**DECIPHERING TAMIL CHARACTERS FROM ANCIENT PALM LEAF MANUSCRIPTS**

## Ms.J.Juslin Sega1, Dr.J.Shiny Duela2

## 1 Assistant Professor, Computer Science and Engineering, SRM INSTITUTE OF SCIENCE AND TECHNOLOGY,RAMAPURAM,CHENNAI.

#### 2 Associate Professor, Computer Science and Engineering, SRM INSTITUTE OF SCIENCE AND TECHNOLOGY,RAMAPURAM,CHENNAI.

#### **1*****rjsega263@gmail.com*****,** **2shinyduj@srmist.edu.in****,**

ABSTRACT

Tamil is one of the oldest languages in the world, with a rich cultural and literary heritage that is conserved in both written and oral forms. Palm leaves were used to write manuscripts and they have been used as a medium to conserve knowledge such as in the field of astronomy, medicine, art, literature etc. There remains a growing need for transcription and digitizing palm leaf manuscripts and recognizing Tamil cursive characters remains a challenging task. This study uses Convolutional Neural Network (CNN) technique to train the characters and characteristics present in the palm leaf, allowing it to significantly perform classification of palm leaf characters in the training session. Background noise were removed during preprocessing phase. Connected component Analysis is then used for segmentation which is a technique used in image processing to identify connected regions in a binary image. Then we perform feature processing that includes text line spacing, spacing without obstacle, and spacing with an obstacle. Finally, datasets is fitted into the model for final classification of the characters. Finally, experiments performed on the collected characters is used to obtain the overall performance of the CNN model and it is calculated with the help of accuracy, recall, precision etc.

1. **INTRODUCTION**

Tamil is a language with a rich literary heritage and is recognized as one of the oldest in the world. In ancient times, poets used Palm leaves, particularly in Tamil Nadu, to conceal information. The ancient literature includes masterpieces, such as Sangam literature, Vaishnava, Saiva, medicinal works, gastronomy, astrology, Vaastu, gems, music, dance, and theatre, as well as Siddha. There has been growing interest among academics in the past decade to preserve ancient medical texts in Tamil and their value. To preserve medical materials, numerous scholars have generated conserved old medical writings in Tamil, such as those by saints like Agathiyar, which have undergone the first phase of a digitalization process. Nearly 10,000 manuscripts have been successfully scanned. For digitizing historical documents, apps that identify handwritten characters have used three key methods: statistical, structural or syntactic, and neural network-based techniques

**2.LITERATURE REVIEW**

[1] A novel ETEDL-THDR technique was developed specifically for the identification of tropical hazardous chemicals (THCs). This approach integrates various essential elements, such as MobileNet feature extraction, BiGRU (Bidirectional Gated Recurrent Unit) recognition, and WSO (Weighted Sum Optimization) for hyperparameter tuning. The WSO algorithm effectively adjusts the hyperparameters of the BiGRU model to optimize recognition accuracy. Rigorous experimental studies were conducted to validate the enhanced performance of the ETEDL-THDR model. Comparative analysis clearly illustrates the superior performance of the ETEDL-THDR approach compared to recent methodologies. Future research directions may involve exploring an ensemble approach with three deep learning-based fusion models to further elevate recognition outcomes.

[2] Character classification, particularly for identifying Old English characters in the Beowulf manuscript, was the focus of our study. We developed a convolutional neural network (CNN) model specifically tailored for this task. Our approach involved training and testing the model using the dataset from the Beowulf manuscript. Additionally, we conducted comparative analyses with various machine learning (ML) models including support vector machines (SVM), nearest neighbors (KNN), decision trees (DT), random forests (RF), and XGBoost. To extract features from the Beowulf manuscript character images, we utilized pretrained models such as VGG16, MobileNet, and ResNet50. These features were then used to train the SVM, KNN, DT, RF, and XGBoost models, followed by testing on our Beowulf test dataset. Recognition accuracy results were recorded, and model performance was evaluated based on recall, precision, and F1 score metrics. Each model's classification performance was visualized using ROC curves (Figure 9). To enhance the dataset, we augmented the Beowulf manuscript character images once, twice, and three times. Our proposed CNN model exhibited superior performance compared to other ML models, achieving the highest accuracy of 98.86% with threefold resampling. Furthermore, we evaluated the CNN model using the MNIST handwritten digits dataset, achieving a benchmark recognition accuracy of 99.03%

[3] The median filter is utilized to reduce noise by replacing each pixel's value with the median value within a specified window. One key limitation of using a 2D adaptive median filter is the loss of edge information that occurs when images undergo this filtering process. Similarly, the primary drawback of employing a 2D adaptive Wiener filter is its effectiveness being limited to scenarios where the variance of noise components in the image is relatively low. Another significant issue arises with the 2D adaptive log filter, as it tends to degrade the image quality. As noise components are minimized through this filter, the image quality diminishes progressively.

