IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125244

Kannada Character Recognition From Ancient Epigraphical Inscription Using OCR

Dr. Sharath Kumar Y H [1], Priyanka K S [2], Tanya M [3], Deepika H [4], Harish Gowda M S [5]

HOD & Professor, Department of ISE, MITM, Mysore, VTU Belagavi, India[1.]

UG Students, Department of ISE, MITM, Mysore, VTU Belagavi, India[2-5]

Abstract: The digitization and recognition of regional and ancient Indian languages have gained significant attention in recent years, driven by the need to preserve linguistic heritage and improve accessibility. This literature survey consolidates recent research focusing on handwriting recognition, translation, and transliteration techniques applied to scripts such as Kannada, Halegannada, Tulu, and Brahmi. Jayanna et al. (2024) explored deep learning models—CNN, RNN-LSTM, and Vision Transformers—for Kannada handwriting recognition, achieving a maximum accuracy of 99.7% with Vision Transformers, although with higher computational costs. Harsha A C et al. (2024) tackled the low-resource Halegannada-to-Hosagannada translation using a hybrid LSTM and dictionary-based approach, achieving 77.92% accuracy while facing challenges in word sense disambiguation. Prathwini et al. (2024) addressed the recognition and translation of the Tulu language using CNNs and encoder-decoder models, reaching up to 92% recognition accuracy and BLEU scores of 0.83 for translation, but struggling with dialect variations and limited datasets. Mubarakkaa et al. (2024) developed an OCR system for Brahmi-to-Tamil transliteration using CNNs and Tesseract OCR, showing promise but limited by degraded character forms and restricted script coverage. Lastly, Bhumika Purant and Mallamma Reddy (2024) proposed a VGG19-based model for converting Kannada inscriptions into modern Hosagannada via OCR, deployed as a web application with 80–90% accuracy, though constrained by data scarcity and non-standard inscription structures.

Keywords: Epigraphical Inscription Recognition

I. INTRODUCTION

Kannada, one of the oldest Dravidian languages, has a rich literary heritage that dates back over 1,500 years. Classical Kannada literature includes poetry, prose, inscriptions, manuscripts,and philosophical texts from different historical periods such as the Halekannada (Old Kannada), Nadukannada (Middle Kannada), and Hosakannada (Modern Kannada) eras. Manyof these ancient texts are not accessible to modern readers due to linguistic differences, obsolete vocabulary, and script variations.

Epigraphical inscriptions in Kannada, dating back to the 5th century**,** hold immense historical and linguistic significance. These inscriptions, often found on temple walls, copper plates, and stone carvings, provide crucial insights into ancient Karnataka's culture, administration, and literature. However, due to natural degradation, variations in writing styles, and the evolution of Kannada script (Halekannada, Nadukannada**,** and Hoskannada), reading and interpreting these inscriptions is challenging.With recent advancements in Natural Language Processing (NLP), Optical Character Recognition (OCR), and Neural Machine Translation (NMT), we can develop an AI-powered system to identify, extract, and translate old Kannada literature into modern Kannada andother languages while preserving contextual accuracy.

This project aims to develop an Optical Character Recognition (OCR) system that can recognize, digitize, and translate Kannada characters from ancient epigraphical inscriptions. The proposed system will use machine learning techniques to enhance text extraction accuracy, thus helping researchers, historians, and Kannada language enthusiasts access and analyze these historical texts more effectively.

II. PROPOSED SYSTEM

Period identification of ancient scripts is important, which enables to know the character bank to be employed for automatic reading of age-old documents. An ancient Kannada epigraph is input to the system and output is the predicted era of the script.

