AnciDev: A Dataset for High-Accuracy Handwritten Text Recognition of Ancient Devanagari Manuscripts

Vriti Sharma[1,2], **Rajat Verma**[1,2], **Rohit Saluja**[1,2]
[1]Indian Institute of Technology, Mandi, India, [2]BharatGen, India
Correspondence: t24156@students.iitmandi.ac.in

Abstract

The digital preservation and accessibility of historical documents require accurate and scalable Handwritten Text Recognition (HTR). However, progress in this field is significantly hampered for low-resource scripts, such as ancient forms of the scripts used in historical manuscripts, due to the scarcity of high-quality, transcribed training data. We address this critical gap by introducing the **AnciDev** Dataset, a novel, publicly available resource comprising 3,000 transcribed text lines sourced from 500 pages of different ancient Devanagari manuscripts. To validate the utility of this new resource, we systematically evaluate and fine-tune several HTR models on the **AnciDev** Dataset. Our experiments demonstrate a significant performance uplift across all fine-tuned models, with the best-performing architecture achieving a substantial reduction in Character Error Rate (CER), confirming the dataset's efficacy in addressing the unique complexities of ancient handwriting. This work not only provides a crucial, well-curated dataset to the research community but also sets a new, reproducible state-of-the-art for the HTR of historical Devanagari, advancing the effort to digitally preserve India's documentary heritage. Code, Dataset and models are available at https://github.com/vriti2003/AnciDev.

1 Introduction

India possesses one of the world's largest and most significant textual heritages, recorded across millions of ancient manuscripts. These documents, written in various languages and scripts, including historical forms of Devanagari, Gurmukhi, Tamil, Telugu, etc, contain vast, untapped knowledge of history, science, philosophy, rituals, dance forms and local traditions (PRADEEP et al., 2024). Critically, these manuscripts are subject to relentless environmental degradation, damage from pests, and natural ageing, placing this invaluable historical record under immediate threat of extinction. (Zhang et al., 2025) The reason to digitize and preserve this heritage is thus not merely academic, but a fundamental act of preserving Indian culture.

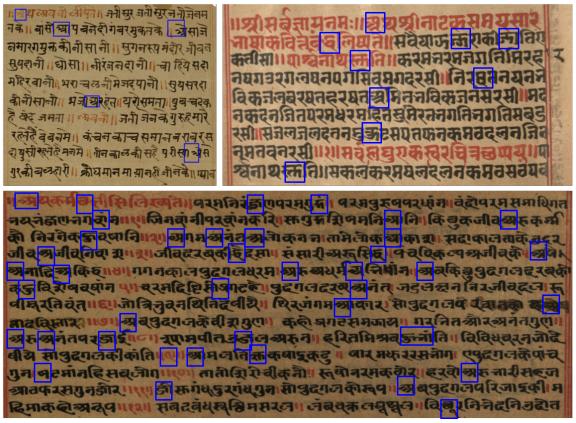


Figure 1: An increasing number of bounding boxes (indecipherable characters) depicts a problem in reading manuscripts from easy (top left), medium (top right), and difficult (bottom).

Digitization is the first step, but accurate preservation and accessibility require that these images be convertible into searchable, machine-readable text. This process is accomplished through Handwritten Text Recognition (HTR). While modern HTR systems have achieved high accuracy for Latin scripts and printed Devanagari, they fail drastically when applied to historical and ancient manuscripts. The complexity is rooted in non-uniform handwriting styles, stylistic variations in ancient scripts, heavy noise, variable paper quality, ink bleed-through, and physical deterioration.

The primary obstacle preventing the accurate HTR of ancient Indian manuscripts is the profound lack of high-quality, expertly annotated, and publicly available training data. Existing research often relies on small, proprietary, or particular datasets that do not generalize. This severely limits the de-

Proceedings of the 1st Workshop on Benchmarks, Harmonization, Annotation, and Standardization for Human-Centric AI in Indian Languages (BHASHA 2025), pages 91–101

velopment of robust, high-performance machine learning and deep learning models necessary for large-scale archival conversion.

This paper describes the dataset created to address this data deficit and the baselines that have been considered for the information extraction task. The main contributions of our work are the following:
(i) To the best of our knowledge, we introduce the **AnciDev** Dataset, the first publicly available, open-source dataset, comprising 3,000 transcribed lines extracted from 500 pages of ancient manuscripts in the Devanagari script.
(ii) We leverage this novel resource to establish reproducible HTR benchmarks by fine-tuning several recognized architectures, including Tesseract (Smith, 2007), a specialised CNN-RNN, and the attention-LSTM models.

Our results demonstrate that the **AnciDev** Dataset enables a significant leap in HTR performance, thus providing both a critical tool and a new state-of-the-art for the digital preservation of Indian textual legacy.

