ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1384

A System for Recognition and Digitalization of Tamil Handwritten Document

Dr. S. Prakasam[1]; S. Yuvan Shankar[2]

[1]Associate Professor, Department of Computer Science and Applications, SCSVMV [Deemed to be University], Enathur, Kanchipuram

[2]Student, II-YEAR MCA, Department of Computer Science and Applications, SCSVMV [Deemed to be University], Enathur, Kanchipuram

Publication Date: 2026/05/30

Abstract: With the rapid advancement of digital technologies and the increasing demand for personalized digital content, the need to convert handwritten text into reusable and scalable digital formats has become highly significant. In particular, regional languages like Tamil, which possess rich script structures and cultural importance, require efficient digitization techniques to preserve and promote their usage in modern digital environments. This project proposes a comprehensive system designed to convert handwritten Tamil text into a fully functional digital font. The system allows users to write Tamil text manually on paper, scan or capture the document as an image, and upload it into the application. The uploaded handwritten document undergoes a series of image preprocessing steps, including noise removal, grayscale conversion, binarization, normalization, and segmentation, to enhance the quality and prepare the data for accurate recognition. Following preprocessing, the system employs character recognition techniques to identify individual Tamil characters. Feature extraction methods are applied to capture the unique structural patterns of each handwritten character. These features are then analyzed using machine learning or pattern recognition algorithms to accurately classify and map each character to its corresponding Tamil Unicode representation. Once the characters are recognized, the system proceeds to generate scalable digital glyphs. Each glyph is carefully designed to retain the stylistic characteristics of the user's handwriting, ensuring personalization. These glyphs are then assembled into a complete font file (such as TTF or OTF format), which can be installed and used across various applications. The system also provides an interactive user interface that enables users to preview the generated font in real-time. Users can test the font with sample text, make adjustments if necessary, and download the finalized font for use in documents, graphic design, publishing, and other digital platforms. This project effectively bridges the gap between traditional handwritten Tamil script and modern digital typography. It not only facilitates the preservation of individual handwriting styles but also promotes the digital adoption of Tamil language content. Furthermore, the system contributes to the broader field of document digitization and font generation, offering potential applications in education, digital archiving, and creative design industries. the proposed solution enhances user creativity and personalization while supporting the cultural preservation of Tamil handwriting. It demonstrates how the integration of image processing, character recognition, and font generation techniques can transform handwritten documents into valuable digital assets.

How to Cite: Dr. S. Prakasam; S. Yuvan Shankar (2026) A System for Recognition and Digitalization of Tamil Handwritten Document. *International Journal of Innovative Science and Research Technology*, 11(5), 2508-2514. https://doi.org/10.38124/ijisrt/26may1384

I. INTRODUCTION

This project focuses on developing a system for the recognition and digitization of handwritten Tamil documents, with the added capability of generating custom Tamil fonts. The system aims to allow users to write Tamil text on paper, upload the handwritten document, and automatically convert it into a usable digital font that retains the user's writing style. This not only enhances user creativity but also provides a meaningful way to preserve personal handwriting digitally. The proposed system utilizes image processing techniques to prepare the handwritten input for recognition. These techniques include noise removal, image enhancement, binarization, segmentation, and normalization. By improving the quality of the input image, the system ensures higher accuracy in character detection and recognition. Once the preprocessing stage is complete, character recognition methods are applied to identify individual Tamil characters. This involves feature extraction, pattern matching, and classification using suitable algorithms. After successful recognition, the system maps each identified character to its corresponding Tamil Unicode value. Unicode mapping is

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1384

essential for ensuring compatibility across different platforms and devices. Following this, the system generates digital glyphs that visually represent each character. These glyphs are then compiled into a standard font format such as TrueType Font (TTF) or OpenType Font (OTF), allowing users to install and use the generated font in various applications. One of the key highlights of this project is the personalization aspect. Unlike conventional fonts, the generated font preserves the unique characteristics of the user's handwriting, enabling customized digital communication. Additionally, the system provides a user friendly interface where users can preview the generated font, test it with sample text, and download it بسهولة for further use. The importance of this project extends beyond individual usage. It contributes to the preservation of handwritten Tamil scripts, supports digital content creation in regional languages, and opens new opportunities in fields such as education, graphic design, and digital publishing. For instance, teachers can create personalized study materials, designers can use custom fonts for creative projects, and researchers can digitize historical handwritten documents. Moreover, this system demonstrates the integration of multiple technologies, including image processing, pattern recognition, machine learning, and font generation. It highlights how interdisciplinary approaches can be used to solve real-world problems effectively. The project also aligns with the broader goal of promoting regional language computing and enhancing accessibility to digital tools for diverse user groups.