The generic Period Identification model is shown in Figure 1.1 and involves the following components:

- Preprocessing: This sub-system pre-processes and enhances the input document image before it is passed to further stages.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125244

- Segmentation: Sampled characters are extracted from the document images which are normalized to a particular size. The characters thus obtained are fed into the Feature Extraction phase.
- Feature Extraction: The distinguishing features are extracted from the sampled characters and saved in a file.
- Database: The database here represents the file used to save the extracted features. The feature vectors are used for training the classifier and later during testing for the prediction of the era.
- Classifier: The Classifier is trained using the features stored in the file. It is also possible to save the trained Classifier for later use. The trained Classifier is used further to predict the era.The proposed work uses two models for predicting era of ancient Kannada epigraphs. The first model is SVM Classifier with Zone-based features and the other is Random Forest Classifier (RF) with Central and Zernike moments.

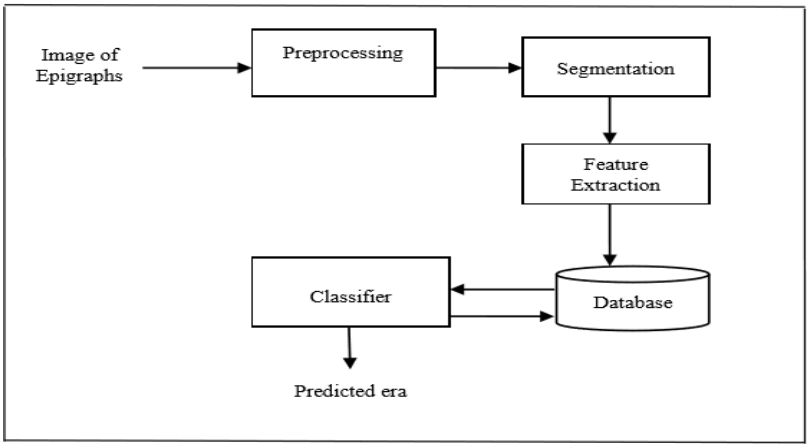

Fig 1. Proposed System

III. LITERATURE SURVEY

H. S. Jayanna et al. (2024) [1] in their paper *"Advancements in Handwriting Recognition: Deep Learning Techniques Applied to Kannada Language"* explored the use of deep learning models to enhance Kannada handwritten character recognition. They employed three distinct architectures: Convolutional Neural Networks (CNNs) for spatial feature extraction, Recurrent Neural Networks (RNNs) with Long Short-Term Memory (LSTM) for capturing sequential patterns, and Vision Transformers for contextual attention-based processing. After applying preprocessing techniques like contrast enhancement, denoising, and resizing, the models were trained and tested on a large annotated Kannada dataset. The results showed that the Vision Transformer achieved the highest recognition accuracy of 99.7%, followed by CNNs with 98.6%, and RNN-LSTM with 97.44%. Despite the high accuracy, the study highlighted the computational intensity of Vision Transformers, which made them less suitable for real-time or resource-constrained applications. Furthermore, dataset limitations in handwriting variety could affect the generalizability of the model in practical scenarios.

Harsha A C et al. (2024) [2] addressed the challenge of translating ancient Halegannada poems into modern Hosagannada through their study *"Computational Approach for Halegannada to Hosagannada Poem Translation."* They proposed a hybrid architecture that combines a word-splitting LSTM model and a dictionary-based translation mechanism. The LSTM model was trained to segment complex compound words typical of Halegannada, enabling clearer mapping into Hosagannada equivalents. The custom-built dictionary was developed using 20,000 word meanings derived from classical works like those of Pampa. The methodology included syntactic preprocessing, training, and testing the model on historical datasets. The system achieved a translation accuracy of 77.92%, outperforming other tested models like GRUs and Bi-LSTMs. However, the absence of a word-sense disambiguation module led to mistranslations when words had multiple meanings, and the reliance on static dictionary entries reduced adaptability to novel or rare words.