2 Relative Work

2.1 Progress and Challenges in Handwritten Text Recognition (HTR)

Handwritten Text Recognition (HTR) has been a significant research area, witnessing substantial breakthroughs, particularly with the advent of deep learning architectures. Early work focused on statistical models and Hidden Markov Models (HMMs) (Anigbogu and Belaid, 1995), but modern approaches predominantly utilize Convolutional Neural Networks (CNNs) for feature extraction coupled with Recurrent Neural Networks (RNNs) or Attention mechanisms for sequence decoding (Dwivedi et al., 2020). For widely used Latin scripts, such as English and German, HTR systems have achieved near-human performance on standardized datasets like IAM and READ (Marti and Bunke, 2002; Peiró et al., 2017).

The challenge intensifies when transitioning from modern cursive scripts to ancient manuscripts. Issues such as degraded document quality, unusual character variations (allography), and heavy noise necessitate specialized approaches (Guan et al., 2025; Souibgui and Kessentini, 2020). Commercial and open-source solutions are widely deployed, function primarily as Optical Character Recognition (OCR) tools, and often serve as a necessary, yet insufficient, baseline for the complex task of HTR, especially on historical data (Fleischhacker et al., 2024; Kim et al., 2025).

2.2 HTR for Indian Scripts and Devanagari

Research efforts dedicated to Indian scripts, including Devanagari, have been gaining momentum. Initial work focused on printed Devanagari text recognition, achieving high accuracy (Chaudhuri, 2009; Bag and Harit, 2013; Sharma and Mudgal, 2018). However, the transition to handwritten and historical manuscripts remains a major hurdle. The complexity of the Devanagari script, with its inherent vowel modifiers, combined with the structural irregularities of ancient handwriting, creates unique HTR problems (Roy et al., 2017).

Several studies have explored HTR for Indic languages, utilizing various contemporary models. For instance, some researchers have employed specialized CNN-RNN architectures combined with Connectionist Temporal Classification (CTC) loss for contemporary Hindi and Marathi handwriting (Bisht and Gupta, 2022). More recent developments have seen the application of Transformer and attention-based encoder-decoder models, similar to the SanskritOCR attention-LSTM model (Dwivedi et al., 2020), demonstrating improved handling of long-range dependencies in complex scripts like Sanskrit and other Indic languages. Despite these architectural advances, the reported success is often confined to modern, relatively clean datasets or proprietary archives.

2.3 The Manuscript Data Scarcity Problem

The most critical barrier to developing robust HTR for historical Hindi and related Devanagari manuscripts is the lack of publicly available, large-scale, annotated datasets. While initiatives exist for digital archiving of manuscripts across various institutions (National Mission for Manuscripts, 2025), the resulting image data is rarely released with expert line-level transcriptions necessary for supervised machine learning training. This contrasts sharply with resource-rich European historical HTR, which benefits from extensive open datasets like those from the DIVA series (Simistira et al., 2016). Previous works that have fine-tuned models for Devanagari HTR have either utilized datasets too small for generalization or relied on synthetic data, which fails to capture the intricate, real-world noise present in aged paper, ink bleed,

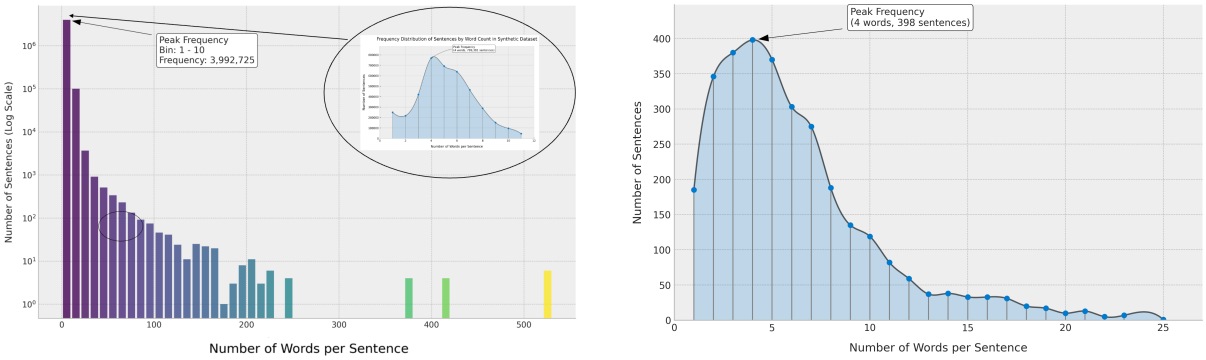


Figure 2: Data distribution of the self-curated **Synthetic** and **AnciDev** Datasets. Depicting similarity of datasets to assist models for pre-training followed by fine-tuning on **AnciDev** dataset.

and the variations in ancient scribe hands (Kasuba et al., 2025).

3 Dataset

We introduce the **AnciDev** Dataset, a novel corpus for ancient Devanagari manuscript recognition on Hindi and Sanskrit languages, comprising 3,000 text lines extracted from 500 historical manuscript pages. This dataset addresses a critical gap in optical character recognition (OCR) research for Indic scripts, particularly for historical documents where conventional models trained on modern printed text exhibit poor performance due to differences in writing styles and conventions.

