

# Embedded Smart Pen for Real-Time Word Recognition and Meaning Search

<sup>1</sup>Shrunkhal Moreshwar Supale, <sup>2</sup>Sahiba Kamal Siddiqui, <sup>3</sup>Unnati Nitin Shrivastava, <sup>4</sup>Isha Gautam Sontakke, <sup>5</sup>Pushpa Tandekar

<sup>1,2,3,4</sup>Student, Computer Science and Engineering, Shri Sai College of Engineering and Technology, Bhadrawati, Chandrapur, India

<sup>5</sup>Professor, Computer Science and Engineering, Shri Sai College of Engineering and Technology, Bhadrawati, Chandrapur, India

**Abstract** - This paper presents a smart embedded pen system designed for real-time text recognition and semantic interpretation. The proposed system uses an embedded camera module to capture textual content, which is then transmitted via a wireless interface to a processing server. Optical Character Recognition (OCR) is performed using Tesseract OCR to extract text from the captured images. To enhance accuracy and usability, a custom preprocessing pipeline is introduced, including error correction using string similarity measures, sentence importance ranking based on frequency analysis, and rule-based detection of basic literary devices. This preprocessing layer improves the quality of the input before further semantic analysis.

The processed text is then passed to a language model for generating simplified meanings, contextual explanations, and key insights. The system follows a distributed architecture where the embedded device is responsible for data acquisition and user interaction, while computationally intensive tasks are handled by the server. This approach ensures efficient performance despite hardware constraints. Experimental results demonstrate that the system can effectively extract and interpret textual information in near real-time, making it suitable for educational assistance and accessibility applications. Future enhancements include improving OCR robustness, expanding linguistic analysis capabilities, and integrating more advanced on-device processing.

**Keywords:** Text Recognition, Optical Character Recognition (OCR), Semantic Analysis, Real-Time Processing, Assistive Technology, Natural Language Processing (NLP), Edge Computing, Embedded Vision.

## I. INTRODUCTION

In the understanding of text, the OCR, i.e. optical character recognition and NLP, i.e. Natural Language Processing, have a crucial role in terms of converting the text into meaningful sentences and information.

OCR, such as Tesseract OCR provide reliable mechanisms for extracting the meaning out of the images, while advanced Large language models facilitate the semantic interpretation, contextual analysis, and also natural language generation.

Having such technological advancements, the existing solutions are only limited to the software-based platforms. So, there is a lack of integration with embedded hardware.

In the process of acquiring text and specifying its interpretation, there could be many challenges:

- Existing OCR systems primarily focus on **text extraction** and do not provide higher-level **semantic understanding** of the content.
- Language models capable of contextual interpretation require **high computational resources**, making them unsuitable for direct deployment on microcontroller-based systems.
- Embedded platforms such as camera-enabled microcontrollers are constrained by **limited memory, processing power, and energy efficiency**, restricting on-device execution of complex algorithms.
- Current implementations are mostly depend on mobile or any other physical devices; there is no availability of embedded user interaction.
- Raw OCR outputs often contain **noise and inaccuracies**, necessitating the use of **preprocessing algorithms** to improve text quality before further analysis.
- Lack of systems that integrate text acquisition, preprocessing, and semantic interpretation within a portable fra-----991272mework.
- A distributed approach combining **edge devices for sensing** and **server-based systems for computation** is required to balance performance and feasibility.

To address these challenges, this paper proposes a smart embedded pen system based on distributed edge-server architecture. The system employs a camera-enabled embedded device i.e camera module, for real-time image acquisition, which transmits captured data to a server for processing. The

server performs OCR, followed by a custom preprocessing pipeline incorporating error correction using edit-distance methods, frequency-based sentence ranking, and rule-based detection of linguistic patterns. Subsequently, a language model is utilized to generate simplified meanings, contextual explanations, and key insights from the extracted text. This approach enables efficient utilization of computational resources while maintaining portability and real-time interaction, thereby providing a practical solution for intelligent text understanding in embedded environments.

### 1.1 Bridging the gap between Reading and understanding

A significant challenge faced by readers, particularly students and non-expert audiences, is the inability to fully comprehend the underlying meaning, tone, and intent of textual content. This issue is especially prominent in literary texts, where authors frequently employ figurative language, such as metaphors, similes, and symbolic expressions. Although readers may be able to decode the literal words, extracting the intended message often requires additional cognitive effort, prior knowledge, or external assistance.

Existing digital solutions primarily focus on surface-level processing tasks, including text recognition, translation, or dictionary-based definitions. These approaches fail to address deeper semantic interpretation, such as identifying contextual meaning, summarizing key ideas, or explaining literary constructs in an intuitive manner. Furthermore, the reliance on smartphones or computers disrupts the natural reading experience and introduces friction in real-time comprehension. This highlights the need for an integrated system that can assist users directly during reading, enabling seamless access to contextual explanations without interrupting the interaction with physical text.

