PsOCR: Benchmarking Large Multimodal Models for Optical Character Recognition in Low-resource Pashto Language

Ijazul Haq
Shien-Ming Wu School of Intelligent Manufacturing,
South China University of Technology,
Guangzhou, 511442, China.
Zirak.ai

Yingjie Zhang
Shien-Ming Wu School of Intelligent Manufacturing,
South China University of Technology,
Guangzhou, 511442, China

Irfan Ali Khan
School of Science and Engineering,
University of Missouri-Kansas City,
Missouri, 64110, USA

Abstract

This paper evaluates the performance of Large Multimodal Models (LMMs) on Optical Character Recognition (OCR) in the low-resource Pashto language. Natural Language Processing (NLP) in Pashto faces several challenges due to the cursive nature of its script and a scarcity of structured datasets. To address this, we developed a synthetic Pashto OCR dataset, PsOCR, consisting of one million images annotated with bounding boxes at word, line, and document levels, suitable for training and evaluating models based on different architectures, including Convolutional Neural Networks (CNNs) and Transformers. PsOCR covers variations across 1,000 unique font families, colors, image sizes, and layouts. A benchmark subset of 10K images was selected to evaluate the performance of several LMMs, including seven open-source models: DeepSeek's Janus, InternVL, MiniCPM, Florence, and Qwen (3B and 7B), and four closed-source models: GPT-4o, Gemini, Claude, and Grok. Experimental results demonstrate that Gemini achieves the best performance among all models, whereas among open-source models, Qwen-7B stands out. This work provides an insightful assessment of the capabilities and limitations of current LMMs for OCR tasks in Pashto and establishes a foundation for further research not only in Pashto OCR but also for other similar scripts such as Arabic, Persian, and Urdu. PsOCR is available at `https://github.com/zirak-ai/PashtoOCR`.

***K*eywords** OCR · LMMs · VLMs · NLP · Datasets · Benchmarks · Computer Vision

1 Introduction

Optical Character Recognition (OCR) is essential for converting scanned and image-based documents into machine-readable text, underpinning digital archiving, automated indexing, and large-scale document analytics. Traditional OCR engines rely on hand-crafted rules and language-specific resources, while modern deep-learning approaches, using convolutional neural networks and transformer architectures, deliver remarkable performance. However, these methods depend heavily on extensive annotated corpora and lexicons, which are scarce or nonexistent for many languages [1]. As a result, OCR performance degrades sharply for low-resource languages, highlighting the need for tailored datasets and models. In recent years, the research community has begun addressing these gaps: [2] has worked on 60 low-resource languages; [3] has focused on handwritten Tamil, Kurdish, Swahili, and Amharic scripts. Similarly, Ethiopian [4], Indic languages [5], [6] and Khmer [7] have also been explored.

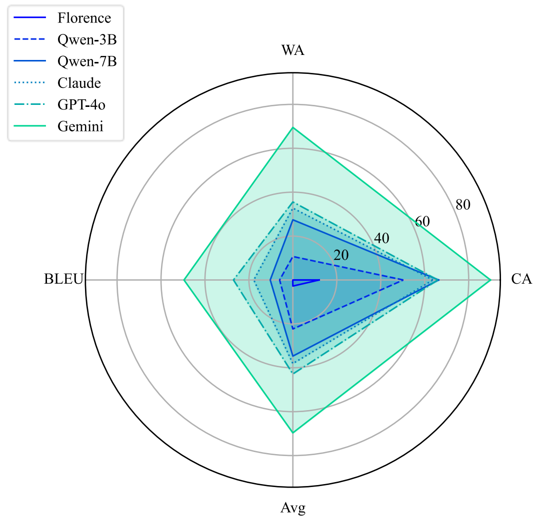


Figure 1: Performance Comparison of the top-6 LMMs on Pashto OCR

Pashto is an Indo-European language of the Perso-Arabic script family, spoken by over 50 million people worldwide. It is the official language of Afghanistan, and the second-largest language of Pakistan by the number of native speakers [8]. Written in cursive script from right to left (RTL), Pashto comprises 44 letters that can take up to four contextual forms: initial, medial, final, and isolated, which raises considerable challenges for OCR [9]. Pashto also makes use of ligatures and diacritical marks, expanding its range of glyph shapes [10]. Furthermore, there are no clear word boundaries in Pashto, and challenges arise from the inconsistent use of diacritical marks that further complicates the training of OCR models to recognize diverse orthographic conventions [11], [12], [13].

Synthetic data has emerged as an invaluable resource for addressing data scarcity issues in OCR research, particularly for low-resource languages and scripts. Generating synthetic datasets allows researchers to develop and benchmark robust AI models without the substantial time and resource investments required for manual annotation. This approach has proven effective across diverse languages and scripts; for instance, [14] demonstrated that incorporating synthetic data alongside real-world data can lead to performance improvements. Similarly, [15] introduced a comprehensive synthetic dataset for 23 Indic languages, significantly improving accuracy through fine-tuning on synthetic samples. Moreover, [16] applied synthetic data generation techniques for Arabic OCR, facilitating extensive system evaluation and comparative studies. These studies underscore the importance and effectiveness of synthetic data in enhancing OCR model performance.

In this work, we introduce PsOCR, a comprehensive dataset for training and evaluating LMMs on Pashto OCR tasks. PsOCR is composed of one million synthetic images, annotated at word, line, and document levels, featuring around a thousand unique font families, varied color schemes, and diverse layouts. A curated subset of 10K images is used as an evaluation benchmark to examine the performance of various flagship LMMs. We have evaluated a total of eleven models, including seven open-source models: InternVL [17], MiniCPM [18], Llama [19], Janus [20], Florence [21], and Qwen (3B and 7B) [22], and four proprietary models: Grok [23], Claude [24], GPT-4o [25], and Gemini [26]. Detailed experimental analyses reveal each model's strengths and weaknesses in Pashto text extraction, paving the way for further research in this domain.

The key contributions of this study are as follows:

- Introduced the first publicly available comprehensive Pashto OCR dataset consisting of one million synthetic images annotated at word, line, and document-level granularity, covering extensive variations including 1000 unique font families, diverse colors, image sizes, and text layouts.
- Developed the first publicly available OCR benchmark comprising 10K images, facilitating systematic evaluation and comparison of OCR systems for the low-resource Pashto.
- Conducted a pioneering evaluation and comparison of state-of-the-art LMMs on Pashto OCR, providing crucial insights into their zero-shot capabilities, strengths, and limitations for low-resource languages written in Perso-Arabic scripts.

2 Related Work

2.1 Pashto OCR

One of the foundational works in Pashto OCR is [10], identifying the primary recognizable units in Pashto script. They suggested "ligature" as the basic unit for OCR and claimed that 7,681 basic shapes were adequate for representing all Pashto ligatures, thus simplifying recognition challenges. [27] proposed a robust OCR method addressing scale, rotation, and location invariances in Pashto script; they developed a holistic recognition framework and created a dataset of 8K images covering 1K unique ligatures. Recognizing dataset availability as a significant bottleneck, [28] introduced the KPTI database containing 17K handwritten and printed text-line images, facilitating benchmarking of deep-learning models. [29] used Sequential Minimal Optimization with horizontal and vertical projections for line, word, and character segmentation and Local Binary Patterns for feature extraction; this model achieved satisfactory performance and emphasized the importance of precise segmentation and robust feature extraction. [30] leveraged transfer learning with pre-trained CNNs, highlighting the efficacy of fine-tuned deep-learning models augmented by data-enhancement techniques. [31] developed a CNN-based classifier specifically targeting handwritten numerals, capable of handling variations in style and orientation. Similarly, [32] and [33] used CNNs and developed their own benchmark datasets for evaluation. [34] tested feed-forward neural networks with varying ReLU layers on a custom dataset, providing crucial insights into neural architectures, while [32] used LSTM for character recognition. [35] developed the publicly accessible HPCID dataset, comprising 15K handwritten samples, offering critical resources for training and evaluation. Lastly, [36] created the PHTI dataset, encompassing 36K segmented text-line images from diverse genres and writers, filling a substantial gap in available Pashto handwritten resources.