3.1 Data Collection and Composition

The manuscript pages were sourced from National Manuscript Mission archives of historical texts spanning the 16th to 19th centuries. These documents represent diverse genres including religious texts, literary works, and administrative records, providing substantial variation in writing styles and vocabulary. Each page was carefully selected to ensure representation of different scribal hands and dialectical variations in Hindi orthography from different historical periods.

The **AnciDev** Dataset consists of **500 manuscript page images** with corresponding ground truth transcriptions. From these pages, we extracted **3,000 individual text lines** using semi-automated segmentation followed by manual verification and correction. Lines were extracted as complete, meaning while maintaining sufficient surrounding context to facilitate text to help with accurate character recognition.

As the **AnciDev** dataset was annotated by a single annotator, reliability was ensured through a quality control procedure in which a randomly selected subset of 25% of the images was independently reviewed, and any inconsistencies were corrected.

3.2 Data Characteristics and Challenges

The **AnciDev** Dataset presents several characteristics that distinguish it from other historical OCR corpora. The primary challenge stems from the significant stylistic differences between historical and modern Devanagari writing. Historical manuscripts exhibit distinctive paleographic features: character forms that differ substantially from contemporary standards, unique ways of forming conjuncts, and writing conventions that are no longer in common use. The handwritten nature introduces high intra-class variability in character morphology, with considerable differences in stroke patterns, character proportions, and spacing conventions across different scribes and time periods.

The writing style variations across different historical periods are particularly notable in the **AnciDev** Dataset. Manuscripts from the 16th century exhibit markedly different character formations compared to those from the 18th or 19th centuries. Each scribe developed an individual hand, resulting in diverse representations of the same characters across the corpus. The cursive nature of historical handwriting, combined with period-specific calligraphic conventions, creates substantial challenges for recognition systems trained on modern, standardized Devanagari.

4 Experiments

In this section, we describe the experimental setup, model architectures, training procedures, and evaluation methodology used to establish baseline performance on the **AnciDev** Dataset.

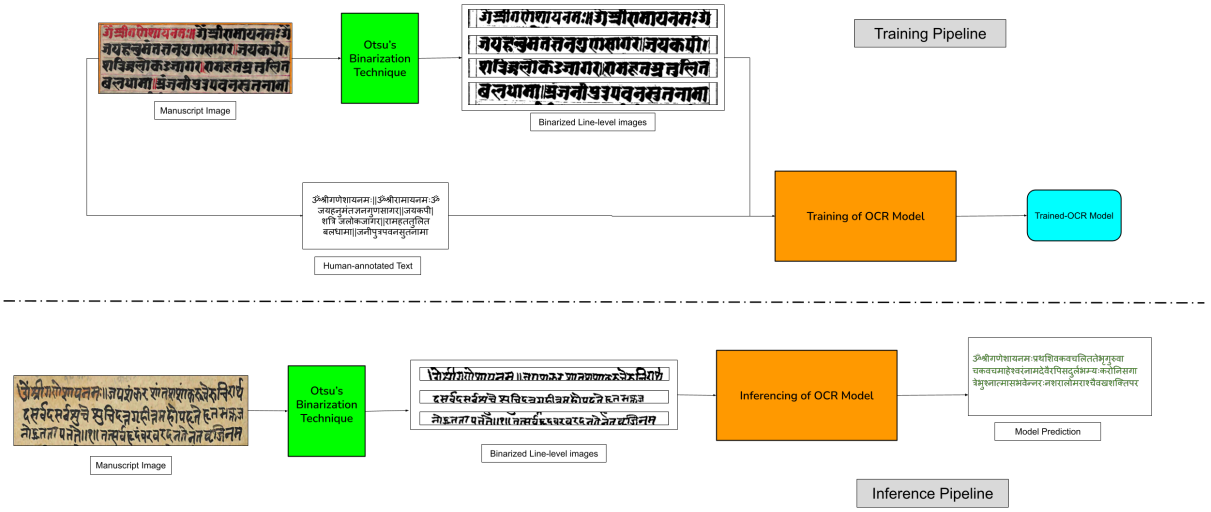


Figure 3: (Up)Line-level segmentation followed by the training pipeline of the training of the OCR model. (Down)Inference pipeline of the OCR model.

Model	CER(%)(↓)	WER(%)(↓)
CNN-RNN	48.59	98.20
Attention-LSTM	46.33	96.73
Tesseract-5	**30.06**	**87.42**

Table 1: Comparison of OCR models' performance keeping the same **AnciDev** test set.

Exp.	LR	Target Error	Arch.	CER(%)(↓)	WER(%)(↓)
1	0.00001	0.01	medium	**33.01**	**88.95**
2	0.0001	0.01	medium	33.38	89.06
3	0.001	0.01	medium	34.88	89.46
Best Learning Rate: 0.00001					
8	0.00001	0.01	small	35.27	90.45
9	0.00001	0.01	medium	33.01	88.95
10	0.00001	0.01	large	**32.27**	**88.95**
Best Architecture: large					
11	0.00001	0.005	large	***30.06***	***87.42***
12	0.00001	0.01	large	32.27	88.95
13	0.00001	0.02	large	33.01	90.09
Best Target Error: 0.005					

Table 2: Hyperparameter tuning for **Tesseract-5**. **LR**: learning rate; **Arch.**: architecture variant (small/medium/large; uninitialized layers randomly initialized for **medium** and **large**. All experiments ran for 10k iterations. **bold** indicates the best model in each category and *italics* indicates the overall best model.