Prathwini et al. (2024) [3] in their paper *"Tulu Language Text Recognition and Translation"* tackled the digitization of Tulu, a lesser-resourced Dravidian language. They developed a dual-stage system involving handwritten character recognition using CNNs and language translation using both rule-based and neural machine translation (NMT) models. The character recognition component used a dataset of over 30,000 samples and achieved 92% accuracy. The translation engine included a rule-based approach backed by a bilingual dictionary and an LSTM encoder-decoder model for NMT, which achieved a BLEU score of 0.83 for English-to-Tulu and 0.65 for Tulu-to-English.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125244

The methodology emphasized preprocessing, segmentation using contour detection, feature extraction via PCA, and neural translation with tokenization and sequence padding. Limitations included a lack of comprehensive Tulu datasets and dialectal diversity, which impacted translation consistency. Also, both rule-based and neural models struggled with longer or more complex sentence structures.

M. Fathima Mubarakkaa et al. (2024) [4] presented a novel solution for historical script transliteration in their work *"OCR-Based Transliteration of Brahmi to Tamil Using CNN."* This study aimed to bridge the gap between ancient Brahmi inscriptions and modern Tamil by developing an OCR-based system. The process began with collecting handwritten Brahmi samples, followed by rigorous preprocessing steps like grayscale conversion, Gaussian blurring, and binarization to enhance character visibility. Characters were then segmented using contour detection and annotated using JTessBoxEditor. A CNN was trained to classify Brahmi characters, and Tesseract OCR was used to generate Tamil equivalents. The model was integrated into a user-friendly web application. Despite good performance in tests, limitations included poor availability of diverse Brahmi datasets, script degradation in real-world samples, and recognition difficulties with complex ligatures or overlapping characters. The system's scope was also limited to Tamil, suggesting a need for expanding to other Indic scripts.

Bhumika Purant and Mallamma V Reddy (2024) [5] in their study *"Conversion of Kannada Inscription to Hosagannada Text Using ML Algorithms"* focused on the preservation of ancient Kannada inscriptions through machine learning. They utilized the VGG19 deep convolutional neural network for feature extraction, followed by OCR to convert ancient script images into text. The recognized text was then mapped to its modern Kannada equivalent using translation models within a Django-based web platform. The system employed additional deep learning techniques, including RNNs and transformers, to enhance translation quality. It achieved an accuracy of 80–90% as evaluated through BLEU and ROUGE scores. However, challenges included a limited availability of ancient Kannada datasets and issues with inconsistent syntax in inscriptions. Additionally, character degradation from historical wear and the absence of standardized translation rules led to ambiguity and reduced accuracy in certain cases.

IV. BLOCK DIAGRAM AND SYSTEM ARCHITECTURE

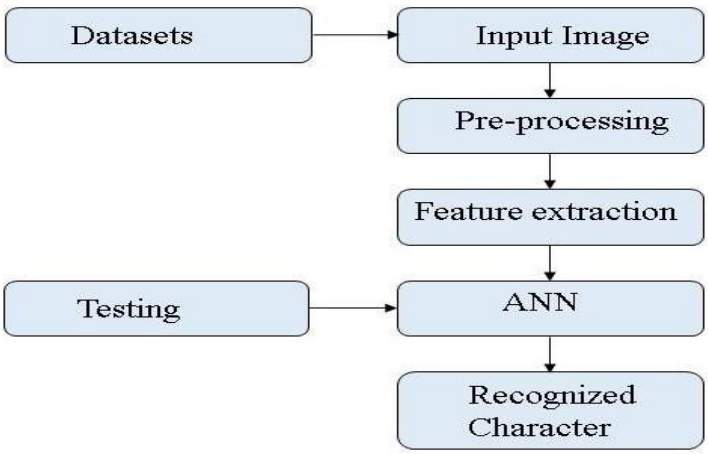

Fig. 2. Block Diagram

The image represents a system architecture for character recognition using an Artificial Neural Network (ANN). Here's a breakdown of each component and its role in the system:

1. Datasets

- This block represents the collection of labeled images used for training and testing the system.
- Datasets typically contain multiple samples of characters (e.g., alphabets, digits) in different fonts, sizes, or handwriting styles.