2.2 Synthetic OCR Datasets

Synthetic OCR datasets address annotated-data scarcity in low-resource languages. [16] developed a method for generating synthetic Arabic OCR datasets, addressing unique script characteristics, facilitating comprehensive system testing and model comparisons. [37] examined synthetic data methods for post-OCR correction, proposing a technique using computer vision-based glyph similarity algorithms. [14] investigated synthetic data's role in enhancing OCR model performance on the SROIE dataset; by combining synthetic data with real-world data, their model achieved approximately 32% performance improvement. [15] introduced a large-scale synthetic OCR benchmark dataset for 23 Indic languages, featuring varied fonts, sizes, colors, and backgrounds. Fine-tuning OCR models on their dataset improved accuracy by approximately 1%. [38] proposed RoundTripOCR, a synthetic data generation technique for post-OCR error correction in low-resource Devanagari languages, significantly enhancing OCR accuracy. [39] and [40] developed synthetic datasets for text localization in natural images.

2.3 OCR Benchmarks for LMMs

Under the large umbrella of OCR, various benchmarks are available for different tasks, such as for Visual Question Answering (VQA): TextOCR [41], TextCaps [42], ST-VQA [43], OCR-VQA [44], TextVQA [45], DocVQA [46], InfographicVQA [47], ChartQA [48], MTVQA [49], MultipanelVQA [50], EST-VQA [51]; for rich-text image understanding: LLaVAR [52], MMR [53]; and for reasoning: MM-GNN [54], textKVQA [55], OCRBench [56]. Various other benchmarks are available for specific purposes, such as KITAB-Bench [57] and CAMEL-Bench [58] for Arabic OCR tasks, MOTBench [59] for menu understanding and translation, CC-OCR [60] for literacy, and Fox [61] for fine-grained and multi-page document understanding. However, there is no significant prior work comparable to ours, which specifically focuses on benchmarking LMMs for text extraction in the low-resource Pashto language.

3 Dataset Development

3.1 Text Corpus Collection

Pashto has limited structured textual content available for large-scale NLP and OCR tasks. Developing an OCR dataset at a significant scale requires a substantial and high-quality raw text corpus. To address this requirement, we collected Pashto text from three primary sources. Firstly, we attempted extraction from the Common Crawl (CC) [1] corpus. Although CC is extensive, the availability of Pashto text is extremely low; filtering a 15,000 GB dataset specifically for Pashto (ISO-639-3 language code: "pus") yielded only ≈1GB of text, constituting about 0.008% of the entire corpus. This indicates that relying solely on the CC corpus is inefficient for large-scale Pashto text collection. Secondly, we crawled open-source websites abundant in Pashto content, significantly supplementing our corpus. Third, we

[1] Common Crawl: `https://commoncrawl.org`

incorporated existing text resources from Twitter, books, and news websites used in previous studies [11], [62], [63]. By combining these three data sources, we created a text corpus, sufficient for building a comprehensive OCR dataset.

3.2 Text Cleaning and Preprocessing

The text corpus underwent rigorous data cleaning and preprocessing steps to ensure its quality and usability. Initially, we removed extraneous data elements such as URLs, HTML tags, and longer chunks of foreign language text. Following this, normalization processes were applied to very large numerical values, repetitive line breaks, excessive spaces, emojis, and other special characters. Nevertheless, we intentionally maintained a controlled threshold of noise within the corpus rather than achieving absolute cleanliness. The rationale behind preserving minor noise was to enhance the versatility and robustness of OCR models trained on this dataset, enabling them to handle and generalize better during inference on real-world, imperfect text samples. Ultimately, the corpus was segmented into one million text chunks, where each chunk varied from as short as one sentence to as long as several paragraphs, facilitating diverse textual representation.

3.3 Text to Image Conversion

To convert textual content into images suitable for OCR model training, we adopted an automated method using Python scripts. Initially, each of the one million Pashto text chunks was programmatically converted into individual HTML pages. Subsequently, diverse yet controlled random styling, via Cascading Style Sheets (CSS), was applied to each HTML page utilizing Python and JavaScript scripts. This variability in style simulated realistic document formatting scenarios, significantly increasing dataset diversity. Finally, we utilized the Selenium[2] library to render these styled HTML pages. Each HTML page was captured as a PNG image screenshot, resulting in one million images of varying dimensions, aspect ratios, and visual styles, closely mimicking real-world document variability and complexities.

3.4 Dataset Composition

3.4.1 Granularity

The PsOCR dataset was explicitly designed with the architectural diversity of AI models in mind, including both CNNs and Transformer-based architectures. The annotation information is provided at three levels of granularity: page-level, line-level, and token-level, as shown in Figure 2. Each annotation includes precise bounding boxes (bbox), characterized by four numerical attributes: (X, Y, width, height) as shown in Figure 3. Here, the coordinates (X, Y) represent the top-left corner of each bounding box, while width and height denote the respective horizontal and vertical dimensions measured in pixels. Page-level annotations comprise a single bounding box encapsulating all text present on an image page. Line-level annotations include individual bounding boxes for each distinct text line within a page. Similarly, token-level annotations define bounding boxes around every space-separated chunk of characters. This structured and rich annotation schema significantly enhances the dataset's applicability across various OCR scenarios and training methodologies, supporting diverse granularity-focused tasks.

3.5 Font Variation

Recognizing that font characteristics substantially influence OCR performance, we emphasized extensive font diversity within our dataset. Initially, approximately 3,000 Pashto-compatible font families were collected from publicly accessible sources. We carefully reviewed and filtered these fonts, removing any proprietary or non-freely distributable ones. Additionally, we manually inspected and removed fonts that were difficult to read. Duplicate fonts and those nearly identical in style and appearance were also removed to avoid redundancy. This rigorous selection process resulted in the inclusion of 1,000 distinct font families; some examples are shown in Figure 4. Furthermore, we ensured font sizes in the dataset ranged between 11px and 30px, providing suitable variability. We also varied font width by controlling text boldness through CSS numerical values ranging from 600 to 900. Such comprehensive font variation makes our PsOCR dataset a robust resource capable of training OCR models adept at handling extensive font-related variability encountered in practical OCR tasks.

3.6 Images Size and Aspect Ratio

Image size and aspect ratio are critical parameters for simulating realistic OCR scenarios and significantly influence model performance. To control variability, we predefined image widths via CSS, randomly selecting from 200, 300,

[2]Selenium: `https://pypi.org/project/selenium/`

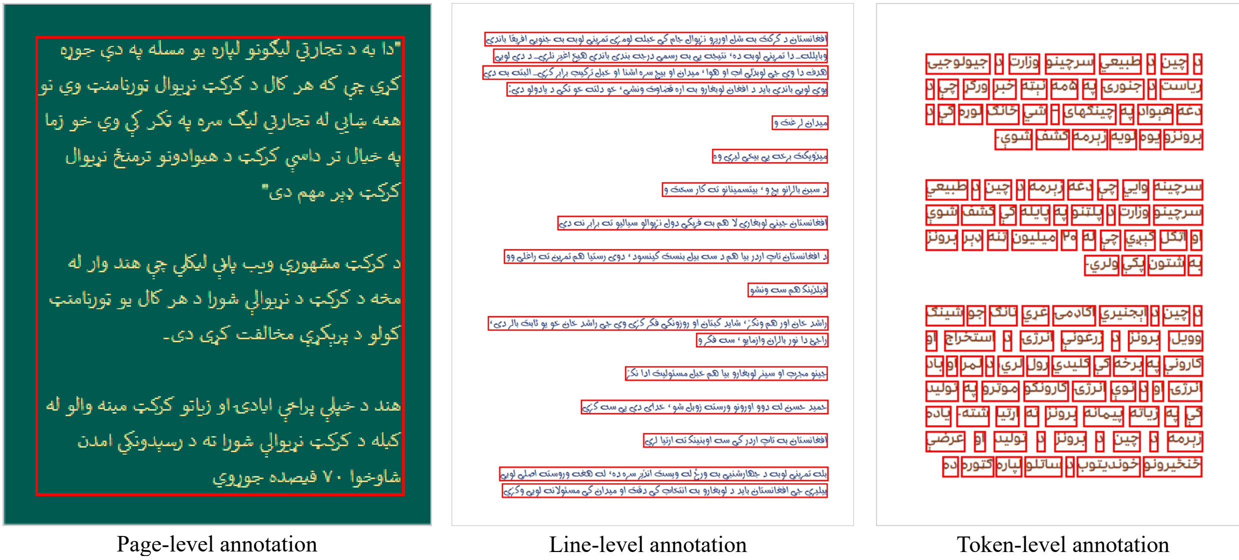


Figure 2: Sample images from the dataset showing different levels of granularity and bounding box annotation