4.1 Experimental Setup

We evaluate three OCR models on the **AnciDev** Dataset: Attention-OCR, CNN-RNN[1], and Tesseract-5[2]. To leverage transfer learning and improve recognition performance on historical manuscripts, we employ a two-stage training strategy: (1) pre-training on large-scale synthetic Devanagari data and (2) fine-tuning on a combina-

[1]GitHub link for CNN-RNN and Attention-LSTM: https://github.com/ihdia/sanskrit-ocr

[2]GitHub link for latest-release Tesseract: https://github.com/tesseract-ocr/tesseract

tion of real manuscript images from the **AnciDev** Dataset and additional synthetic data.

4.2 Pre-training Phase

To initialise our models with knowledge of the Devanagari script, we pre-trained the Attention-OCR and CNN-RNN models on a large synthetic dataset of document images. This synthetic dataset was generated using 820 different Devanagari fonts applied to 5,000 text images, resulting in a total of 4.1M samples. The synthetic data was split into training, validation, and test sets with a ratio of 7:2:1, yielding 2.8M training samples, 0.8M validation samples, and 0.41M test samples. Pre-training was conducted using a batch size of 32. This pre-training phase allows the models to learn general Devanagari character shapes, common conjuncts, and basic script features from document images before exposure to the more challenging historical manuscript data.

4.3 Finetuning Phase

After pre-training, we fine-tuned all three models on the **AnciDev** Dataset. The fine-tuning dataset consists of 2,458 real manuscript line images for training and 627 images for validation, maintaining an 80:20 split ratio. To augment the training data and improve model generalization, we supplemented the real manuscript images with synthetically generated samples that mimic historical writing styles. Table 2 shows the hyperparameter tuning of the best model **Tesseract-5**[1] based on learning rate, and model architecture.

These experiments were designed to assess the impact of synthetic data augmentation on model performance and to determine the optimal balance between real and synthetic training samples for historical manuscript recognition.

5 Results

In this section, we present the quantitative and qualitative results obtained from our experiments on the AnciDev Dataset. We analyze the performance of three OCR models—CNN-RNN, Attention-OCR, and Tesseract-5—across different training configurations and discuss the factors contributing to their recognition accuracy. Table 1 summarizes the Character Error Rate (CER) and Word Error Rate (WER) achieved by each model on the test set. The results clearly demonstrate that Tesseract-5 significantly outperforms both CNN-RNN and Attention-OCR across all metrics. Tesseract-5 achieves a CER of 30.06% and WER of 87.42%, representing substantial improvements of approximately 16-18 percentage points in CER and 9-11 percentage points in WER compared to other models. To optimize the performance of Tesseract-5, we conducted systematic hyperparameter tuning experiments. Table 2 presents the results of these experiments, where we evaluated different configurations of learning rate, model architecture, and target error threshold. The hyperparameter optimization process revealed that:

- A small learning rate of 0.00001 provides the best convergence for historical manuscript fine-tuning.
- The large architecture variant offers superior capacity for learning complex historical character patterns.
- A target error threshold of 0.005 enables more refined training convergence.

The superior performance of Tesseract-5 demonstrates that exposure to real handwritten samples during pre-training is crucial for adapting to historical writing styles. Historical Devanagari manuscripts exhibit characteristics such as cursive connections, varying stroke pressure and non-uniform spacings that are better learned from authentic handwritten data rather than synthetic font-based samples.

The qualitative results comparing the three models are presented in Figure 4, where character-level errors are highlighted in red. We present four representative samples: the first three samples represent relatively high manuscript quality, and Tesseract-5 consistently achieves the highest accuracy with minimal character-level errors. In contrast, the fourth sample represents a challenging case with irregular spacing and complex cursive connections, where all models struggle significantly. Tesseract-5, although still producing errors, maintains the best performance, with a recognisable text structure and limited error propagation.

The qualitative analysis reveals that Tesseract-5's pre-training on real handwritten data provides superior generalization, enabling it to maintain reasonable accuracy even on challenging manuscripts. In contrast, CNN-RNN and Attention-OCR, which

[1]The IITB OCR team trained Tesseract-5 on 7,000 synthetic lines created using real verse text and 3,000 printed text lines.