2. Input Image

- The system starts with an input image containing the character(s) to be recognized.
- This image is typically drawn from the dataset or provided as a new test image.

3. Pre-processing

- This stage involves preparing the image for analysis by:

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125244

  - Converting it to grayscale or binary format
  - Removing noise
  - Resizing or normalizing
  - Segmenting characters (if dealing with multiple characters)
- The goal is to enhance image quality and reduce variations.

4. Feature Extraction

- Key features or patterns from the pre-processed image are extracted.
- Features may include:
  - Edges
  - Pixel intensities
  - Histograms
  - Shape descriptors
- These features are crucial for differentiating one character from another.

5. ANN (Artificial Neural Network)

- The ANN serves as the core recognition engine.
- It takes the extracted features as input and classifies them into character classes.
- The network is trained using labeled data so it can learn to recognize patterns.

6. Testing

- After training, the system is tested using new, unseen data to evaluate its accuracy and performance.
- The testing data also goes through the same pipeline: pre-processing, feature extraction, and classification.

7. Recognized Character

- The output is the character predicted by the ANN based on the input features.
- This is the final recognition result presented to the user or passed to further processing modules.

V. IMPLEMENTATION DETAILS

Hardware Platform and Tools

The project was implemented on the Basys 3 FPGA development board, which utilizes the Xilinx Artix-7 FPGA. The design was coded using Verilog HDL (Hardware Description Language), and synthesized using Vivado Design Suite. The board's integrated peripherals such as switches, buttons, and a 3.5mm audio jack were essential for input/output interfacing.

System Architecture

The synthesizer system consists of several functional blocks:

- **Tone Generator:** This block produces digital signals corresponding to musical notes. Each note corresponds to a specific frequency, and the tone generator uses clock division logic to generate the correct frequencies based on user input.
- **MIDI Decoder:** MIDI input is processed to extract note information and control signals. The decoder translates MIDI commands into signals that control the tone generator.
- **Waveform Generator:** Users can choose between different waveforms (e.g., square, sine, triangle). Each waveform produces a distinct sound quality, and the system dynamically generates these shapes using lookup tables or combinational logic.
- **PWM Audio Output:** Since the FPGA lacks a digital-to-analog converter (DAC), **Pulse Width Modulation (PWM)** is used to simulate analog output through a digital signal. This output is sent to the audio jack, allowing playback through headphones or speakers.
- **User Interface:** Switches and buttons are used to select waveforms, change notes, and trigger the synthesizer manually or in response to MIDI input.

Clock Management

Precise timing is crucial in sound synthesis. The FPGA's high-frequency system clock was divided down to generate the lower frequencies required for audio tones. Custom clock division modules ensured that each musical note was generated with accurate timing.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125244