Table 1: Existing systems vs. proposed system

Feature	OCR Systems	NLP/LLM Systems	Mobile Apps	Proposed Smart Pen System
Text Extraction	Yes	No	Yes	Yes
Semantic Understanding	No	Yes	Limited	Yes
Real-Time Processing	Limited	Yes	Yes	Yes
Embedded Hardware Integration	No	No	No	Yes

Feature	OCR Systems	NLP/LLM Systems	Mobile Apps	Proposed Smart Pen System
Portability	No	No	Yes	Yes
Contextual Explanation	No	Yes	Limited	Yes

### 1.2 Enabling Real-Time Context aware text interpretation through embedded systems

To address the limitations of existing approaches, this work proposes the design and implementation of a smart embedded pen system capable of real-time text recognition and semantic interpretation. The objective is to develop a portable, user-friendly interface that allows users to capture textual content directly from physical sources and obtain meaningful explanations instantly.

The proposed system adopts a distributed edge-server architecture to overcome the computational constraints of embedded hardware. A camera-enabled embedded device is used for real-time image acquisition, while processing tasks are offloaded to a server. The system performs text extraction using OCR techniques, followed by a custom preprocessing pipeline that includes error correction using string similarity measures, sentence importance ranking based on frequency analysis, and rule-based detection of basic linguistic constructs. The refined text is then processed by a language model to generate simplified meanings, contextual interpretations, and key insights.

By integrating embedded sensing with advanced AI-driven semantic analysis, the system aims to transform passive reading into an interactive and assisted experience. This approach not only enhances comprehension but also demonstrates the potential of combining edge devices with server-based intelligence to create scalable and efficient real-time text understanding systems.

## II. METHODOLOGY

The proposed system is designed using a distributed edge-server architecture that separates data acquisition from computational processing to overcome the limitations of embedded hardware. The architecture consists of three primary layers: (i) the Embedded Acquisition Layer, (ii) the Communication Layer, and (iii) the Processing and Interpretation Layer. This modular design ensures scalability, real-time performance, and efficient utilization of system resource.

## 1. Embedded Acquisition Layer

This layer is responsible for real-time data capture and user interaction. It is implemented using a camera-enabled microcontroller, specifically the ESP32-CAM. The device captures images of textual content upon user input (e.g., button press). The captured image is temporarily stored in the device buffer and encoded in JPEG format to reduce transmission overhead.

In addition to image capture, this layer includes a basic output interface (e.g., OLED display) to provide user feedback such as system status (“Capturing”, “Processing”, “Completed”). The embedded device operates as an edge node, focusing on sensing and communication rather than computational processing.

## 2. Communication Layer

The communication layer facilitates data transfer between the embedded device and the processing server. A wireless communication protocol (WiFi) is employed, using an HTTP-based client-server model. The embedded device acts as an HTTP client, sending captured images to the server through POST requests.

This layer ensures reliable and low-latency transmission of image data. The use of standard HTTP protocols enables seamless integration with backend services and simplifies implementation. The communication layer is critical for maintaining real-time performance and ensuring synchronization between the hardware and processing components.

## 3. Processing and Interpretation Layer

The processing layer is implemented on a server or high-performance computing system and is responsible for executing computationally intensive tasks. It consists of the following sub modules:

### 1. Optical Character Recognition (OCR):

The received image is processed using Tesseract OCR to extract textual content. Preprocessing steps such as grayscale conversion and noise filtering are applied to improve recognition accuracy.

### 2. Text Preprocessing Module:

The extracted text is refined using custom algorithms, including:

- Edit-distance-based error correction.
- Frequency-based sentence ranking.
- Rule-based detection of linguistic patterns (e.g., similes, metaphors)

### 3. Semantic Analysis Module:

The processed text is passed to a language model, such as GPT-4, to generate simplified meanings, contextual interpretations, and key insights.

## 4. Output Layer

The output generated by the processing layer is delivered to the user through a display interface. In the current implementation, results are presented on a connected system interface (e.g., web application or console), while the embedded device provides minimal feedback via an OLED display. This separation ensures that detailed information is displayed on a suitable interface without overloading the embedded hardware.

## 5. System Workflow

The overall system operation follows a sequential pipeline:

- User initiates image capture using the embedded device
- The embedded device captures and encodes the image
- The image is transmitted to the server via HTTP over WiFi
- The server performs OCR to extract text
- The extracted text is refined using preprocessing algorithms
- A language model generates semantic interpretation
- The final output is displayed to the user

## 6. Architectural Advantages

- **Scalability:** Computational tasks can be upgraded independently of hardware
- **Efficiency:** Heavy processing is offloaded from resource-constrained devices
- **Real-Time Operation:** Low-latency communication ensures quick response
- **Modularity:** Each layer can be modified or improved without affecting the entire system

This architecture effectively bridges the gap between embedded hardware limitations and the requirements of advanced AI-based text processing, enabling a practical and efficient solution for real-time text recognition and semantic interpretation.