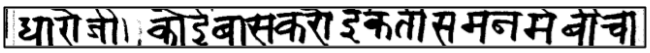
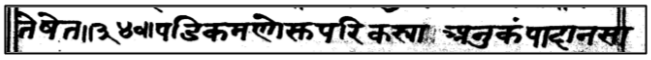
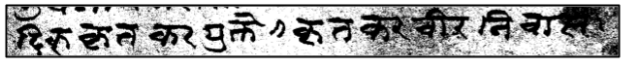
Sample Image	
Ground Truth	करई॥इसहिमंत्रपठरोगनिवारै छाया।
CNN-RNN	करई रसहिमं परोगतिवारै छाया ।
Attention-OCR	कर ॥इसहिमंतपटरोगनिवारै छाय।
Tesseract-5	करई॥इरहिमंत्रपतरोगनिवारै छाय।
Sample Image	
Ground Truth	धारोजी॥कोई बासकरो इकतीसमनमेबीचा
CNN-RNN	धारोजी कोई बासकरीई कतीस मनम बीचा
Attention-OCR	ध्योजी॥कोई बासकरो कतीसमनमेबीचा
Tesseract-5	धारोजी॥कोई बासकरो इकतीसमनमबीचा
Sample Image	
Ground Truth	तेषेत ॥३४०॥ पडिकमणेसपरिकरया अनुकं पादानसा
CNN-RNN	तेषेत॥३४० पडिकमणोसरपरिकलसा॥नुकं ानस
Attention-OCR	तेषेत ॥३४०॥ पडिकमणेसपरिकस्या अनुकं पदानसा
Tesseract-5	तेषेत ॥३४०॥ परिकमणेसपरिकरया अनुकं पादानसा
Sample Image	
Ground Truth	दिक् कृत कर युक्ते ॥ कृत करवीर निवासे ।
CNN-RNN	्रिसतक्ागुवे कल मजिनिदिा
Attention-OCR	दिरु कत कर युत्ते कत करबीर निवाला
Tesseract-5	दिक कृत कर पुत्ते ॥ कृत करकीर निबा

Figure 4: Qualitative examples comparing model predictions on the **AnciDev** Dataset. Samples 1-3 show successful recognition cases, while Sample 4 represents a challenging failure case. Red highlights indicate character-level errors.

are primarily trained on synthetic data, exhibit significant performance degradation on challenging samples, with CNN-RNN failing catastrophically. The observed error rates, while appearing high compared to modern printed text recognition (typically <5% CER), are consistent with the complexity of ancient handwritten manuscripts. Historical Devanagari HTR faces unique challenges like (i)

Paleographic variations across 16th-19th century manuscripts showing markedly different character formations, (ii) Physical degradation, including ink bleeding, paper deterioration, and fading, (iii) Inconsistent spacing and cursive connections between characters, and (iv) Scribal variations with each scribe developing individual writing styles. These results strongly reinforce the quantitative findings and confirm the critical importance of handwriting-aware pre-training for the recognition of historical manuscripts.

We have experimented with transformer-based models like trocr-large-handwritten (Li et al., 2023) and OCR-Donut-CORD (Kim et al., 2022). The results were very discouraging, which is most likely due to the smaller amount of data in our proposed **AnciDev** dataset. Refer to Appendix F for more details.

6 Conclusion and Future Work

The **AnciDev** dataset addresses a significant gap in OCR research for historical devanagari script by capturing the distinctive writing style variations of ancient Devanagari manuscripts that differ substantially from modern standardized script. We established baseline performance by evaluating two OCR models, (i) Attention-OCR, (ii) CNN-RNN, using a two-stage training approach combining pre-training on large-scale synthetic Devanagari data with fine-tuning on **AnciDev** dataset, and (iii) fine-tuning of Tesseract-5 on **AnciDev** dataset. Our experiments provide valuable insights into optimal training strategies for historical document recognition. Among the evaluated models, Tesseract-5 demonstrated superior performance, highlighting the effectiveness of LSTM-based architectures for handling the unique challenges posed by historical writing styles.

Future work will focus on expanding the **AnciDev** Dataset to include a larger variety of manuscript types, additional time periods, and diverse scribal hands to improve model generalization. We aim to investigate more advanced transformer-based architectures and develop specialized data augmentation techniques that better simulate historical writing variations. Additionally, developing language model-based post-processing systems that leverage Sanskrit and Hindi linguistic constraints could further reduce error propagation, while conducting multi-annotator studies would help quantify annotation reliability through inter-annotator agreement analysis with multiple expert transcribers.

Acknowledgments

We gratefully acknowledge the BharatGen team and IIT Bombay OCR team (https://www.cse.iitb.ac.in/~ocr/) for generously sharing their trained models (Tesseract-5) on printed Sanskrit data, which significantly contributed to establishing baseline performance on our manuscript dataset. We extend our sincere thanks to the Department of Science and Technology (DST), Government of India, and the Technology Innovation Hub at IIT Bombay (TiH- IITB) for their financial support and institutional backing that made this research possible.

References

Julian C. Anigbogu and Abdel Belaid. 1995. Hidden markov models in text recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 9(6):925–958.

S. Bag and Gaurav Harit. 2013. A survey on optical character recognition for bangla and devanagari scripts. *Sādhanā*, 38(1):133–168. Survey of OCR work, including printed Devanagari text recognition and its high accuracy in many works.

Mamta Bisht and Richa Gupta. 2022. Offline handwritten devanagari word recognition using cnn-rnn-ctc. In *SN Computer Science, Volume 4, Issue 1*, page Article 88, Singapore. Springer Nature Singapore.