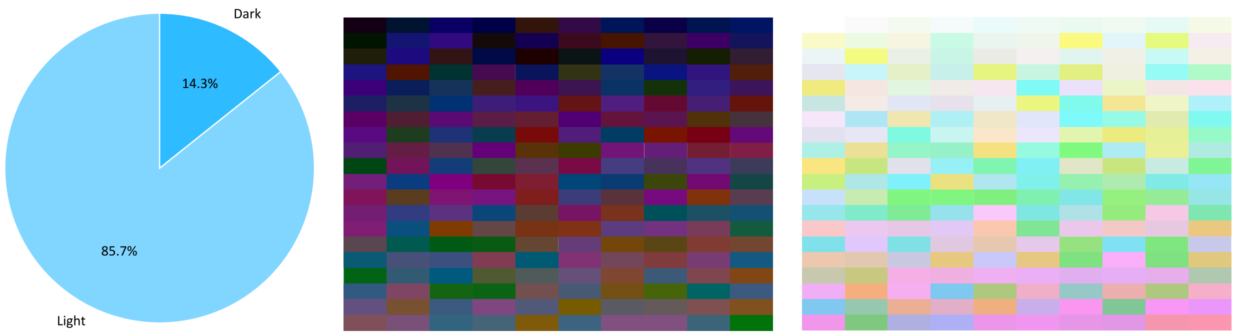
Figure 3: A sample image from the dataset in PNG format, with the corresponding details and annotation information in JSON format

400, 500, 600, 700, and 800px, while heights were set to "auto," resulting in variations based on text length, font size, line height, and the number of line breaks. The final rendered width and height include additional padding around the text area, slightly exceeding the original text element dimensions. Consequently, the images exhibit diverse aspect ratios. Figure 6 (A) shows the distribution of image sizes, Figure 6 (C) the histogram of aspect ratios, and Figure 6 (B) the histogram of image file sizes, which affects storage requirements.

3.7 Themes and Colors

The PsOCR dataset encompasses a broad spectrum of color combinations, crucial for improving model adaptability to various visual environments. Primarily, two color themes were included: "Dark" and "Light"; in the "Dark" theme, a

Figure 4: A few examples of font families in PsOCR dataset



(A) Themes (B) Dark Colors (C) Light Colors

Figure 5: Colors and themes in the dataset and their ratio

darker background is paired with lighter-colored text, and vice versa. Figure 5 (A) depicts the thematic distribution across the dataset. The dataset contains a total of 232 dark and 271 light colors; the top 200 most frequent dark and light colors are shown in Figure 5 (B) and (C), respectively. Given these color counts, theoretically, approximately 126K unique color combinations could have been generated (2 themes × 232 dark × 271 light). However, the actual number of unique combinations included is around 66K. This discrepancy arises from our controlled color selection strategy designed to ensure optimal readability and visual contrast. Specifically, the following procedure was used for selecting colors for each image:

- **Step 1:** Randomly select a theme for each image. The probability of selecting the "Light" theme was set six times higher than selecting the "Dark" theme, reflecting common readability practices in real-world documents.
- **Step 2:** If the selected theme is Dark, randomly choose a dark color for the background and a light color for the text; if the theme is Light, choose a light background and a dark text color.
- **Step 3:** Calculate the luminance for both selected background and font colors. The luminance ratio is computed by dividing the higher luminance value by the lower luminance value of the color pair.
- **Step 4.** If the luminance ratio is less than 6, discard the selected color pair and return to Step 2. If the luminance ratio is ≥ 6, accept the chosen background and font colors. This contrast threshold (≥ 6) was chosen empirically based on human readability perception across numerous samples.

This method ensures that all included color combinations provide clear visual contrast, facilitating effective recognition by OCR models.

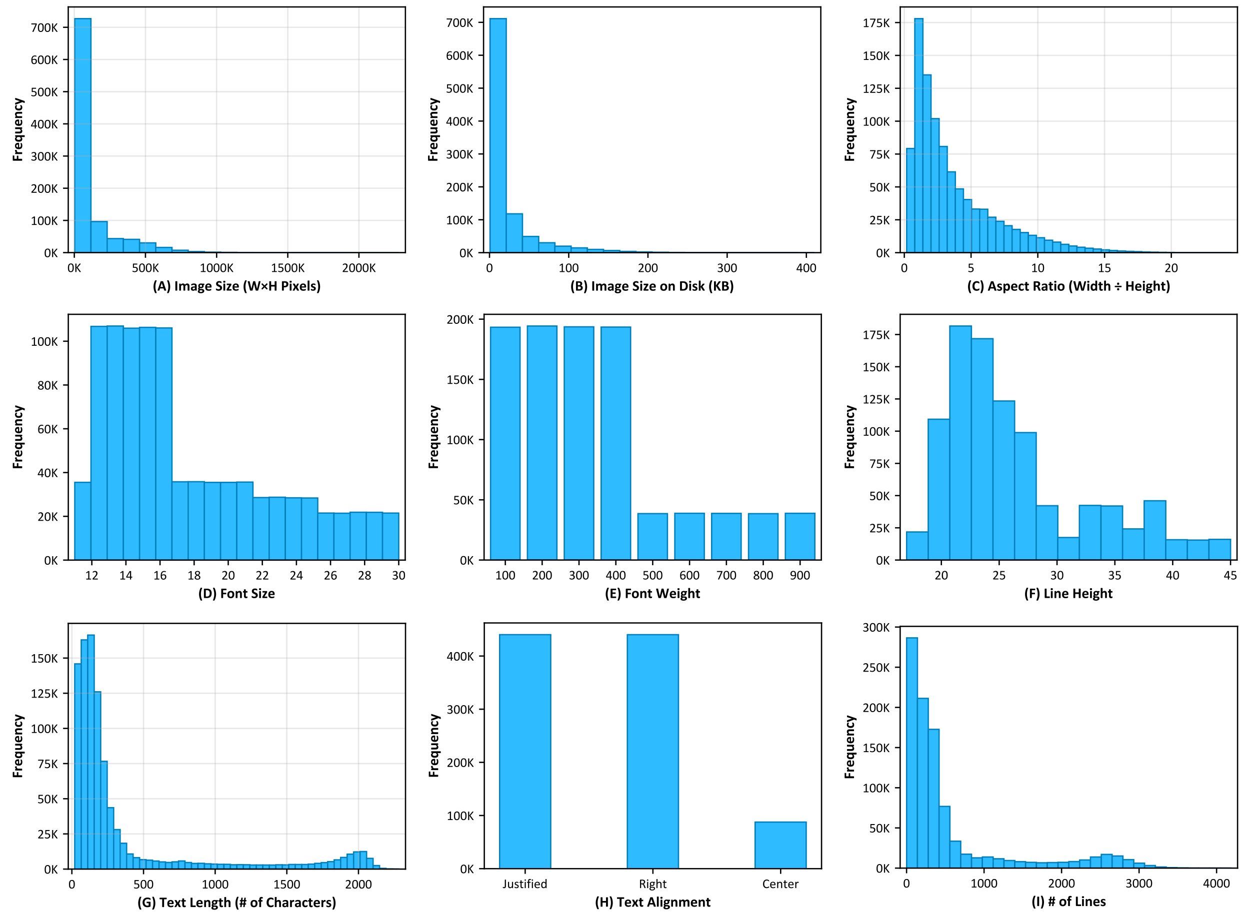


Figure 6: Dataset Statistics

3.8 Other Variations

In addition to the previously described attributes, several other variations were introduced, briefly explained as follows:

- **Padding:** Padding denotes the empty space between the text content and the image borders. We allowed padding values of 10, 20, 30, 40, and 50px for all the images, while padding was limited to 10, 20, or 30px for images narrower than 200px to avoid excessive white space.
- **Text Alignment:** To simulate common document layouts, we randomly applied one of three text alignment options to each document: “right,” “justified,” or “center.” For images using justified alignment, we further varied the final line, aligning it either to the “right” or “center.”
- **Number of Lines and Paragraphs:** The dataset reflects organic variability in text structure by allowing each image to contain an arbitrary number of paragraphs and lines. Paragraph counts follow the natural segmentation of the source text, without artificial splitting or merging.
- **Line Height:** The vertical spacing between consecutive text lines was also varied across the dataset to mirror real-world document formatting.