- *Problem Statement*

The rapid growth of digital communication and content creation has increased the need for converting handwritten documents into digital formats. Although many Optical Character Recognition (OCR) systems are available for recognizing printed and handwritten text, most existing solutions are mainly designed for global languages such as English and provide limited support for regional languages like Tamil. Tamil script contains complex character structures, curves, and combinations, making handwritten character recognition a challenging task.

In the current digital environment, users can access numerous pre-designed Tamil fonts; however, these fonts do not preserve the uniqueness of an individual's handwriting style. Existing systems mainly focus on converting handwritten text into editable digital text and lack the capability to generate personalized font files from handwritten Tamil characters. As a result, users who wish to use their own handwriting in digital documents, creative designs, or educational materials often depend on manual font creation methods, which are time-consuming and require technical expertise.

Moreover, this system demonstrates the integration of multiple technologies, including image processing, pattern recognition, machine learning, and font generation. It highlights how interdisciplinary approaches can be used to solve real-world problems effectively. The project also aligns with the broader goal of promoting regional language computing and enhancing accessibility to digital tools for diverse user groups.

II. EXISTING SYSTEM DRAWBACKS

In the current scenario, the conversion of handwritten text into digital format is primarily handled by Optical Character Recognition (OCR) systems. These systems are widely used to scan and convert printed documents into editable digital text. Some advanced OCR tools also attempt to recognize handwritten text; however, most of them are designed for global languages and are less effective when dealing with complex regional scripts like Tamil. Existing Tamil OCR systems mainly focus on recognizing handwritten or printed Tamil text and converting it into plain digital text. They do not support the generation of personalized fonts based on individual handwriting styles. Additionally, most available digital Tamil fonts are pre-designed and standardized, lacking the ability to reflect a user's unique writing characteristics. Users who wish to use their own handwriting in digital form often have to rely on manual methods or generic font creation tools, which are time-consuming and require technical expertise. Furthermore, many existing solutions require high-quality scanned inputs and controlled writing conditions to achieve acceptable accuracy. Variations in handwriting styles, stroke thickness, spacing, and alignment often reduce the performance of these systems. As a result, achieving consistent and accurate recognition of handwritten Tamil characters remains a challenge.

III. PROPOSED SYSTEM

Once the recognition process is completed, the system proceeds to generate digital glyphs that visually represent each recognized character. These glyphs are carefully designed to retain the original handwriting style of the user. The glyphs are then compiled into a standard font file format such as TrueType (TTF) or OpenType (OTF), which can be easily installed and used in various software environments. The system also provides an interactive interface that allows users to preview the generated font in real-time. Users can test the font with sample text, verify its accuracy, and make necessary adjustments if required. Finally, the font can be downloaded and used in applications such as word processing, graphic design, and digital publishing. The proposed system not only enhances personalization but also contributes to the preservation of Tamil handwriting styles in digital form. It has wide applications in education, creative design, document digitization, and cultural archiving. Furthermore, the system can be extended in the future to support multiple languages, improve recognition accuracy using advanced machine learning models, and integrate cloud-based services for storage and processing.

IV. SYSTEM ARCHITECTURE

The system design describes the overall structure, architecture, and workflow of the proposed handwritten Tamil text to font generation system. It explains how different components of the system interact with each other to achieve

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1384

the desired functionality. The design follows a modular and layered approach to ensure flexibility, scalability, and ease of maintenance.

➢ *Frontend Layer*

The frontend layer of the handwritten Tamil font generation system is responsible for providing an interactive and user-friendly interface through which users can interact with the application. The frontend is developed using modern web technologies such as HTML, CSS, JavaScript, and Streamlit/React.js to ensure a responsive and visually appealing user experience. This layer allows users to download handwriting templates, upload handwritten Tamil text images, preview uploaded files, generate fonts, and download the final font file. Responsive design techniques are used to support different devices such as desktops, tablets, and mobile phones. The frontend communicates with the backend services through RESTful API calls and displays the processing status, recognition results, and font preview dynamically. Additionally, the interface provides validation mechanisms to ensure that users upload valid image formats such as JPG or PNG for accurate processing.

