this system has numerous applications, not just in large-scale archival projects, but in any domain that seeks to bridge the gap between the printed page and the digital world, ensuring that India's rich linguistic diversity can thrive in our shared digital future.

References:

List all the material used from various sources for making this project proposal

Research Papers:

1. Du, Y., et al. (2020). *PP-OCR: A Practical Ultra Lightweight OCR System*. ArXiv preprint arXiv:2009.09941.
2. Smith, R. (2007). *An Overview of the Tesseract OCR Engine*. Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR).
3. Shi, B., Bai, X., & Yao, C. (2016). *An End-to-End Trainable Neural Network for Image-based Sequence Recognition and its Application to Scene Text Recognition*. ArXiv preprint arXiv:1507.05717.
4. LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). *Gradient-based learning applied to document recognition*. Proceedings of the IEEE, 86(11), 2278-2324.
5. Joshi, P., et al. (2020). *The State and Fate of NLP in Thirty-Two Indian Languages*. Proceedings of the 1st Conference on Language, Data and Knowledge.
6. Ghosh, S., & Das, A. (2020). *A Survey of OCR Techniques for Indic Scripts*. ACM Transactions on Asian and Low-Resource Language Information Processing.

7. Kumar, R., & Singh, P. (2021). *A Deep Learning Approach for OCR of Devanagari Script Documents*. Proceedings of the International Conference on Machine Learning and Data Science.

8. Hochreiter, S., & Schmidhuber, J. (1997). *Long short-term memory*. Neural Computation, 9(8), 1735-1780.

9. Chaudhuri, B. B., & Pal, U. (2009). *An OCR system to read two Indian language scripts: Bangla and Devanagari (Hindi)*.

10. Nagdev, K., & Sharma, V. (2022). *Comparative Analysis of OCR Tools for Digitally Under-Represented Scripts*. Journal of Digital Humanities.

11. Souza, J., & Kumar, A. (2018). *Evaluating OCR Performance: A Study on Modern Metrics for Document Digitization*.

12. Joshi, N., et al. (2022). *Fine-tuning Pre-trained Models for Low-Resource Indic Language OCR*.

13. Goyal, M., & Lehal, G. S. (2010). *A survey of methods for feature extraction for OCR of Indian scripts*.