4 Validation and Evaluation

4.1 Models Selection

This is a pioneering study of LMMs evaluation on any Pashto benchmark, thus limited previous literature exists to guide model selection. Given the rapid growth in the number of available LMMs, careful selection criteria were essential to

Model	Org	Exact Variant	# of Parameters
InternVL [17]	OpenGVLab	InternVL2_5-8B	8.08B
MiniCPM [18]	OpenBMB	MiniCPM-o-2_6	8.67B
Llama [19]	Meta	Llama-3.2-11B-Vision-Instruct	10.7B
Janus [20]	DeepSeek	Janus-Pro-7B	7B
Florence [21]	Microsoft	Florence-2-large	0.77B
Qwen-3B [22]	Alibaba	Qwen2.5-VL-3B-Instruct	3.75B
Qwen-7B [22]	Alibaba	Qwen2.5-VL-7B-Instruct	8.29B
Grok [23]	X-AI	grok-2-vision-1212	-
Claude [24]	Anthropic	claude-3-7-sonnet-20250219	-
GPT-4o [25]	OpenAI	GPT-4o (2024-08-06)	-
Gemini [26]	Google	gemini-2.0-flash	-

Table 1: Details of the Evaluated Models

determine the most relevant models to include in our analysis. We primarily selected models based on their popularity in existing research literature, their documented performance on public leaderboards, and, for open-source models, their download frequency and current trends on platforms such as ModelScope[3] and HuggingFace[4]. Additionally, we ensured all selected open-source variants are of similar size in terms of parameter count; thus, we chose the 7B variants or, if unavailable, those closest to 7B.

In total, this study evaluates eleven LMMs: seven open-source (InternVL, MiniCPM, Llama, Janus, Florence, Qwen 3B, and Qwen 7B) and four proprietary (GPT-4o, Gemini, Claude, and Grok). A concise overview of these models, along with their parent organizations, specific variants, and parameter counts, is summarized in Table 1.

4.2 Experimental Setup

The core objective of this study is to assess the zero-shot OCR performance of selected LMMs on our newly developed PsOCR benchmark. Neither pre-training nor fine-tuning was applied; models were evaluated directly in their original form. The inference pipeline for proprietary models: GPT-4o, Gemini, Claude, and Grok was developed through their respective APIs. Conversely, selected open-source models, including InternVL, MiniCPM, Janus, Llama, and Qwen, were downloaded and tested locally. Decisions regarding the use of APIs versus local inference were driven by model accessibility and available computational resources. For the experiment, we used Python-based tools and libraries such as openai[5], HuggingFace's Transformers[6], Google's GenAI SDK[7], and Anthropic's API[8]. Local inference ran on a PC with an Intel Core i7 processor, 32GB of RAM, and an NVIDIA RTX 4080 GPU. During evaluation, each of the 10K images was provided one by one to each model along with a specifically engineered prompt, and the outputs were recorded. Although most evaluated models support conversational context, our experiment explicitly avoided using histories; the evaluation strictly adhered to a zero-shot setting, providing no contextual examples or prior information, to ensure an unbiased assessment of their innate OCR capabilities.

4.3 Prompt Engineering

A critical factor affecting evaluation results in zero-shot scenarios is the quality and specificity of the instructions provided to the models, known as "prompt." Effective prompt design is essential to achieving accurate model responses. However, no universal standards or guidelines exist for crafting optimal prompts. Different models have been trained and fine-tuned by diverse research teams using varying methodologies, leading to notable differences in how models interpret and respond to prompts. Consequently, a prompt effective for one model may not be equally suitable for another. To address this, we invested considerable effort in designing customized prompts for each model, adopting a trial-and-error approach to iteratively refine instructions and maximize performance. A primary difficulty encountered during prompt engineering involved clearly instructing the models to return only the exact text displayed within the images, without providing additional contextual information, explanations, or translations. Several models demonstrated a tendency to translate the extracted text or include extraneous details. Thus, significant prompt refinement was

[3]ModelScope: `https://www.modelscope.cn`
[4]HuggingFace: `https://huggingface.co`
[5]OpenAI Python API library: `https://github.com/openai/openai-python`
[6]HuggingFace Transformers: `https://github.com/huggingface/transformers`
[7]Google Gen AI SDK: `https://github.com/googleapis/python-genai`
[8]Anthropic API: `https://docs.anthropic.com/en/api/getting-started`

You are an AI assistant tasked with extracting only the text from images. Your goal is to accurately identify and transcribe any text present in the image, without describing or interpreting other visual elements.

Please follow these steps:

1. Transcribe text in its original language.
2. Carefully examine the entire image for any visible text.
3. Extract any text regardless of its size, font, color, or orientation.
4. Maintain the original capitalization, punctuation, and line breaks of the text as they appear in the image.
5. If there are multiple separate text elements, list them in a logical order (e.g., top to bottom, left to right).
6. Do not include any descriptions of the image, its contents, or the context of the text.
7. Focus solely on extracting text. Do not describe or interpret any non-text elements in the image.
8. Include all text, even if it's partially obscured or difficult to read.
9. Only return the extracted text, nothing else.

Figure 7: Prompt for Llama and Claude

You are an OCR engine, specialized in extracting text from images. I will provide you with an image containing text. Your task is to extract the text exactly as it appears in the image. Please ensure that you preserve all characters, punctuation, diacritics, and formatting as accurately as possible. Do not add any additional commentary, explanation, or modifications to the text, and return only the extracted text.

Figure 8: Prompt for all the other models, including GPT4-o, Gemini, and Qwen

necessary to explicitly discourage translation and ensure that the outputs precisely reflected the original Pashto text. Figure 7 and Figure 8 present two prompt examples that yielded the best results in our experiments.

4.4 Post Processing Models' Responses

Despite rigorous efforts in prompt engineering, some models still produced responses that did not strictly adhere to the desired output format, occasionally including additional comments or irrelevant contextual information. Therefore, after collecting predictions from all models, a manual verification step was performed to identify and correct invalid or improperly formatted responses. During this validation, it was observed that most models reliably provided responses in the expected format, except GPT-4o and the Llama model. The GPT-4o model, in particular, occasionally generated erroneous or incomplete responses due to content flagged as inappropriate by its internal safety filters. In the case of GPT-4o, such erroneous responses accounted for approximately 4% of the total benchmark. Similarly, the Llama model demonstrated considerable difficulty in following the instructions, consistently producing responses contaminated with additional commentary. To handle these outputs, a semi-automatic cleaning procedure was employed to remove unnecessary text, isolating the predicted OCR content for accurate comparison with the ground-truth texts.

4.5 Evaluation Matrices

To quantitatively assess the OCR capabilities of the evaluated LMMs, we employed three well-established performance metrics: Character Accuracy, Word Accuracy, and BLEU score, and also computed the average (Avg) of these three metrics.

4.5.1 Character Accuracy

Character Accuracy (CA) measures the accuracy at the character level and is derived from the Character Error Rate (CER), as follows:

$$CA = 1 - CER \tag{1}$$

CER itself is calculated by counting the minimum number of character-level operations (insertions, deletions, substitutions) required to transform the predicted text string into the corresponding reference text string (ground-truth text), normalized by the total number of characters in the reference text:

$$\text{CER} = \frac{\text{Number of Character Errors (Insertions + Deletions + Substitutions)}}{\text{Total Number of Characters in Reference Text}}$$

4.5.2 Word Accuracy

Word Accuracy (WA) evaluates accuracy at the word level and is derived similarly from the Word Error Rate (WER), as follows:

$$WA = 1 - WER \tag{2}$$

The WER calculation involves counting the minimum number of word-level operations (insertions, deletions, substitutions) required to convert the predicted text into the corresponding reference text, normalized by the total number of words in the reference text:

$$\text{WER} = \frac{\text{Number of Word Errors (Insertions + Deletions + Substitutions)}}{\text{Total Number of Words in Reference Text}}$$

4.5.3 BLEU Score

The BLEU score measures the similarity of the predicted text compared to the reference text based on matching word n-grams. A higher BLEU score indicates greater similarity to the reference text; a detailed explanation can be found in [64].

4.5.4 Average

To compress each model's overall performance into a single scalar for easier comparison, we compute an Average (Avg) score by equally weighting CA, WA, and BLEU for each test image, and then averaging these per-image scores across the entire benchmark. Formally:

$$Avg = \frac{1}{n}\sum_{i=1}^{n}\frac{(CA_i + WA_i + BLEU_i)}{3} \tag{3}$$

Where n is the total number of images in the benchmark (10K).