➢ *Backend Layer*

The backend layer acts as the core processing unit of the system and manages all business logic and server-side operations. The backend can be developed using Python Flask/Django or Node.js with Express.js. It follows a modular architecture where separate modules handle image preprocessing, segmentation, feature extraction, character recognition, Unicode mapping, and font generation. The backend receives uploaded handwritten images from the frontend and processes them using image processing libraries such as OpenCV and PIL. Machine learning models are integrated into the backend to recognize Tamil handwritten characters accurately. The backend also handles file management, API routing, authentication (if required), and communication with the database. Font generation tools such as FontForge are integrated into the backend to create scalable TrueType (TTF) or OpenType (OTF) font files. Error handling and standardized API responses ensure smooth communication between the frontend and backend layers.

➢ *Database Layer*

The database layer is responsible for storing and managing user information, uploaded handwritten samples, processed character data, generated font files, and application logs. Databases such as MongoDB or MySQL can be used depending on the project requirements. MongoDB is suitable because it supports flexible document-based storage for images, Unicode mappings, and generated font metadata. The database stores user profiles, uploaded templates, recognized characters, generated glyph information, and downloadable font records. Proper indexing and schema design help improve retrieval speed and system performance. The database layer also supports scalability by allowing multiple users to generate and store personalized fonts simultaneously. Additionally, cloud database integration can be implemented for secure storage and remote access.

➢ *AI & Media Services*

The AI layer plays a major role in recognizing handwritten Tamil characters and improving the accuracy of the system. Machine learning and deep learning techniques are used to identify handwritten Tamil characters from segmented images. Convolutional Neural Networks (CNNs) are commonly used because they are highly effective in image-based character recognition tasks. TensorFlow, Keras, or Scikit-learn frameworks can be used to train and deploy the recognition models. The AI model analyzes extracted features such as curves, edges, loops, and stroke patterns to classify Tamil characters accurately.

V. METHODOLOGY

The methodology of the OCR Paddle-based handwritten Tamil text recognition system explains the complete process used to extract and recognize handwritten Tamil text from images using Optical Character Recognition (OCR) techniques. The system integrates image preprocessing, segmentation, deep learning-based text detection, text recognition, and Unicode conversion to achieve accurate handwritten Tamil text extraction. The overall methodology is divided into several stages, where each stage processes the output from the previous stage to improve recognition accuracy and system performance.

➢ *Data Collection and Image Acquisition*

cwords, or sentences on paper and upload the scanned image or captured photograph into the system. The uploaded images may contain handwritten notes, forms, or documents. The system verifies the uploaded files to ensure they are in supported formats such as JPG, JPEG, or PNG and satisfy the required image quality standards.

➢ *Image Preprocessing*

After image acquisition, preprocessing techniques are applied to enhance the quality of the handwritten image before OCR recognition. Since handwritten images may contain noise, shadows, blur, or uneven lighting conditions, preprocessing helps improve OCR accuracy. The preprocessing operations include:

- Grayscale Conversion
- Noise Removal
- Image Binarization
- Contrast Enhancement
- Image Resizing

➢ *Text Detection and Segmentation*

In this stage, the OCR Paddle system detects the handwritten text regions from the uploaded image. The image is segmented into lines, words, and individual text regions for accurate recognition. PaddleOCR uses deep learning-based text detection algorithms to identify Tamil handwritten text areas effectively. Segmentation helps isolate text portions while reducing background interference and overlapping character issues.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1384

➢ *Feature Extraction*

After segmentation, important visual features are extracted from the handwritten Tamil text. Feature extraction helps the system identify the structural patterns and writing style of Tamil characters. The extracted features include:

- Character edges and curves
- Stroke patterns
- Loops and intersections
- Character shapes and writing structures

These features help the OCR model differentiate between similar Tamil characters and improve recognition.

➢ *Handwritten Text Recognition Using PaddleOCR*

The extracted features are processed using the PaddleOCR framework for handwritten Tamil text recognition. PaddleOCR uses deep learning models such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) to recognize text from images with high accuracy. The OCR model is trained using Tamil handwritten datasets to improve character prediction and text extraction performance.