[4] This paper presents machine learning techniques for recognizing Urdu characters. Initially, datasets were collected for Urdu characters. Basic techniques such as the confusion matrix, ROC curve, and K-fold cross-validation are performed initially. It is followed by preprocessing and segmentation, which provide the character skeleton.

Among SVM, SMO, MLP, and simple logistic classifiers, SVM (Support Vector Machine) provides a high accuracy of over 98%.

[5] This paper introduces a novel approach to character recognition using feature fusion within a machine learning framework. The primary emphasis of this study revolves around two specific classifiers: neural networks and least squares support vector machines (LSSVM). Additionally, the paper delves into the structural aspects of a character recognition system grounded in machine learning principles. Furthermore, the document explores the merits and drawbacks associated with character recognition technologies and proposes a method that integrates feature fusion and machine learning for enhanced performance.

[6] This paper describes the process of building a Gujarati poetry corpus by collecting and curating poems from various poets in Gujarati literature. Each word within the corpus is then cross-referenced with a manually annotated metadata set that categorizes words according to the concept of "Rasa," aligning with the traditional Indian theory of "Navarasa." By applying this method to each poem, a counter value is computed, representing the predominant emotional theme ranging from 0 to 8. This value is then compared against a dictionary named 'ras' to determine the specific 'Rasa' associated with the poem, classifying it according to the principles of "Navarasa." This methodology efficiently extracts metadata from Gujarati poetry, encompassing a wide spectrum of emotional expressions inherent in the poems. The authors have endeavored to develop an automated system capable of categorizing poems based on their emotional content using the traditional Indian framework of 'Navarasa.'

[9] This study presents the challenges of character recognition in palm leaf manuscripts and their solutions. Convolutional Neural Network (CNN) is typically addressed, which is then followed by pattern recognition using OCR and textual processing analysis. Various optical character recognition (OCR) techniques are employed and compared, out of which the Tamil optical character recognition system and the recognition of CAPTCHA characters by supervised machine learning algorithms have an efficiency of 99%. These findings can be applied to palm leaf manuscripts of different languages as a kickstart to the complex character and text recognition problems.

[10] The solution for Tamil character recognition involves the implementation of a ResNet-50- based system, encompassing a database, an algorithm, and an application. The dataset comprises over 15,000 images, each with dimensions of 128x128 pixels. Notably, this proposed Tamil character recognition system attains an impressive accuracy rate of 96%.

The system's potential lies in establishing a robust handwritten documentation and digitization platform. Moreover, there's potential to expand into Tamil word recognition, integrating with an open-source database for all 256 Tamil characters. ResNet incorporates an "identity shortcut connection" layer, allowing it to bypass redundant layers and reuse activation layers, thus preventing excessive complexity. Rigorous analysis methods ensure the system's efficiency and accuracy. By effectively addressing the limitations of previous methods, this working model consistently achieves a remarkable accuracy rate of 96%, surpassing the performance of alternative systems.Ali and Joseph [8] developed a CNN perfect for dispensing real-time input pictures including Malayalam characters and the job of segmenting words and typescripts from an image and attractiveness prediction using the CNN model. The model's feature extraction process for digitizing the Malayalam script seamlessly incorporates convolutional neural network (CNN) techniques. This intricate procedure, encompassing recognition of 36 consonants and 13 vowels, unfolds progressively, achieving an outstanding accuracy of 97.26% on the training dataset.

In alignment with the insights shared by Narenthiran and Ravichandran [13], historical India bore witness to the documentation of diverse knowledge systems on palm leaf scrolls. The repositories of ancient wisdom predominantly expressed their contents through an array of characters from the Sanskrit language. Traditional knowledge, in accordance with Devika and Vijayakumar [12], aids in establishing lasting relationships between people and nature. This study aimed to distinguish between the contents of palm leaf documents that can be digitised, the specifics for digitization, and the various methods of palm leaf document scanning

**3. PROPOSED METHODOLOGY**

* 1. **Dataset:**Tamil palm leaf manuscript datasets were obtained from online sources and stored for the purpose of character extraction. In total, 100 samples of palm leaf manuscripts were collected. Traditionally, each set of palm leaves is bound together using cord threads inserted through two holes that are pierced through the entire stack, often reinforced with bamboo strips. The assembled bundle is then secured with heavy wooden covers on both sides and is either tied with cords or wrapped in a soft cloth for preservation.