5 Results and Discussion

5.1 Models Performance Comparison

The evaluation of LMMs for Pashto OCR demonstrates substantial variability in performance across different metrics. A summary of the evaluation results is given in Table 2, accompanied by a graphical representation in Figure 9. Among the evaluated models, Gemini consistently achieved superior performance, obtaining the highest CA of 89.92%, WA of 69.5%, BLEU score of 49.54%, and an overall average of 69.65%. Conversely, the InternVL model exhibited the lowest performance, with an average score of -355.32%. Within the group of open-source models, Qwen-7B significantly outperformed the others, achieving a CA of 66.33%, WA of 27.41%, BLEU of 10.30%, and an average score of 34.68%. Among proprietary models, Grok showed the lowest performance, with an average of -16.66%. Notably, apart from Gemini, none of the proprietary models surpassed the open-source Qwen-7B in character accuracy, not even GPT-4o. Another important observation is that all models exhibited notably higher CA compared to WA and BLEU. This suggests that while these models can recognize individual characters, they struggle to generate coherent word sequences and phrases. The findings emphasize that Gemini holds considerable promise for applications demanding high-accuracy OCR without additional training. Meanwhile, for scenarios involving fine-tuning or further model customization, Qwen may be suitable due to its strong baseline performance and open-source availability.

5.2 In-depth Results Analysis:

To gain deeper insights into how various image and text properties influence Pashto OCR performance, we plotted each model's average score against key image attributes, as shown in Figure 10. The following discussion summarizes the effect of each attribute on models' performance.

Effect of Image File Size: In Figure 10 (A), we observe a modest upward trend in model performance as image size increases. While the overall trend is subtle, models such as Claude, Gemini, and Qwen-7B exhibit noticeable performance gains on the largest images (600-700K px).

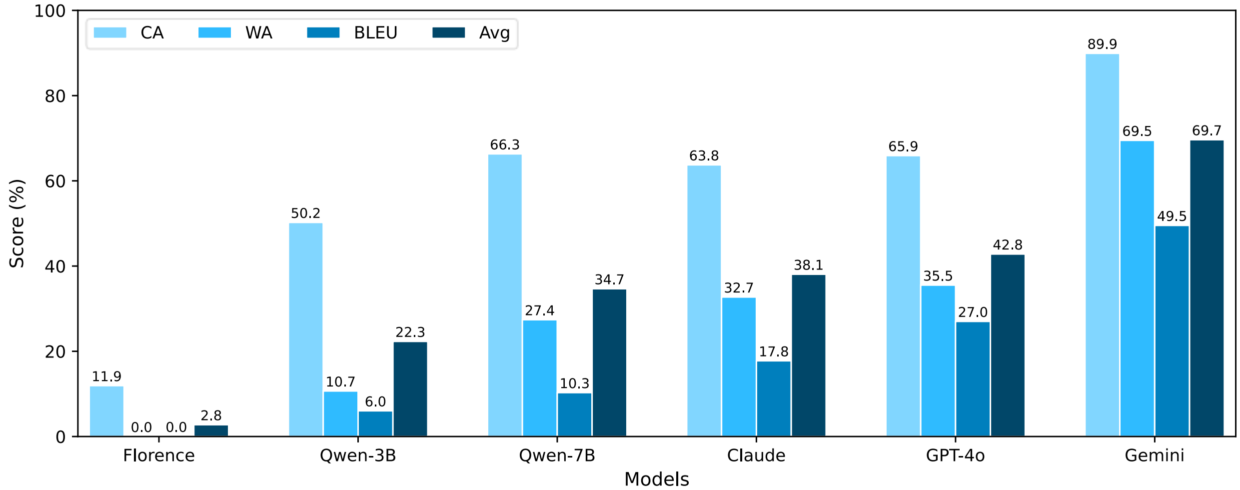


Figure 9: Performance comparison of evaluated LMMs on the Pashto OCR benchmark across different evaluation metrics

Effect of Image Aspect Ratio: Figure 10 (B) shows that aspect ratio exerts minimal influence on OCR accuracy. Nevertheless, all the models demonstrate slightly better performance on wider images compared to taller ones.

Effect of Font Size: Unlike image size and aspect ratio, font size clearly impacts accuracy. All models perform better on images with larger font sizes, with Qwen-3B showing the greatest sensitivity to this factor. Gemini, by contrast, maintains consistently high performance across the font sizes.

Effect of Font Weight: As illustrated in Figure 10 (D), variations in font weight (from light to bold) have negligible impact on model performance. The largely horizontal lines in this plot demonstrate that both thin and heavy typefaces are handled similarly by the models.

Effect of Line Height: Line spacing proved to be one of the most influential factors. Models such as Qwen, Claude, and GPT-4o struggle completely on images with very tight line spacing ($\leq$20px), yielding average scores near zero. Performance improves steadily as line spacing increases.

Effect of Text Alignment: Text alignment shows little overall effect on model accuracy. A slight improvement is visible for “justified” alignment and a minor performance dip for “left” alignment, which aligns with Pashto’s RTL writing direction.

Effect of Text Length: As Figure 10 (G) demonstrates, text length has minimal impact on most models’ performance. An exception is GPT-4o, whose average score declines significantly on images containing longer passages, suggesting that this model may struggle to maintain accuracy for longer text sequences.

Model	CA (%)	WA (%)	BLEU (%)	Avg (%)
InternVL	-457.08	-608.94	0.06	-355.32
Janus	-260.52	-261.86	0.03	-174.12
CPM	-188.44	-193.65	0.84	-127.08
Llama	-30.94	-98.74	5.64	-41.35
Florence-ft	6.29	-12.98	0.03	-2.22
Florence	11.94	-3.64	0.03	2.77
Qwen-3B	50.24	10.68	6.03	22.32
Qwen-7B	66.33	27.41	10.30	34.68
Grok	-12.25	-38.16	0.44	-16.66
Claude	63.75	32.73	17.77	38.08
GPT-4o	65.92	35.53	27.01	42.82
Gemini	89.92	69.50	49.54	69.65

Table 2: Evaluation Results of LMMs on Pashto OCR Benchmark across different matrices