B. B. Chaudhuri. 2009. On ocr of major indian scripts: Bangla and devanagari. In *Guide to OCR for Indic Scripts*. Springer. Describes multi-font printed Devanagari OCR; includes segmentation, symbol recognisers and reports encouraging results.

Agam Dwivedi, Rohit Saluja, and Ravi Kiran Sarvadevabhatla. 2020. An ocr for classical indic documents containing arbitrarily long words. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*.

David Fleischhacker, Wolfgang Goederle, and Roman Kern. 2024. Improving ocr quality in 19th century historical documents using a combined machine learning based approach. *arXiv preprint arXiv:2401.07787*. Shows how conventional OCR (e.g. Tesseract) still has high error rates on historical sources and is often used together with structure detection and ML for improvements.

Shuhao Guan, Moule Lin, Cheng Xu, Xinyi Liu, Jinman Zhao, Jiexin Fan, Qi Xu, and Derek Greene. 2025. Prep-ocr: A complete pipeline for document image

restoration and enhanced ocr accuracy. In *arXiv preprint arXiv:2505.20429*. Two-stage pipeline combining image restoration and post-OCR correction to address heavy noise and distortions in historical documents.

Badri Vishal Kasuba, Akhilesh Kumar, Manish Singh, Umapada Pal, and C. V. Jawahar. 2025. Platter: A page-level handwritten text recognition system for indic scripts. *arXiv preprint arXiv:2502.06172*. Proposes a page-level HTR model for Indic scripts; discusses data scarcity and limitations of synthetic augmentation for capturing real-world document degradations.

Geewook Kim, Teakgyu Choi, Junyeop Lee, and 1 others. 2022. Ocr-free document understanding transformer. In *European Conference on Computer Vision*. Springer Nature Switzerland. Proposes an end-to-end transformer model (DONUT) that performs document understanding without relying on traditional OCR pipelines.

Seorin Kim, Julien Baudru, Wouter Ryckbosch, Hugues Bersini, and Vincent Ginis. 2025. Early evidence of how llms outperform traditional systems on ocr/htr tasks for historical records. *arXiv preprint arXiv:2501.11623*. Compares commercial/open OCR standard HTR systems (e.g. EasyOCR, Tesseract, TrOCR) with newer methods; shows OCR/HTR as baseline for historical transcription tasks.

Minghao Li, Tengchao Lv, Lei Cui, Yijuan Lu, Jiachen Gu, Dongdong Zhang, Yutong Lu, and Furu Wei. 2023. Trocr: Transformer-based optical character recognition with pre-trained models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37. Introduces a transformer-based OCR model leveraging large-scale pre-training to achieve state-of-the-art recognition performance.

U.-V. Marti and H. Bunke. 2002. The iam handwriting database: An english sentence database for offline handwriting recognition. offline handwriting recognition database. Contains 1,539 handwritten pages by 657 writers; includes 115,320 isolated and labeled words.

Government of India National Mission for Manuscripts. 2025. Digitization of manuscripts: Mission initiatives and institutional collaboration. *National Mission for Manuscripts Website*.

Joan Andreu Sánchez Peiró, Verónica Romero, Alejandro H. Toselli, Mauricio Villegas, and Enrique Vidal. 2017. Dataset for icdar2017 competition on handwritten text recognition on the read dataset. Includes Train-A, Train-B, Test splits of pages from historical archival documents.

NANDHINI PRADEEP, DIVYAR SUBRAMANIAN, and MADHAVAN K GANAPATHY. 2024. Digitizing india's ancient texts: Ai for tamil palm leaf manuscript preservation and accessibility.

Partha Pratim Roy, Ayan Kumar Bhunia, Ayan Das, Prasenjit Dey, and Umapada Pal. 2017. Hmm-based indic handwritten word recognition using zone segmentation. In *arXiv preprint arXiv:1708.00227*.

Richa Sharma and Tarun Mudgal. 2018. Primitive feature-based optical character recognition of the devanagari script. In *Progress in Advanced Computing and Intelligent Engineering: Proceedings of ICACIE 2017, Volume 2*, pages 249–259, Singapore. Springer Singapore.

Fotini Simistira, Mathias Seuret, Nicole Eichenberger, Angelika Garz, Marcus Liwicki, and Rolf Ingold. 2016. Diva-hisdb: A precisely annotated large dataset of challenging medieval manuscripts. In *Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition (ICFHR 2016)*, page 471–476.

Ray Smith. 2007. An overview of the tesseract ocr engine. In *Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Vol. 2*. IEEE.

Mohamed Ali Souibgui and Yousri Kessentini. 2020. De-gan: A conditional generative adversarial network for document enhancement. *arXiv preprint arXiv:2010.08764*. Restores severely degraded document images (e.g. blur, watermark, noise) using GAN; improves performance of OCR on those degraded inputs.

Wenjie Zhang, Shan Wang, Liuyang Han, and Hong Guo. 2025. Aging effects of relative humidity on palm leaf manuscripts and optimal humidity conditions for preservation. *npj Heritage Science*, 13(1):218.

A Anci-Dev Dataset

Table 3 provide the details of manuscripts with their names and number of pages digitzed.