The recognition module converts handwritten Tamil text images into editable digital text while preserving the original textual information.

➢ *Unicode Conversion and Text Mapping*

After recognition, the identified Tamil characters are mapped to their corresponding Unicode values. Unicode conversion ensures that the recognized Tamil text can be displayed correctly across various operating systems, software applications, and digital platforms. This stage transforms the OCR output into standardized machine-readable Tamil text.

➢ *Text Post-Processing*

The recognized Tamil text undergoes post-processing to improve accuracy and readability. The system corrects minor recognition errors using language-based validation and Tamil text formatting techniques. Spell correction and character refinement methods help generate more accurate OCR output.

➢ *Digital Text Generation and Storage*

After successful recognition and correction, the extracted Tamil text is converted into editable digital content. The recognized text is stored in the system database or exported into standard document formats such as:

- TXT
- PDF
- DOCX

This stage enables users to reuse, edit, and manage the extracted Tamil text efficiently.

➢ *OCR Output Preview and Download*

Finally, the system provides a preview feature that allows users to verify the recognized Tamil text before downloading. Users can compare the OCR-generated text with the original handwritten document and validate recognition accuracy. Once satisfied, users can download the extracted text file for further use in document editing, digital archiving, publishing, and communication purposes**.**

VI. SCREENSHOT

➢ *Tamil Deep OCR:*

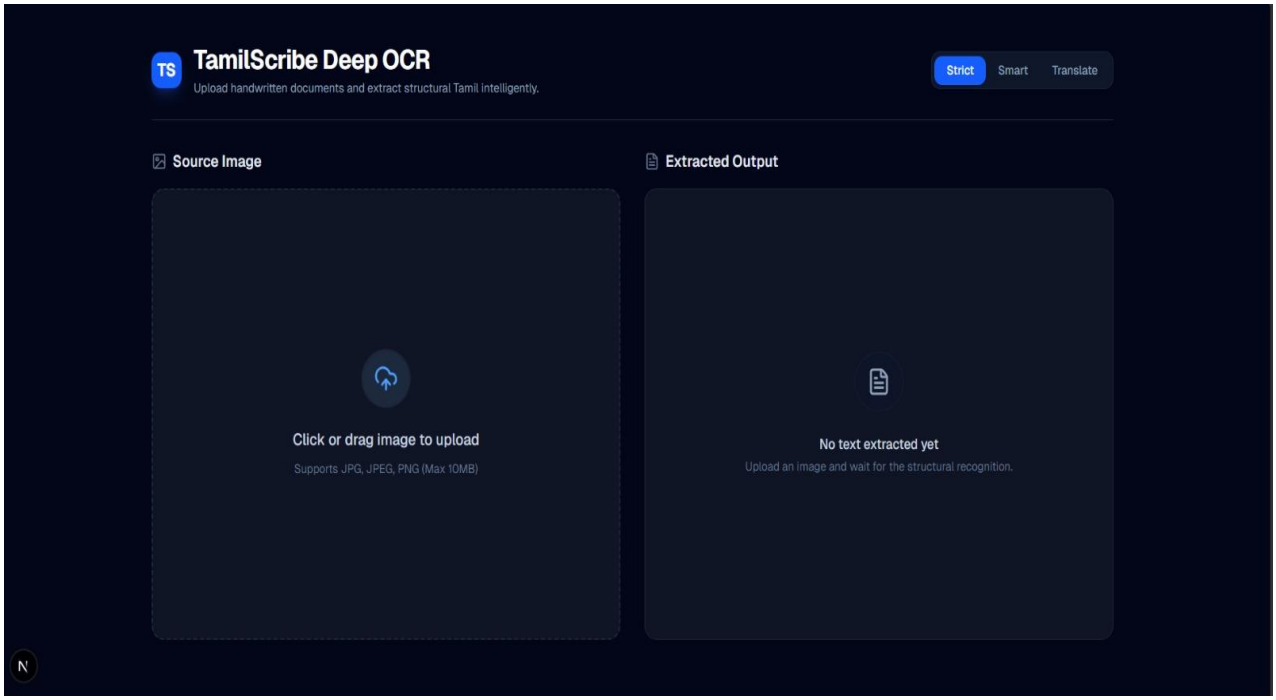

Fig 1 Tamil Deep OCR

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1384

➢ *Extracted Output:*

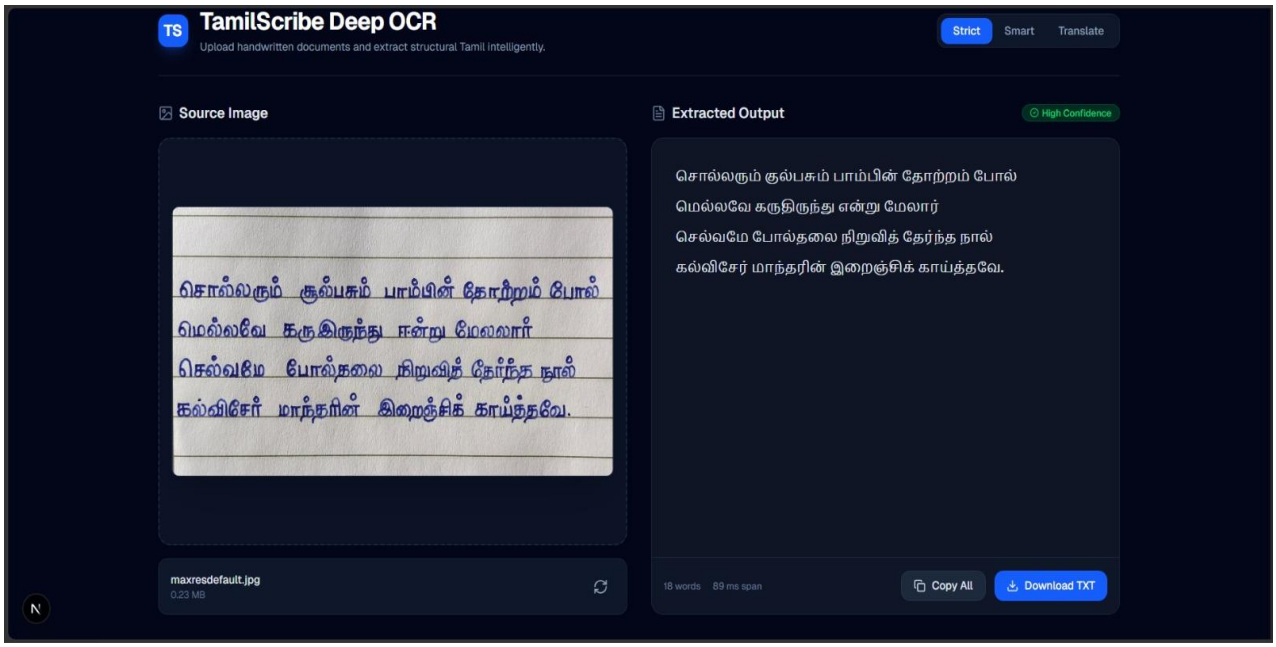