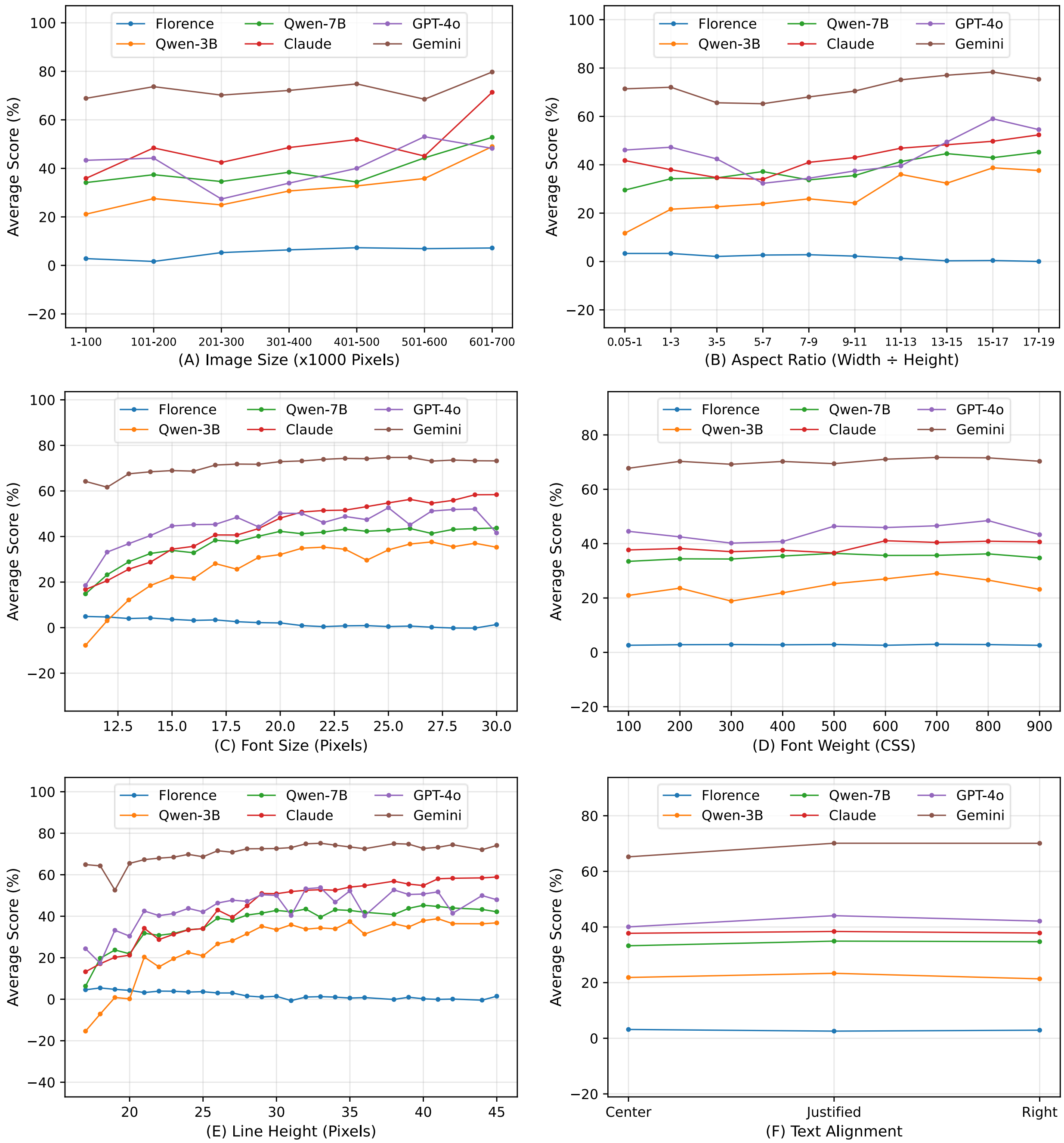


Figure 10: Effect of Image Size, Aspect Ratio, Font Size, Font Weight, Line Height and Text Alignment on Models' Performance

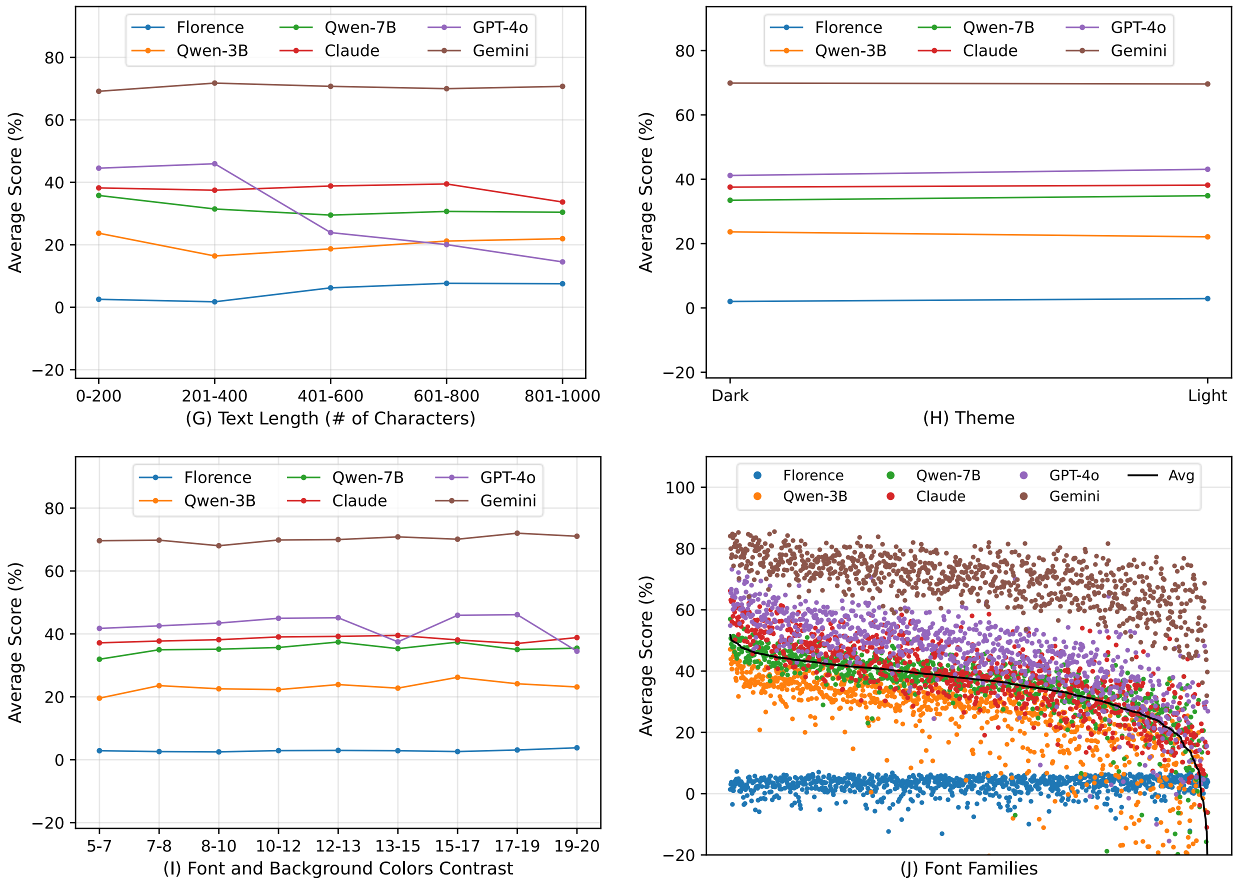


Figure 11: Effect of Text Length, Theme, Colors and Font Families on Models' Performance

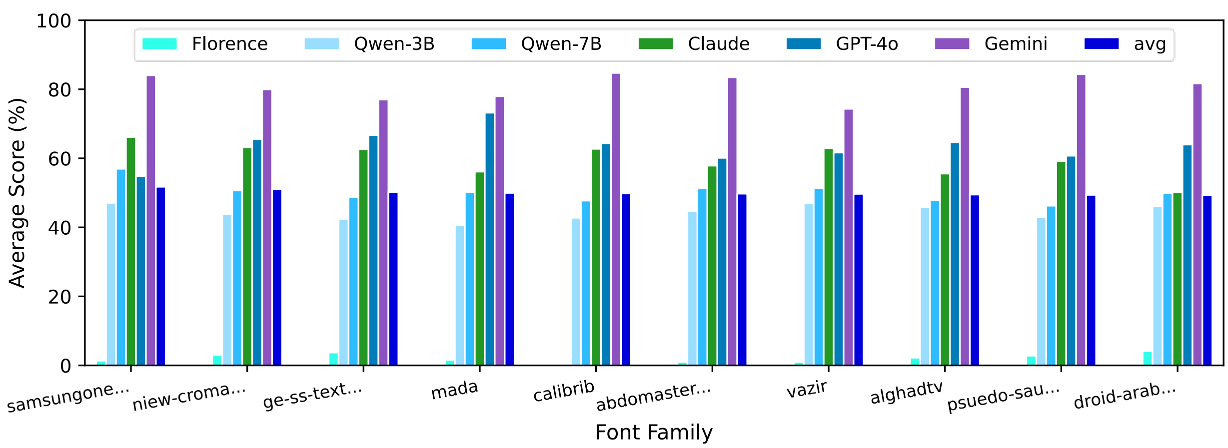


Figure 12: Comparative Analysis of Models' Scores for Top-10 Best-performing Font Families

Rank	Font Name	Avg Score	Font Style
1.	samsungonearabic-400	51.76	دا د پښتو ژبې د لیکلو لپاره یوه ډېره ښکلې فونټ بڼه ده.
2.	niew-cromagnon	51.05	دا د پښتو ژبې د لیکلو لپاره یوه ډېره ښکلې فونټ بڼه ده.
3.	ge-ss-text-medium	50.21	دا د پښتو ژبې د لیکلو لپاره یوه ډېره ښکلې فونټ بڼه ده.
4.	mada	49.98	دا د پښتو ژبې د لیکلو لپاره یوه ډېره ښکلې فونټ بڼه ده.
5.	calibrib	49.81	دا د پښتو ژبې د لیکلو لپاره یوه ډېره ښکلې فونټ بڼه ده.
6.	abdomaster-normal	49.76	دا د پښتو ژبې د لیکلو لپاره یوه ډېره ښکلې فونټ بڼه ده.
7.	vazir	49.70	دا د پښتو ژبې د لیکلو لپاره یوه ډېره ښکلې فونټ بڼه ده.
8.	alghadtv	49.49	دا د پښتو ژبې د لیکلو لپاره یوه ډېره ښکلې فونټ بڼه ده.
9.	psuedo-saudi	49.41	دا د پښتو ژبې د لیکلو لپاره یوه ډېره ښکلې فونټ بڼه ده.
10.	droid-arabic-kufi	49.34	دا د پښتو ژبې د لیکلو لپاره یوه ډېره ښکلې فونټ بڼه ده.