B Tesseract-5 LSTM Network Architectures

Tesseract-5 employs Long Short-Term Memory (LSTM) networks for optical character recognition. This appendix details three standard architectures with varying capacities.

C Network Architecture Notation

Tesseract-5 uses specialized notation for LSTM architectures:

$$\text{Architecture} = [I,\ L_1,\ L_2,\ \ldots,\ L_n,\ O] \quad (1)$$

where:

- I = Input specification: $[c, h, 0, d]$

Name of the Manuscript	Pages
Bhaṭṭī Kāvyā Bhaṭṭī	18
Gauḍīpārśvastavana, Kamalā Ārtī, Madanāṣṭaka	4
Hanumān Cālīsā	5
Śānti Pāṭha	7
Jānakī Prāta Padakam	6
Bārahkharī	36
Rāmāyaṇa Bāla Kāṇḍa	65
Lāvanī Pada Saṅgraha	69
Śiva Stotra, Skanda Purāṇa, Candrakumāra Caupaī	5
Gommaṭasāra	22
Kṛtibodha	25
Kiśansinha Kavi	100
Rakṣā Bandhana Kathā	7
Vicāramālā	36
Candan aṣṭhī Vrata Kathā	18
Mukhavāstrikā Carcā Dohā	7
Vinatī-Saṅgraha	18
Samaya Sāra Nāṭaka	8

Table 3: Details of the **AnciDev** dataset.

  - c = number of channels (1 for grayscale)
  - h = input height (typically 36 pixels)
  - d = depth/dimension

- L_i = Layer specification

- O = Output layer specification

C.1 Layer Type Notation

Notation	Description
$Ct_{k,k,f}$	Conv + Tanh, $k \times k$ kernel, f maps
$Mp_{k,k}$	Max pooling, $k \times k$ window
$Lfys_n$	Forward LSTM, n units, y-dim
Lfx_n	Forward LSTM, n units, x-dim
Lrx_n	Reverse LSTM, n units, x-dim
$O1c$	Output Softmax, $\|\Sigma\|$ classes

Table 4: Layer notation in Tesseract-5 LSTM

D Tesseract-5 Architecture Specifications

We detail three standard architectures with increasing capacity: Small (S), Medium (M), and Large (L).

D.1 Small Architecture

Network String:

```
[1,36,0,1 Ct3,3,16 Mp3,3Lfys48,
Lfx96 Lrx96, Lfx128 O1c]
```

Mathematical Form:

$$\mathcal{A}_{\text{s}} = I \to C_{16} \to P \to H_{48}^{y} \to H_{96}^{\rightarrow} \to H_{96}^{\leftarrow} \to H_{128}^{\rightarrow} \to S \tag{2}$$

Layer	Type	Params	Purpose
L_0	Input	0	Image
L_1	Conv	144	Features
L_2	Pool	0	Reduction
L_3	LSTM-F	13K	Vertical
L_4	LSTM-F	56K	L→R
L_5	LSTM-R	56K	R→L
L_6	LSTM-F	115K	Deep
L_7	Softmax	$128\|\Sigma\|$	Class
Total		**240K**	

Table 5: Small architecture details

D.2 Medium Architecture

Network String:

```
[1,36,0,1 Ct3,3,16 Mp3,3Lfys48,
Lfx96 Lrx96,Lfx256 O1c]
```

Mathematical Form:

$$\mathcal{A}_{\text{m}} = I \to C_{16} \to P \to H_{48}^{y} \to H_{96}^{\rightarrow} \to H_{96}^{\leftarrow} \to H_{256}^{\rightarrow} \to S \tag{3}$$

Layer	Type	Params	Purpose
L_0	Input	0	Image
L_1	Conv	144	Features
L_2	Pool	0	Reduction
L_3	LSTM-F	13K	Vertical
L_4	LSTM-F	56K	L→R
L_5	LSTM-R	56K	R→L
L_6	LSTM-F	**266K**	**Rich**
L_7	Softmax	$256\|\Sigma\|$	Class
Total		**391K**	

Table 6: Medium architecture details

D.3 Large Architecture

Network String:

```
[1,36,0,1 Ct3,3,16 Mp3,3Lfys64 Lfx128,
Lrx128, Lfx256 Lrx256 O1c]
```

Mathematical Form:

$$\mathcal{A}_{\text{l}} = I \to C_{16} \to P \to H_{64}^{y} \to H_{128}^{\rightarrow} \to H_{128}^{\leftarrow} \to H_{256}^{\rightarrow} \to H_{256}^{\leftarrow} \to S \tag{4}$$

Layer	Type	Params	Purpose
L_0	Input	0	Image
L_1	Conv	144	Features
L_2	Pool	0	Reduction
L_3	LSTM-F	**17K**	**Rich V**
L_4	LSTM-F	**100K**	**L→R**
L_5	LSTM-R	**100K**	**R→L**
L_6	LSTM-F	395K	Deep 1
L_7	LSTM-R	**395K**	**Deep 2**
L_8	Softmax	$512\|\Sigma\|$	Class
Total		**1.0M**	

Table 7: Large architecture details

E Mathematical Formulation

E.1 LSTM Cell

For time step t with input x_t, hidden h_{t-1}, cell state c_{t-1}:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{5}$$
$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{6}$$
$$\tilde{c}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \tag{7}$$
$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{8}$$
$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{9}$$
$$h_t = o_t \odot \tanh(c_t) \tag{10}$$

where σ is sigmoid, $\odot$ is element-wise product, and W_*, b_* are learnable parameters.