Fig 2 Extracted Output

VII. TECHNOLOGY STACK

The complete technology stack is summarised below. All dependencies are open-source or available under free tiers appropriate for academic and early-stage commercial use.

Category	Technology	Purpose
Component	**Technology Used**	**Purpose**
Programming Language	Python	Core project development
OCR Framework	PaddleOCR	Handwritten Tamil text recognition
Deep Learning Framework	PaddlePaddle	OCR model training and processing
Image Processing	OpenCV	Image preprocessing and segmentation
Font Generation	FontForge	Creating TTF and OTF font files
Frontend	HTML, CSS, JavaScript	User interface development
Backend	Flask / Django	Web application backend
Database	MySQL / SQLite	Data storage and management

VIII. AI-POWERED MODULES

The handwritten Tamil font generation system integrates several AI-powered modules to automate handwritten text recognition and digital font creation. These modules improve the accuracy, efficiency, and scalability of the system.

➢ *Handwritten Text Detection Module*

This module identifies and detects handwritten Tamil text regions from uploaded images using PaddleOCR. It separates the text area from the background for accurate processing.

➢ *Character Recognition Module*

The recognition module uses deep learning algorithms such as Convolutional Neural Networks (CNNs) to recognize handwritten Tamil characters from segmented images with high accuracy.

➢ *Image Preprocessing Module*

This module enhances image quality by applying grayscale conversion, noise removal, binarization, and contrast enhancement techniques before OCR processing.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1384

- *Unicode Mapping Module*

The recognized Tamil characters are automatically mapped into standard Unicode values to ensure compatibility across digital platforms and applications.