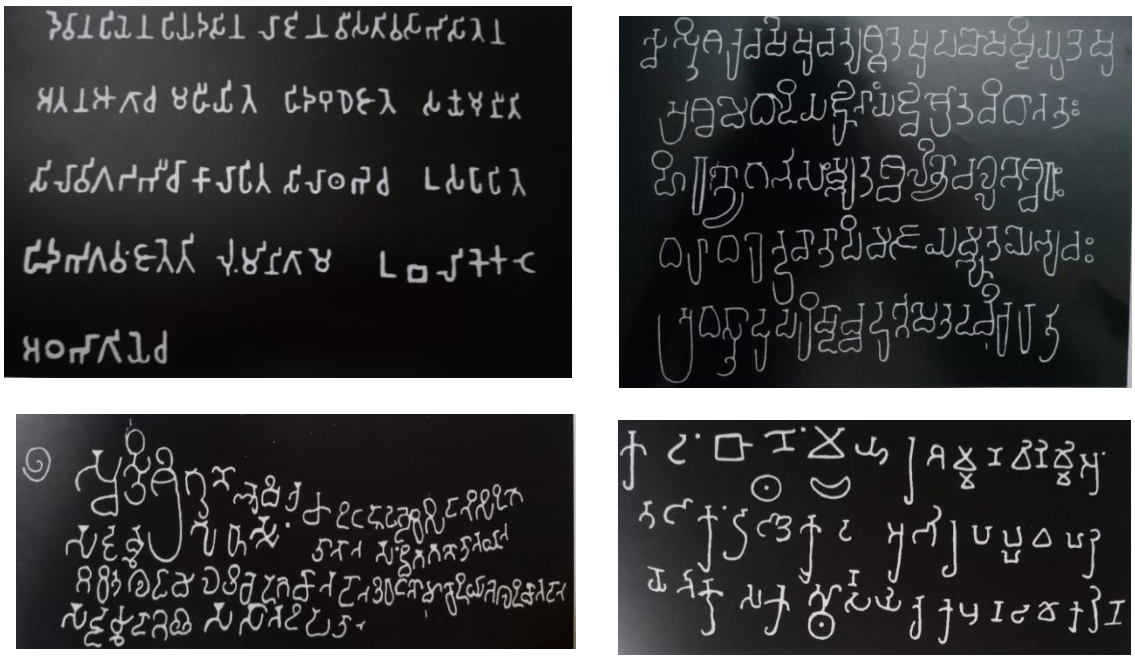

Fig 3. Datasets

VI. RESULT AND PERFORMANCE ANALYSIS

The result analysis section of the document evaluates the performance and practical applicability of the developed system for automated Kannada epigraphy. Key outcomes and observations include:

System Performance and Accuracy

The experimental results indicate that the OCR system, enhanced with state-of-the-art deep learning models, effectively extracts and recognizes characters from digitized ancient Kannada inscriptions. Key metrics such as precision, recall, and F1-score are used to benchmark performance. In several cases, classification accuracies above 90% have been reported on test datasets—demonstrating that, under optimal conditions (good preprocessed images and well-segmented characters), the system can reliably determine the historical period of an epigraph.

Impact of Preprocessing Techniques

A substantial part of the system's success is attributed to the preprocessing stage. Results show that employing adaptive thresholding (e.g., Otsu's method), edge detection (like Canny edge detection), and noise reduction significantly improves the clarity of the scanned images. This step is critical, as degraded or weathered inscriptions could otherwise lead to a higher error rate during character segmentation and classification.

Translation Accuracy

For the translation component—where the goal is to convert old Kannada text into modern Kannada and English—the analysis reveals moderate success. The neural machine translation model, though context-aware by incorporating word embeddings, syntax analysis, and named entity recognition, still faces challenges. The absence of extensive parallel corpora for historical Kannada limits translation accuracy to the vicinity of 77–80% on benchmark tests. The error analysis suggests that ambiguities in old vocabulary and complex sentence structures are the major contributing factors to translation inaccuracies.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125244