Table 3: List of Top-10 Best-performing Font-families

Effect of Theme and Color Contrast: Figure 10 (H) and (I) reveal that neither overall theme nor specific foreground-background color pairings significantly affect OCR performance.

Effect of Font Family: Font family exerts the strongest influence on OCR accuracy, as shown in Figure 10 (J). Given the dataset's diverse font families, models exhibit wide performance variation. These findings highlight font diversity as one of the primary challenges in Pashto OCR. Figure 11 presents a comparative analysis of models' performance on the top 10 fonts, and Table 3 lists those fonts.

6 Conclusion and Future Work

In this work, we presented the first large-scale evaluation of eleven state-of-the-art LMMs on a newly developed Pashto OCR benchmark. Our dataset, comprising one million synthetically generated images annotated with bounding boxes at page, line, and token levels, represents the first publicly available resource of its kind for the low-resource Pashto language. We assessed seven open-source models (DeepSeek's Janus, InternVL, MiniCPM, Florence, Qwen-7B, and Qwen-3B) alongside four proprietary models (GPT-4o, Gemini, Claude, and Grok) under zero-shot conditions. Experimental results revealed that Gemini delivered the highest overall performance, achieving 89.92% character accuracy, 69.5% word accuracy, and a BLEU score of 49.54%. Among open-source models, Qwen-7B stood out, significantly outperforming other freely available models. These findings underscore the zero-shot capabilities of current LMMs on cursive, complex scripts like Pashto and highlight the promise of open-source models such as Qwen for subsequent fine-tuning.

This study also has some known limitations. First, the dataset consists solely of computer-generated text and does not include handwritten samples. Additionally, the image backgrounds are plain without textures or other natural scenes commonly found in real-world documents. Furthermore, we did not apply image augmentations, such as skewing, rotation, or perspective distortion, which could further challenge model robustness.

Looking ahead, we are extending this work in two key directions. We are currently developing a Pashto VQA dataset and benchmark, and we are also working on the first large-scale handwritten Pashto OCR dataset. Together, these efforts will deepen our understanding of multimodal model performance on low-resource languages and drive future improvements in Pashto document analysis.

7 Data Availability

The PsOCR dataset is divided into two parts: 1) the PsOCR Benchmark, comprising 10K images for evaluation, which is publicly available on HuggingFace[9] and Kaggle[10]; and 2) the PsOCR Train Set, containing approximately one million images, which is available upon request by emailing the corresponding author.

[9] https://huggingface.co/datasets/zirak-ai/PashtoOCR
[10] https://www.kaggle.com/datasets/drijaz/PashtoOCR

References

[1] M. Agarwal, A. Anastasopoulos, A concise survey of ocr for low-resource languages, in: Proceedings of the 4th Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP 2024), pp. 88–102.

[2] O. Ignat, J. Maillard, V. Chaudhary, F. Guzmán, Ocr improves machine translation for low-resource languages, arXiv preprint arXiv:2202.13274 (2022).

[3] R. Sharma, M. Nagpal, Handwritten language detection for low-resource languages using a cnn-bilstm hybrid model, in: 2024 5th IEEE Global Conference for Advancement in Technology (GCAT), IEEE, pp. 1–5.

[4] L. Eyachew, S. Tadesse, B. H. Belay, Language identification in low-resource multilingual document images using deep learning techniques (2024).

[5] A. Ghosh, D. Barman, A. Sufian, I. A. Hameed, Advancing optical character recognition for low-resource scripts: A siamese meta-learning approach with psn framework, IEEE Access (2024).

[6] A. Sarkar, A. Mondal, G. S. Lehal, C. Jawahar, Printed ocr for extremely low-resource indic languages, in: Proceedings of the International Conference on Language Processing (ICLP), Vol. 10, pp. 89–101.

[7] R. Buoy, M. Iwamura, S. Srun, K. Kise, Toward a low-resource non-latin-complete baseline: an exploration of khmer optical character recognition, IEEE Access 11 (2023) 128044–128060.

[8] K. Katzner, K. Miller, The languages of the world, Routledge, 2002.

[9] S. Ullah, T. Enayat, N. Nadeem, I. Din, Y. Saeed, Offline Pashto OCR Using Machine Learning, 2019. `doi:10.1109/iEECON45304.2019.8938859`.

[10] R. Ahmad, M. Z. Afzal, S. F. Rashid, M. Liwicki, A. Dengel, T. Breuel, Recognizable units in pashto language for ocr, in: 2015 13th International Conference on Document Analysis and Recognition (ICDAR), IEEE, pp. 1246–1250.

[11] I. Haq, W. Qiu, J. Guo, P. Tang, Correction of whitespace and word segmentation in noisy pashto text using crf, Speech Communication 153 (2023) 102970.

[12] S. Khan, S. Nazir, Deep learning based pashto characters recognition: Lstm-based handwritten pashto characters recognition system, Proceedings of the Pakistan Academy of Sciences: A. Physical and Computational Sciences 58 (3) (2022) 49–58. `doi:10.53560/PPASA(58-3)743`.
URL `https://ppaspk.org/index.php/PPAS-A/article/view/619`

[13] M. Wahab, H. Amin, F. Ahmed, Shape analysis of Pashto script and creation of image database for OCR, 2009. `doi:10.1109/ICET.2009.5353160`.

[14] Y. Yashchuk, Y. Yurchenko, N. Yurchenko, Y. Andrashko, Enhancing ocr model training with synthetic data: A case study with sroie dataset, in: 2024 IEEE 4th International Conference on Smart Information Systems and Technologies (SIST), IEEE, pp. 482–486.

[15] N. Saini, P. Pinto, A. Bheemaraj, D. Kumar, D. Daga, S. Yadav, S. Nagaraj, Ocr synthetic benchmark dataset for indic languages, arXiv preprint arXiv:2205.02543 (2022).

[16] V. Margner, M. Pechwitz, Synthetic data for arabic ocr system development, in: Proceedings of sixth international conference on document analysis and recognition, IEEE, pp. 1159–1163.

[17] S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang, Qwen2. 5-vl technical report, arXiv preprint arXiv:2502.13923 (2025).

[18] Y. Yao, T. Yu, A. Zhang, C. Wang, J. Cui, H. Zhu, T. Cai, H. Li, W. Zhao, Z. He, Minicpm-v: A gpt-4v level mllm on your phone, arXiv preprint arXiv:2408.01800 (2024).

[19] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, The llama 3 herd of models, arXiv preprint arXiv:2407.21783 (2024).

[20] X. Chen, Z. Wu, X. Liu, Z. Pan, W. Liu, Z. Xie, X. Yu, C. Ruan, Janus-pro: Unified multimodal understanding and generation with data and model scaling, arXiv preprint arXiv:2501.17811 (2025).

[21] B. Xiao, H. Wu, W. Xu, X. Dai, H. Hu, Y. Lu, M. Zeng, C. Liu, L. Yuan, Florence-2: Advancing a unified representation for a variety of vision tasks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4818–4829.

[22] P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, W. Ge, Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution, arXiv preprint arXiv:2409.12191 (2024).

[23] (April 30, 2025 August 13, 2024). [link].
URL https://x.ai/news/grok-2

[24] (2025). [link].
URL https://assets.anthropic.com/m/785e231869ea8b3b/original/claude-3-7-sonnet-system-card.pdf

[25] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, Gpt-4o system card, arXiv preprint arXiv:2410.21276 (2024).

[26] (Dec 11, 2024). [link].
URL https://blog.google/technology/google-deepmind/google-gemini-ai-update-december-2024

[27] R. Ahmad, S. Naz, M. Z. Afzal, S. H. Amin, T. Breuel, Robust optical recognition of cursive pashto script using scale, rotation and location invariant approach, PloS one 10 (9) (2015) e0133648.

[28] R. Ahmad, M. Z. Afzal, S. F. Rashid, M. Liwicki, T. Breuel, A. Dengel, Kpti: Katib's pashto text imagebase and deep learning benchmark, in: 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), IEEE, pp. 453–458.