- *Glyph Generation Module*

This module converts recognized Tamil characters into vector-based glyphs while preserving the original handwriting style of the user.

- *Font Generation Module*

The font generation module compiles all generated glyphs into standard digital font formats such as TTF and OTF using AI-assisted font creation techniques.

- *Error Correction Module*

This module performs post-processing and text validation to reduce OCR recognition errors and improve the quality of the generated Tamil text and fonts.

IX. RESULT AND DISCUSSION

The handwritten Tamil font generation system using PaddleOCR successfully recognizes handwritten Tamil characters and converts them into editable digital text and personalized font files. The system effectively performs image preprocessing, character detection, OCR recognition, Unicode mapping, and font generation with good accuracy.

The implemented system was tested using multiple handwritten Tamil samples collected from different users. The OCR module accurately detected and recognized Tamil characters even when variations in handwriting styles, stroke thickness, and writing patterns were present. Image preprocessing techniques such as noise removal and binarization improved recognition performance and reduced errors during text extraction.

The generated font preserved the original handwriting style of the user while creating scalable digital font files in TTF and OTF formats. The live preview module allowed users to verify the generated font before downloading. The system also demonstrated compatibility across different applications and operating systems through Unicode-based Tamil text mapping.

X. CONCLUSION

The handwritten Tamil font generation system using PaddleOCR successfully converts handwritten Tamil text into editable digital text and personalized font files. The system integrates image preprocessing, OCR-based handwritten text recognition, Unicode mapping, glyph creation, and font generation techniques to achieve accurate and efficient results.

By using deep learning and AI-powered OCR technology, the system can recognize different handwriting styles with improved accuracy and generate scalable digital fonts in standard formats such as TTF and OTF. The project reduces manual effort in font creation and helps preserve individual handwriting styles in digital form.

Overall, the system provides an effective solution for handwritten Tamil text digitization and font generation, making it useful for document processing, publishing, digital communication, and personalized typography applications.

FUTURE SCOPE

The handwritten Tamil font generation system using PaddleOCR can be further enhanced by improving OCR accuracy for complex handwriting styles and low-quality images. Future development may include support for additional regional languages and multilingual font generation. Advanced AI models can be integrated to improve character recognition speed and reduce recognition errors.

The system can also be enhanced with cloud-based storage and real-time font generation features for better accessibility and scalability. Mobile application support can be developed to allow users to capture handwritten text directly using smartphone cameras. Additional features such as automatic spell correction, handwriting style customization, and AI-based font beautification can further improve the overall user experience and font quality.

REFERENCES

[1]. PaddleOCR, “PaddleOCR: Multilingual OCR System,” GitHub Repository, 2025.
[2]. PaddlePaddle, “PaddlePaddle Deep Learning Platform,” Official Documentation, 2025.
[3]. OpenCV, “Open Source Computer Vision Library,” Official Documentation, 2025.
[4]. FontForge, “FontForge Font Creation and Editing Tool,” Official Documentation, 2025.
[5]. TensorFlow, “TensorFlow for OCR and Deep Learning Applications,” Google Documentation, 2025.
[6]. Python Software Foundation, “Python Programming Language Documentation,” 2025.
[7]. Digital Image Processing, Gonzalez, R. C., and Woods, R. E., *Digital Image Processing*, Pearson Education, 4th Edition, 2018.
[8]. Pattern Recognition and Machine Learning, Bishop, C. M., *Pattern Recognition and Machine Learning*, Springer, 2006.
[9]. Optical Character Recognition, Research papers on OCR systems and handwritten text recognition, IEEE Xplore Digital Library, 2024.
[10]. Machine Learning, CNN-based handwritten Tamil character recognition research articles, Springer Journals, 2024.
[11]. Deep Learning, Deep learning approaches for handwritten text recognition, Elsevier Publications, 2023.
[12]. Image Processing, Research articles on preprocessing and segmentation techniques for OCR applications, IEEE Journals, 2023.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1384

[13]. Computer Vision, Szeliski, R., *Computer Vision: Algorithms and Applications*, Springer, 2022.
[14]. Artificial Intelligence, AI-based font generation and handwriting analysis research papers, ACM Digital Library, 2024.