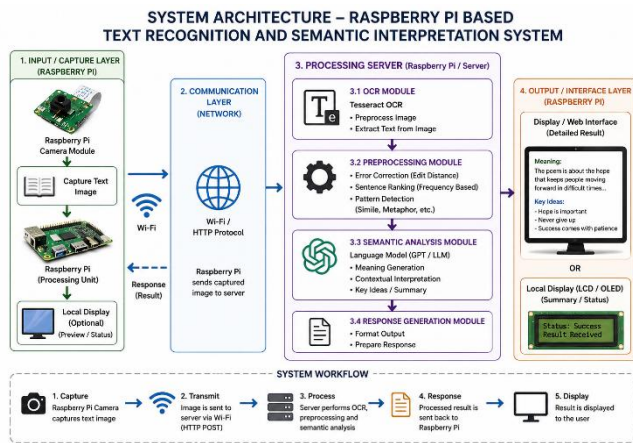


Figure 1: System architecture

### III. RESULTS AND DISCUSSIONS

The proposed system was evaluated using multiple textual inputs captured from printed sources. The system successfully performed end-to-end processing, including image acquisition, text extraction, preprocessing, and semantic interpretation. The OCR module, implemented using Tesseract OCR, was able to extract text with reasonable accuracy under standard lighting conditions. However, minor errors were observed in cases of low contrast and complex fonts.

The preprocessing stage significantly improved the quality of the extracted text. The edit-distance-based error correction reduced OCR inaccuracies, while the sentence ranking mechanism effectively identified relevant content. Rule-based pattern detection provided basic identification of linguistic constructs such as similes and metaphors, although its effectiveness was limited to predefined patterns.

The semantic analysis module generated meaningful interpretations of the processed text, including simplified explanations and key ideas. The system demonstrated the ability to provide context-aware insights, particularly for literary sentences. Overall, the distributed architecture enabled efficient real-time performance, with processing delays primarily dependent on network latency and server response time.

Despite its effectiveness, the system has certain limitations. The accuracy of the output is dependent on the quality of OCR, and the rule-based approach to linguistic analysis lacks generalization. Future improvements can include advanced NLP techniques and improved image preprocessing to enhance robustness and accuracy.

### IV. CONCLUSION

This paper presented a smart embedded pen system for real-time text recognition and semantic interpretation using a

distributed edge-server architecture. The proposed system integrates a camera-enabled embedded device with server-side processing to overcome the computational limitations of microcontroller-based platforms. Textual data is captured through the embedded interface and transmitted to a server, where Optical Character Recognition (OCR) using Tesseract OCR is applied, followed by a custom preprocessing pipeline incorporating error correction, sentence ranking, and rule-based linguistic analysis. The refined text is then processed using a language model such as GPT-4 to generate simplified meanings and contextual interpretations.

The experimental results demonstrate that the system is capable of extracting and interpreting textual information with reasonable accuracy under standard conditions. The preprocessing stage enhances the quality of OCR output, and the semantic analysis module provides meaningful insights that improve user comprehension, particularly in literature-based contexts. The use of a distributed architecture ensures efficient performance while maintaining portability and real-time interaction.

However, the system is subject to certain limitations, including dependency on OCR accuracy, sensitivity to image quality, and limited capability of rule-based linguistic analysis. Future work may focus on improving robustness through advanced image preprocessing, incorporating machine learning-based text correction methods, and enhancing semantic analysis using more sophisticated models. Overall, the proposed system demonstrates the feasibility of combining embedded systems with AI-driven processing to create an effective and practical solution for real-time text understanding.

### ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to the faculty members and project guides for their continuous support, valuable guidance, and constructive feedback throughout the development of this work. Their insights played a crucial role in shaping both the technical and research aspects of this project.

The authors also acknowledge the availability of open-source tools and frameworks, including Tesseract OCR and modern language models such as GPT-4, which significantly contributed to the implementation and evaluation of the proposed system.

Finally, the authors extend their appreciation to their institution for providing the necessary resources and infrastructure to successfully complete this research work.

**REFERENCES**

- [1] Levenshtein Distance V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," Soviet Physics Doklady, 1966.
- [2] P. Norvig, "How to Write a Spelling Corrector," 2007.
- [3] TF-IDF, G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," Information Processing & Management, 1988.
- [4] H. P. Luhn, "The Automatic Creation of Literature Abstracts," IBM Journal, 1958.
- [5] D. Jurafsky and J. Martin, Speech and Language Processing.
- [6] C. Manning & H. Schütze, Foundations of Statistical NLP.
- [7] Rule-based NLP. E. Charniak, Statistical Language Learning, MIT Press.
- [8] Tesseract OCR paper.

**Citation of this Article:**

Shrunkhal Moreshwar Supale, Sahiba Kamal Siddiqui, Unnati Nitin Shrivastava, Isha Gautam Sontakke, & Pushpa Tandekar. (2026). Embedded Smart Pen for Real-Time Word Recognition and Meaning Search. *International Research Journal of Innovations in Engineering and Technology - IRJIET*, 10(5), 1-5. Article DOI <https://doi.org/10.47001/IRJIET/2026.105001>

\*\*\*\*\*