**3.2 Background normalization / Background removal:**

Historical documents often present two major challenges: the first being physical deterioration or decay over time, and the second arising from digital conversion, which frequently results in uneven or noisy backgrounds. To overcome these issues, image enhancement techniques are employed—particularly useful for improving the clarity of low-contrast images. These methods help reduce background irregularities, making the textual content more accessible for further processing.

The preprocessing phase begins with a pre-enhancement step applied to the input image III, where a linear contrast stretching function expands the grayscale values to the full dynamic range. This adjustment enhances the brightness and contrast of the image, thereby improving the separation between text and background. It effectively restores sections of the text that may appear faded or unclear.

While this enhancement may slightly increase image noise by intensifying already bright pixels, it is a necessary compromise to ensure the preservation of faint textual details, especially in DIBCO benchmark datasets and palm leaf manuscripts. The method also includes a noise reduction step to eliminate any additional artifacts introduced during enhancement. The resulting enhanced image IEI\_EIE​ is then used to generate the gradient image GdG\_dGd​, which serves as input for the subsequent processing stages.

[Start] --> [Input Image I] --> [Pre-enhancement using contrast stretching] --> [Enhanced Image IE] --> [Calculate Gradient Image Gd] --> [Noise Removal] --> [Binarization] --> [Output Result]

**3.3 Morphological Operations**

The **opening** operation is a fundamental technique in image processing, frequently used in applications like object detection and feature extraction. It helps remove irrelevant background noise, thereby enhancing the precision of analysis. In the context of Optical Character Recognition (OCR), opening is particularly useful during preprocessing, as it improves both the quality and legibility of textual data.

In formal mathematical terms, the opening operation is defined as:

A∘B=(A⊖B)⊕BA \circ B = (A \ominus B) \oplus BA∘B=(A⊖B)⊕B

Where:

* AAA represents the original image,
* BBB is the structuring element that defines the shape and size of the operation,
* ⊖\ominus⊖ stands for erosion,
* ⊕\oplus⊕ denotes dilation.

**Grayscale morphology** is an extension of traditional morphological operations that operates on grayscale images instead of binary ones. While it employs the same basic operations—erosion, dilation, opening, and closing—the structuring elements used in grayscale morphology contain varying intensity values rather than binary ones. This allows each pixel in the structuring element to contribute proportionally to the outcome, offering finer control over the processing.

This capability makes grayscale morphology especially effective in handling images with complex or gradual intensity variations. It is widely used in tasks such as edge detection, image segmentation, and feature extraction. Moreover, it is invaluable in high-precision fields like medical imaging and remote sensing, where detailed and accurate analysis of image data is essential.

**4. FEATURE PROCESSING**

Connected Component Analysis (CCA) is a technique in image processing used to identify and label individual connected regions in a binary image. In such an image, foreground pixels (representing the object of interest) are set to one, while background pixels are set to zero. The CCA algorithm identifies groups of connected foreground pixels and assigns each group a unique label. This labeled image can then be used for further processing, such as character recognition or object detection. Various algorithms, like the two-pass algorithm or depth-first search (DFS), can be used for CCA. It's particularly useful in tasks like Optical Character Recognition (OCR), where it helps to identify individual characters in a text image.

In the context of Tamil Win Flake scripts, CCA is applied to pre-processed text images to segment text lines. The presence of a "handicap," where strokes from one line extend into the space between text lines, complicates this process. When such strokes occur, they are considered a "handicap," which impacts line segmentation and must be addressed during the recognition process. This approach helps in accurate text line segmentation by identifying whether such overlaps are present between lines.

Connected Component Analysis plays a crucial role in Optical Character Recognition (OCR) systems, where it helps in isolating and identifying individual characters, despite challenges like overlap between lines in complex scripts such as Tamil.

 ⮚ First, the image is binarized by applying a thresholding operation to convert it into a double image. Let I(x, y) be the input image and B( x, y) be the binarized image, also the thresholding operation can be represented as B( x, y) = { 1 if I( x, y)> T{ 0 if I( x, y) ≤ T

 ⮚ Next, the connected factors in the double image are linked. This is done by assigning a unique marker to each group of connected pixels in the double image. The labels are supporter integers starting from 1 and incremented for each new connected element. The process can be represented as L( x, y) = { 0 if B( x, y) = 0{ L( p) if B( x, y) = 1 and p is a neighbour of( x, y){ new marker if B( x, y) = 1 and p is not a neighbour of( x, y) where L( x, y) is the marker assigned to the pixel( x, y), p is a neighbour of( x, y), and new marker is a unique marker not previously assigned.