[29] S. Ullah, T. Enayat, N. Nadeem, I. ud Din, Y. Saeed, M. Junaid, Offline pashto ocr using machine learning, in: 2019 7th International Electrical Engineering Congress (IEECON), IEEE, pp. 1–4.

[30] S. Naz, N. H. Khan, S. Zahoor, M. I. Razzak, Deep ocr for arabic script-based language like pastho, Expert systems 37 (5) (2020) e12565.

[31] M. S. Amin, S. M. Yasir, H. Ahn, Recognition of pashto handwritten characters based on deep learning, Sensors 20 (20) (2020) 5884.

[32] S. Khan, S. Nazir, Deep learning based pashto characters recognition: Lstm-based handwritten pashto characters recognition system, Proceedings of the Pakistan Academy of Sciences: A. Physical and Computational Sciences 58 (3) (2021) 49–58.

[33] M. Z. Rehman, N. Mohd. Nawi, M. Arshad, A. Khan, Recognition of cursive pashto optical digits and characters with trio deep learning neural network models, Electronics 10 (20) (2021) 2508.

[34] I. Uddin, D. A. Ramli, A. Khan, J. I. Bangash, N. Fayyaz, A. Khan, M. Kundi, Benchmark pashto handwritten character dataset and pashto object character recognition (ocr) using deep neural network with rule activation function, Complexity 2021 (1) (2021) 6669672.

[35] S. Khan, H. U. Khan, S. Nazir, Offline pashto characters dataset for ocr systems, Security and Communication Networks 2021 (1) (2021) 3543816.

[36] I. Hussain, R. Ahmad, S. Muhammad, K. Ullah, H. Shah, A. Namoun, Phti: Pashto handwritten text imagebase for deep learning applications, IEEE Access 10 (2022) 113149–113157.

[37] S. Guan, D. Greene, Advancing post-ocr correction: A comparative study of synthetic data, arXiv preprint arXiv:2408.02253 (2024).

[38] H. Kashid, P. Bhattacharyya, Roundtripocr: A data generation technique for enhancing post-ocr error correction in low-resource devanagari languages, arXiv preprint arXiv:2412.15248 (2024).

[39] A. Gupta, A. Vedaldi, A. Zisserman, Synthetic data for text localisation in natural images, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2315–2324.

[40] M. Jaderberg, K. Simonyan, A. Vedaldi, A. Zisserman, Synthetic data and artificial neural networks for natural scene text recognition, arXiv preprint arXiv:1406.2227 (2014).

[41] A. Singh, G. Pang, M. Toh, J. Huang, W. Galuba, T. Hassner, Textocr: Towards large-scale end-to-end reasoning for arbitrary-shaped scene text, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 8802–8812.

[42] O. Sidorov, R. Hu, M. Rohrbach, A. Singh, Textcaps: a dataset for image captioning with reading comprehension, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16, Springer, pp. 742–758.

[43] A. F. Biten, R. Tito, A. Mafla, L. Gomez, M. Rusinol, E. Valveny, C. Jawahar, D. Karatzas, Scene text visual question answering, in: Proceedings of the IEEE/CVF international conference on computer vision, pp. 4291–4301.

[44] A. Mishra, S. Shekhar, A. K. Singh, A. Chakraborty, Ocr-vqa: Visual question answering by reading text in images, in: 2019 international conference on document analysis and recognition (ICDAR), IEEE, pp. 947–952.

[45] A. Singh, V. Natarajan, M. Shah, Y. Jiang, X. Chen, D. Batra, D. Parikh, M. Rohrbach, Towards vqa models that can read, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 8317–8326.

[46] M. Mathew, D. Karatzas, C. Jawahar, Docvqa: A dataset for vqa on document images, in: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp. 2200–2209.

[47] M. Mathew, V. Bagal, R. Tito, D. Karatzas, E. Valveny, C. Jawahar, Infographicvqa, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1697–1706.

[48] A. Masry, D. X. Long, J. Q. Tan, S. Joty, E. Hoque, Chartqa: A benchmark for question answering about charts with visual and logical reasoning, arXiv preprint arXiv:2203.10244 (2022).

[49] J. Tang, Q. Liu, Y. Ye, J. Lu, S. Wei, C. Lin, W. Li, M. F. F. B. Mahmood, H. Feng, Z. Zhao, Mtvqa: Benchmarking multilingual text-centric visual question answering, arXiv preprint arXiv:2405.11985 (2024).

[50] Y. Fan, J. Gu, K. Zhou, Q. Yan, S. Jiang, C.-C. Kuo, X. Guan, X. E. Wang, Muffin or chihuahua? challenging multimodal large language models with multipanel vqa, arXiv preprint arXiv:2401.15847 (2024).

[51] X. Wang, Y. Liu, C. Shen, C. C. Ng, C. Luo, L. Jin, C. S. Chan, A. v. d. Hengel, L. Wang, On the general value of evidence, and bilingual scene-text visual question answering, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10126–10135.

[52] Y. Zhang, R. Zhang, J. Gu, Y. Zhou, N. Lipka, D. Yang, T. Sun, Llavar: Enhanced visual instruction tuning for text-rich image understanding, arXiv preprint arXiv:2306.17107 (2023).

[53] J. Chen, R. Zhang, Y. Zhou, R. Rossi, J. Gu, C. Chen, Mmr: Evaluating reading ability of large multimodal models, arXiv preprint arXiv:2408.14594 (2024).

[54] D. Gao, K. Li, R. Wang, S. Shan, X. Chen, Multi-modal graph neural network for joint reasoning on vision and scene text, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 12746–12756.

[55] A. K. Singh, A. Mishra, S. Shekhar, A. Chakraborty, From strings to things: Knowledge-enabled vqa model that can read and reason, in: Proceedings of the IEEE/CVF international conference on computer vision, pp. 4602–4612.

[56] L. Fu, B. Yang, Z. Kuang, J. Song, Y. Li, L. Zhu, Q. Luo, X. Wang, H. Lu, M. Huang, Z. Li, G. Tang, B. Shan, C. Lin, Q. Liu, B. Wu, H. Feng, H. Liu, C. Huang, J. Tang, W. Chen, L. Jin, Y. Liu, X. Bai, Ocrbench v2: An improved benchmark for evaluating large multimodal models on visual text localization and reasoning, ArXiv abs/2501.00321 (2024).

[57] A. Heakl, A. Sohail, M. Ranjan, R. Hossam, G. Ahmed, M. El-Geish, O. Maher, Z. Shen, F. Khan, S. Khan, Kitab-bench: A comprehensive multi-domain benchmark for arabic ocr and document understanding, arXiv preprint arXiv:2502.14949 (2025).

[58] S. Ghaboura, A. Heakl, O. Thawakar, A. Alharthi, I. Riahi, A. Saif, J. Laaksonen, F. S. Khan, S. Khan, R. M. Anwer, Camel-bench: A comprehensive arabic lmm benchmark, ArXiv abs/2410.18976 (2024).

[59] Z. Wu, T. Song, N. Xie, W. Zhang, M. Zhu, S. Wu, S. Sun, H. Yang, Evaluating menu ocr and translation: A benchmark for aligning human and automated evaluations in large vision-language models, arXiv preprint arXiv:2504.13945 (2025).

[60] Z. Yang, J. Tang, Z. Li, P. Wang, J. Wan, H. Zhong, X. Liu, M. Yang, P. Wang, Y. Liu, Cc-ocr: A comprehensive and challenging ocr benchmark for evaluating large multimodal models in literacy, arXiv preprint arXiv:2412.02210 (2024).

[61] C. Liu, H. Wei, J. Chen, L. Kong, Z. Ge, Z. Zhu, L. Zhao, J. Sun, C. Han, X. Zhang, Focus anywhere for fine-grained multi-page document understanding, ArXiv abs/2405.14295 (2024).

[62] I. Haq, W. Qiu, J. Guo, P. Tang, Pashto offensive language detection: a benchmark dataset and monolingual pashto bert, PeerJ Computer Science 9 (2023) e1617.

[63] I. Haq, Y. Zhang, I. A. Qadri, Pos tagging of low-resource pashto language: annotated corpus and bert-based model, Language Resources and Evaluation (2025). `doi:10.1007/s10579-025-09834-3`.
URL `https://doi.org/10.1007/s10579-025-09834-3`

[64] K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, Bleu: a method for automatic evaluation of machine translation, in: Proceedings of the 40th annual meeting of the Association for Computational Linguistics, pp. 311–318.