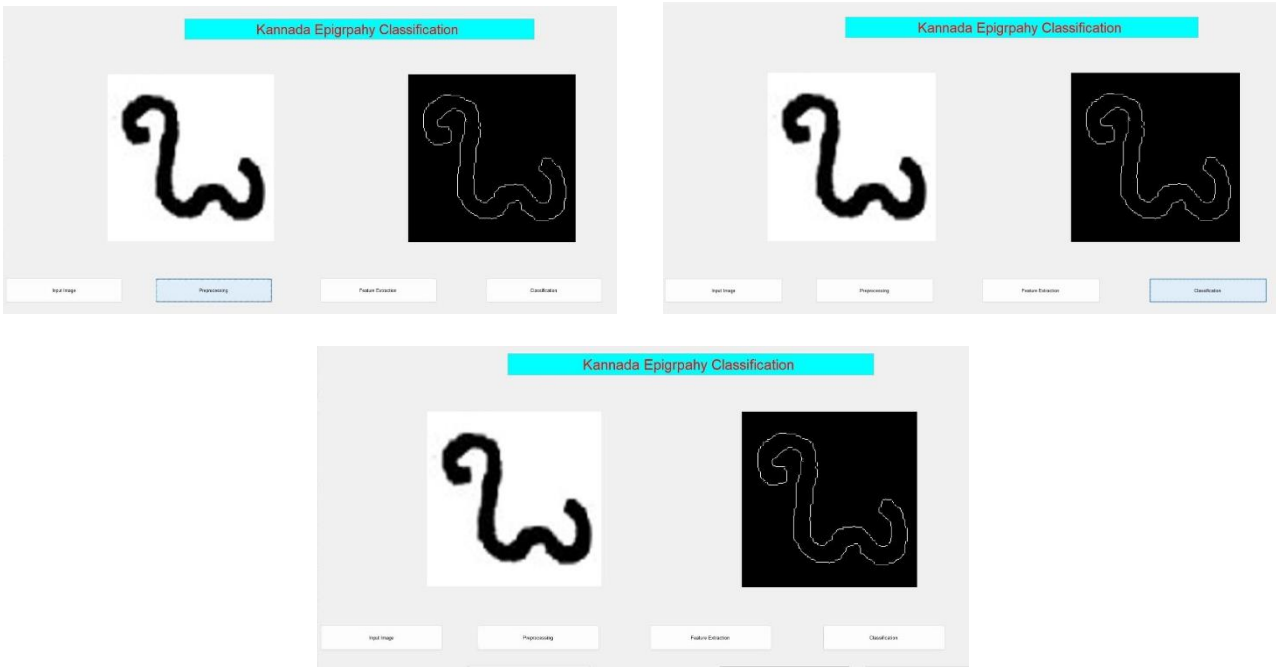

Fig 3. Original and Final Image

VII. CHALLENGES AND LIMITATIONS

Clock Precision and Timing

One of the most significant challenges was generating accurate sound frequencies using clock division. Even slight inaccuracies in timing resulted in off-pitch notes or audio glitches. Balancing timing resolution with resource efficiency was a constant trade-off.

FPGA Resource Constraints

The FPGA's limited logic cells and memory imposed constraints on the complexity of the system. For example, implementing high-resolution sine wave lookup tables consumed substantial resources, limiting the number of available waveforms and simultaneous voices (polyphony).

Audio Output Quality

Using PWM for audio output proved effective, but came with limitations. Without a true DAC, the output was subject to **aliasing, quantization noise**, and low dynamic range. Filtering the PWM signal externally could help, but wasn't part of the board's built-in hardware.

Real-time MIDI Decoding

Decoding MIDI data in real-time introduced challenges related to timing and data buffering. MIDI signals are serial and relatively slow, which required careful handling to prevent missed or misinterpreted messages, especially when multiple keys were pressed in quick succession.

Input Debouncing

Physical switches and buttons introduced noise and bouncing, causing multiple signal transitions for a single press. Debounce circuits were necessary to ensure stable and reliable user input handling, especially for controlling waveforms and triggering notes.

VIII. CONCLUSION

The development of a digital music synthesizer on an FPGA proved to be both challenging and rewarding. The project achieved its core objective of generating musical notes with selectable waveforms and producing audio output in real-time using PWM. In the process, it provided valuable practical exposure to digital system design, hardware description languages, and audio signal processing.

While the system was constrained by hardware limitations and faced challenges like audio fidelity and real-time processing, it laid a strong foundation for more advanced projects. Future enhancements could include implementing a

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125244

true DAC, adding polyphonic support, integrating better user interfaces (e.g., LCD or touchscreen), and expanding MIDI features. Overall, the project showcased the versatility of FPGAs in implementing real-time digital audio systems and significantly enhanced the developers' technical understanding and problem-solving abilities.