E.2 Bidirectional LSTM

Bidirectional combines forward and reverse:

$$h_t = [\overrightarrow{h}_t; \overleftarrow{h}_t] \tag{11}$$

where $\overrightarrow{h}_t$ is forward and $\overleftarrow{h}_t$ is backward.

F Additional Experiments

Table 9 compares the performance of two transformer-based OCR models—**trocr-large-handwritten**[3] and **OCR-Donut-CORD**[4]—on the **AnciDev** test set. Both models exhibit extremely poor recognition accuracy, with character error rates (CER) and word error rates (WER) exceeding 99.9%. These findings show that state-of-the-art transformer-based OCR systems trained on ancient or degraded manuscript data, highlighting the need for domain-specific training strategies or specialized architectures for historical document recognition.

Table 8 presents the performance of three OCR models—CNN-RNN, Attention-LSTM, and Tesseract-5—under different training data compositions combining manuscript (*m*) and synthetic (*s*) samples. The results show that Tesseract-5 consistently achieves the lowest word error rate (WER) across all dataset ratios, while CNN-RNN generally performs more reliably than Attention-LSTM, whose error rates increase substantially as the proportion of synthetic data grows. Notably, none of the models show significant improvement when synthetic data is added; in several cases, performance even degrades, particularly for Attention-LSTM. These findings suggest that synthetic data does not effectively substitute for real manuscript samples in historical OCR tasks, and that model robustness depends strongly on the availability of authentic manuscript training data.

[3]Hugging Face Link for microsoft/trocr-large-printed: https://huggingface.co/microsoft/trocr-large-printed

[4]Hugging Face Link for jinhybr/OCR-Donut-CORD: https://huggingface.co/jinhybr/OCR-Donut-CORD

Dataset Ratio (m:s)	Model	CER(%) (↓)	WER(%) (↓)
100::0	CNN-RNN	32.59	96.20
	Attention-LSTM	46.33	98.73
	Tesseract-5	***30.06***	***87.42***
60::40	CNN-RNN	**35.33**	96.24
	Attention-LSTM	67.50	98.10
	Tesseract-5	42.18	**94.17**
50::50	CNN-RNN	**32.75**	96.16
	Attention-LSTM	68.44	98.57
	Tesseract-5	43.19	**94.90**
40::60	CNN-RNN	**33.82**	95.86
	Attention-LSTM	71.57	99.75
	Tesseract-5	43.49	**95.02**

Table 8: Comparison of OCR models' performance across different manuscript-to-synthetic dataset ratios, keeping the same **AnciDev** test set. *m* and *s* represents the manuscript and synthetic dataset.**bold** indicates the best model in each category and *italics* indicates the overall best model.

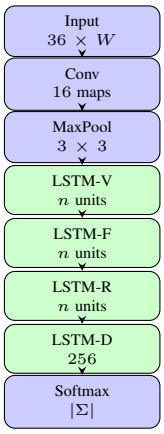


Figure 5: Generic LSTM pipeline

Model	CER(%) (↓)	WER(%) (↓)
trocr-large-handwritten	99.9	99.9
OCR-Donut-CORD	99.9	99.9

Table 9: Comparison of transformer-based models on the **AnciDev** test set.

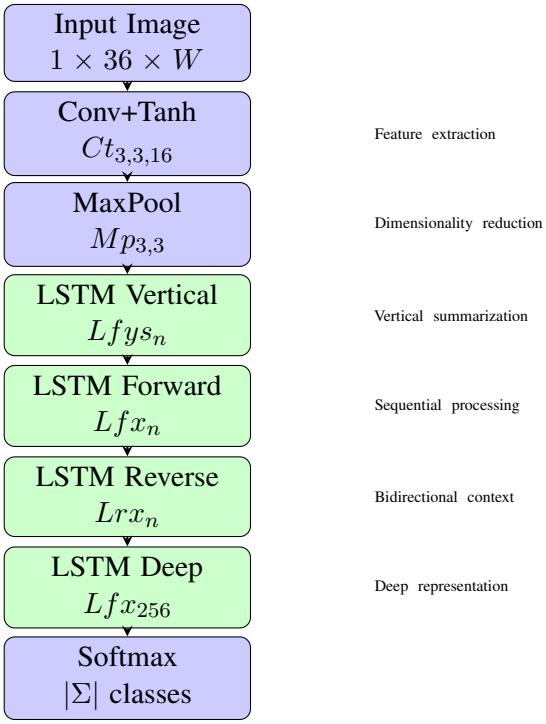


Figure 6: LSTM architecture processing pipeline