 ⮚ ultimately, the connected factors are segmented by lodging the pixels corresponding to each marker. This can be represented as = {(x, y)| L( x, y) = k}where S\_k is the set of snaps- rails belonging to the k- th connected element.

**5. CONVOLUTIONAL NEURAL NETWORKS**

A widely used deep learning framework for image recognition and classification tasks is the Convolutional Neural Network (CNN). It consists of several layers, including convolutional, pooling, and fully connected layers. The convolutional layer uses filters to extract features from the input image, while the pooling layer reduces the image's size to enhance computational efficiency. Finally, the fully connected layer is responsible for making the final prediction. The network learns the best filters through techniques like multi-scattering and gradient descent.

5.1 CNN Architecture

Convolutional Layer (CL)

In the convolutional layer, the output is produced by applying a convolution operation between the input image and a filter. This operation involves computing the dot product of the filter with the corresponding pixels in the input image. Mathematically, this can be expressed as:

Output = (Filter \* Input Image) + Bias

Where:

'\*' denotes the convolution operation,

Filter is the learned parameter in the convolutional layer,

Input Image refers to the input image data,

Bias is a scalar value added to the output elements.

The result of applying a single filter is a 2D feature map, and the combined output from all filters results in the complete 2D feature map for the input image.

The convolution process involves sliding the filter (also called a kernel or feature map) over the input image in a sliding window manner, calculating the dot product between the filter's weights and the input pixel values. This generates a feature map that captures local features such as edges, corners, and other significant patterns in the image. The convolution operation can be represented mathematically as:

 y(i,j,k) = ∑∑∑ x(m,n,l) \* w(i-

m+1,j-n+1,l,k)

where y (i, j, k) is the output feature map at

location (i, j) and pixel (m, n) at location x

(m, n, l) for the whale filter and channel for

lath, and w (i-m ). + 1, j-n + 1, l, k) is the

weight of the filter at the location (i-m + 1,

j-n + 1) and the lath channel and whale

filter.

Pooling can be done using different methods such as peak pooling or average

pooling where the maximum or average value is stored in a sliding window. Pooling

helps reduce the number of parameters in

the set, making changes to the input image

more efficient and reliable. The mathematical operations performed by the

pool layer can be defined as:

y(i,j,k) = f({x(i&#39;,j&#39;,k) | i&#39; ∈ [iS, iS+H), j&#39; ∈

[jS, jS+W)})

where y(i,j,k) is the output value at location

(i,j) and x(i&#39;,j&#39;,k) is the input value at location (i&#39;,j&#39;) for the whale channel and S is the step for whale channel, H and W are the height and width of the pool window, respectively. The function f (.) can be either a maximum or an average pool.

The output of a convolutional layer in a neural network (CNN) can be expressed

mathematically as:

h(i,j,k) = f(∑∑∑ x(m,n,l) \* w(i-

m+1,j-n+1,l,k) + b(k))

where h(i,j,k) is the output feature map

at position (i,j) and for the whale filter

x(m,n,l) is the input value at position (m,n)

and lath is the channel. , w(i-m+1, j-n + 1, l,

k) is the condition of the filter (i-m + 1, j-n

+ 1) and for lath channels and whale filters,

b(k) is the bias term. for the whale filter, f

(.) Activation function.

**6. Results and Discussion**

As mentioned earlier, the goal of this work is to create a CNN model that is more efficient than existing models for the received course data set. Proposal for neural construction is

PyTorch7 is used as a Python-based framework for network architecture. Several state-of-the-art models such as LeNet5, ResNet (18/34/50), Alex Net, DenseNet121, Inception Net v3 and others have been evaluated in this database to conduct comparative studies. For all these tests, an Intel Core i3 processor, 16 GB of RAM and 4 GB of internal memory and an NVIDIA graphics card use 768 CUBA cores.

**References**

 [1] D. Ganapathy, “Preserving India’s palm Leaf Manuscripts for the Future,” WAGLOBAL, Kerala, India, 2016.

[2] N. S. Panyam, V. L. T.R., R. Krishnan, and K. R. N.V., “Modeling of palm leaf character recognition system using transform based techniques,” Pattern Recognition Letters, vol. 84, pp. 29–34, 2016.