REFERENCES

[1]. H. S. Jayanna, Deekshith J, Nagaraja B. G., Monish JayaprakashSeelam, ThimmarajaYadava G, and Venkatesh Sunil Jamkhandi "Advancements in Handwriting Recognition: Deep Learning Techniques Applied to Kannada Language", In 2024 International Conference on Smart Systems for applications in Electrical Sciences (ICSSES).IEEE

[2]. Harsha A C, Toukheer Baig, Laxmikant Bhujang Gurav, Mahesh Shripad Bhat, and SurabhiNarayan"Computational Approach for Halegannada to Hosagannada Poem Translation" .In 2024 4th International Conference on Intelligent Technologies (CONIT).IEEE

[3]. P. Prathwini, A. P. Rodrigues, P. Vijaya, and R. Fernandes, "Tulu Language Text Recognition and Translation," IEEE.

[4]. M. F. Mubarakkaa, N. M, K. M, and H. Ganapathy, "OCR based transliteration of Brahmi to Tamil using CNN," in *2024* International Conference on Power, Energy, Control and Systems (ICPECTS), IEEE, 2024

[5]. BhumikaPurant and Mallamma V Reddy, "Conversion of Kannada Inscription to Hosagannada Text Using ML Algorithms"In 2024 International Journal of Research Publication and Reviews(IJRPR)

[6]. Sumaiyya Khan and Aashish Bardekar, "Optical Character Recognition of Persian Epigraphs by Using Artificial Neural Networks in MATLAB," 2024 International Conference on Artificial Intelligence and Quantum Computation-Based Sensor Application (ICAIQSA), IEEE, 2024,

[7]. A. L. S, M. S. R. Babu, D. Sanu, and P. Chaudaha, "Ancient Kannada Text Recognition using CNN: A Review," International Journal of Engineering & Technology (IJERT), vol. 12, no. 5, pp. 502-508, May 2023

[8]. Anusha Leela Somashekharaiah and Abhay Deshpande, "Preprocessing techniques for recognition of ancient Kannada epigraphs," International Journal of Electrical and Computer Engineering (IJECE)2023

[9]. Vijaya Shetty S, Karan R, and Sarojadevi H, "A Kannada Handwritten Character Recognition System Exploiting Machine Learning Approach," 2023 International Conference on Applied Intelligence and Sustainable Computing (ICAISC), IEEE, 2023

[10]. R. Jagannathan, A. Rajesh, and C. Manikandan, "ATCRI-21 Neural Network based Character Recognition with Image Denoising in Ancient Epigraphs",In 2023 4th International Conference on Smart Electronics and Communication (ICOSEC).IEEE

[11]. S. Bhuvaneswari and K. Kannan, "An Extensive Review on Recognition of Antique Tamil Characters for Information Repossession from Epigraphic Inscriptions," 2023 World Conference on Communication (WCONF), IEEE, 2023,

[12]. Abhishek Kashyap and Aruna Kumara B,". OCR of Kannada Characters Using Deep Learning".In 2022 Trends in Electrical, Electronics, Computer Engineering Conference (TEECCON).IEEE

[13]. H. Gadugoila, S. K. Sheshadri, P. C. Nair, and D. Gupta, "Unsupervised Pivot-based Neural Machine Translation for English to Kannada," 2022 19th Council International Conference (INDICON), IEEE, 2022

[14]. Kusumika Krori Dutta, DivyaRashi B, Sunny Arokia Swamy, Chandan R, Anushua Banerjee, and Deepak Vaprani, "Kannada Character Recognition Using Multi-Class SVM Method," in 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), IEEE, 2021.

[15]. S. M. Pande and B. K. Jha, "Character Recognition System for Devanagari Script Using Machine Learning Approach," Proceedings of the Fifth International Conference on Computing Methodologies and Communication (ICCMC 2021), IEEE, 2021