 [3] K. P. Geena and G. Raju, “View-based feature extraction and classification approach to Malayalam palm leaf document image,” International Journal of Innovative Research in Computer and Communication Engineering, vol. 2, no. 5, pp. 264–267, 2014.

[4] R. Chamchong and C. C. Fung, “A framework for the selection of binarization techniques on palm leaf manuscripts using support vector machine,” Advances in Decision Sciences, vol. 2015, Article ID 925935, 7 pages, 2015.

[5] N. P. Challa and R. V. K. Mehta, “Applications of image processing techniques on palm leaf manuscripts-A survey,” in Proceedings of the Conference on Cognitive Science and Artificial Intelligence, CA, USA, February 2017.

[6] S. Athisayamani, A. R. Singh, and A. S. Kumar, “Recurrent neural network-based character recognition system for Tamil palm leaf manuscript using stroke zoning,” in Inventive Communication and Computational Technologies, pp. 165– 176, Springer, Singapore, 2021.

[7] R. S. Ratheash and M. M. Sathik, “A detailed survey of text line segmentation methods in handwritten historical documents and palm leaf manuscripts,” International Journal Of Computer Sciences And Engineering, vol. 7, p. 99, 2019.

[8] J. Ali and J. T. Joseph, “A convolution neural network-based approach for recognizing Malayalam handwritten characters,” Malayalam Handwritten character recognition using cnn, vol. 9, no. 12, 2018.

[9] P. K. S. Balakrishnan and L. Pavithira, “Multi-font optical character recognition using Deep Learning,” International Journal of Recent Technology and Engineering, vol. 8, 2019.

[10] M. A. Hossain and S. Afrin, “Optical character recognition based on template matching,” Global Journal of Computer Science and Technology: C Software & Data Engineering, vol. 19, 2019.

 [11] K. Baskar, “Manuscript Libraries in Tamil Nadu: a study of their organisation and preservation in the digital environment,” Doctoral Dissertation in Information Science, Madras University, Tamilnadu, 2018.

[12] S. Devika and K. Vijayakumar, “Digitization of palm leaf manuscripts in Tamil Nadu (India): a study,” Journal of Library Science and Research (JLSR), vol. 2, no. 1, pp. 1–10, 2016.

[13] R. Narenthiran and P. Ravichandran, “Cataloguing and digitization of multilingual manuscript libraries in Tamil Nadu: an evaluative study,” Journal of Advances in Library and Information Sciences, vol. 5, no. 3, pp. 248–253, 2016.

[14] T. K. M. Sageer and A. T. Francis, “Analysis of the palm leaves manuscripts collection for digital archiving: a case study of Sree Sankaracharya University of Sanskrit, Kalady,” Journal Impact Factor, vol. 5, no. 1389, pp. 90–99, 2016.

[15] R. S. Sabeenian, M. E. Paramasivam, P. M. Dinesh, R. Adarsh, and G. R. Kumar, “Classification of handwritten Tamil characters in palm leaf manuscripts using SVM based smart zoning strategies,” in Proceedings of the 2nd International Conference on Biomedical Signal and Image Processing, pp. 18–21, Kitakyushu, Japan, August 2017.

[16] B. Kiruba, A. Nivethitha, and M. Vimaladevi, “Segmentation of handwritten Tamil character from palm script using histogram program approach,” International Journal of Informative and Futuristic Research, vol. 4, no. 5, pp. 6418–6424, 2017.

[17] S. Ghosh, A. Mahajan, and S. Banerjee, “Palm leaf manuscript conservation, the process of seasoning with special reference to Saraswati Mahal library, Tamilnadu in India: some techniques,” International Journal of Information Movement, vol. 2, pp. 122–128, 2017.

[18] N. P. Challa and R. V. K. Mehta, “Applications of image processing techniques on palm-leaf manuscripts—a survey,” Helix: e Scientific Explorer, vol. 7, no. 5, pp. 2013–2017, 2017.

[19] R. Vinoth, R. Rajesh, and P. Yoganandhan, “Intelligence system for Tamil Vattezhuttuoptical character recognition,” International Journal of Computer Science Engineering and Technology, vol. 8, no. 4, pp. 22–26, 2017.

[20] M. Sornam and M. D. Poornima, “Tamil palm-leaf manuscript character segmentation using GLCM feature extraction,” International Journal of Computer Science and Engineering, vol. 6, no. 4, pp. 167–172, 2018.