International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** IX **Month of publication:** September 2025

DOI: https://doi.org/10.22214/ijraset.2025.74251

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IX Sep 2025- Available at www.ijraset.com

Robust OCR Pipeline for Digit Display Recognition Using TrOCR, YOLOv8, and Multi-Layered Fallbacks

Deon Menezes[1], Sajal Patel[2], Abdul Shaikh[3], Nishaad Parikh[4]
[1]*CEO/Research Engineer,* [2, 3, 4]*AI/ML, Intern Virelity Pvt. Ltd*

Abstract: The variable lighting conditions, segmentation complexity, and inconsistent formatting of digital displays, like those found on utility meters and fuel pumps, present ongoing challenges for optical character recognition (OCR) systems. For ROI detection, we suggest a reliable, multi-model OCR pipeline that combines a refined TrOCR model with YOLOv8 and is enhanced with fallback mechanisms utilizing Tesseract and EasyOCR. Numerical output integrity is improved by a custom decimal correction procedure. After post-processing, our suggested approach outperforms standalone OCR engines by achieving a 97% success rate on real-world digit displays. We examine failure cases from previous CNN-based segmentation attempts, present comparative performance analysis, and describe upcoming work for wider deployment.
Index Terms: OCR, TrOCR, YOLOv8, digit display recognition, Tesseract, EasyOCR, decimal correction

I. INTRODUCTION

Although optical character recognition (OCR) technologies have made great strides, low contrast, segment overlap, and misaligned digits still pose problems for realworld digit-based displays like fuel pumps or digital meters. These structured formats are frequently misinterpreted by conventional OCR engines such as Tesseract, which calls for sophisticated pipelines that integrate intelligent preprocessing with deep learning-based OCR. The need for dependable digit recognition in a variety of settings is further highlighted by the growth of IoT devices and automated systems.

Using a refined TrOCR model, YOLOv8 for ROI detection, and fallback OCR techniques (Tesseract, EasyOCR) with decimal correction, this paper proposes an end-toend OCR pipeline designed for structured digit displays. With results compiled as of July 07, 2025, we outline the system design, performance enhancements, and errorhandling mechanisms. We also assess development failure points. The suggested method seeks to solve real-world deployment issues in consumer and industrial applications.

II. RELATED WORK

Conventional OCR systems, like Tesseract [2], are good at digitizing structured documents, but they have trouble with scene text in unpredictable environments. While object detectors like YOLO [3] have been used for ROI detection to guide OCR focus, transformer-based models like TrOCR [1] have demonstrated superior generalization for scene text recognition.

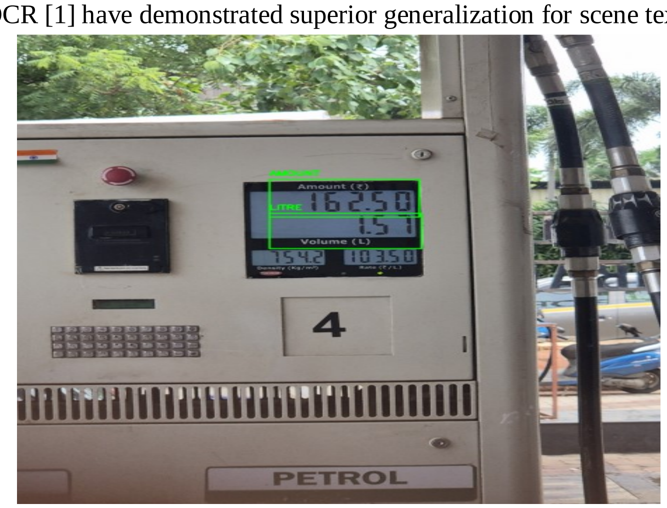


Fig. 1. Example of ROI detection using YOLOv8 on a fuel pump display

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IX Sep 2025- Available at www.ijraset.com

Deep learning advances in OCR, including attention mechanisms and multimodal approaches, are highlighted in recent works like [5] and [6]. Our hybrid approach was motivated by the lack of research on integrating multiple OCR systems with domain-specific error handling.

III. METHODOLOGY

A. System Overview

Our pipeline comprises four stages:

1) ROI Detection: YOLOv8 identifies the digit display region in the input image, leveraging its high recall for object detection.
2) OCR Inference: Fine-tuned TrOCR processes the ROI; if it fails or produces malformed outputs, fallback OCRs (Tesseract, EasyOCR) are invoked.
3) Error Handling: A decimal correction routine ensures numeric output integrity by addressing common OCR misinterpretations.
4) Consensus Selection: A majority-voting mechanism selects the final result from multiple OCR outputs, enhancing reliability.

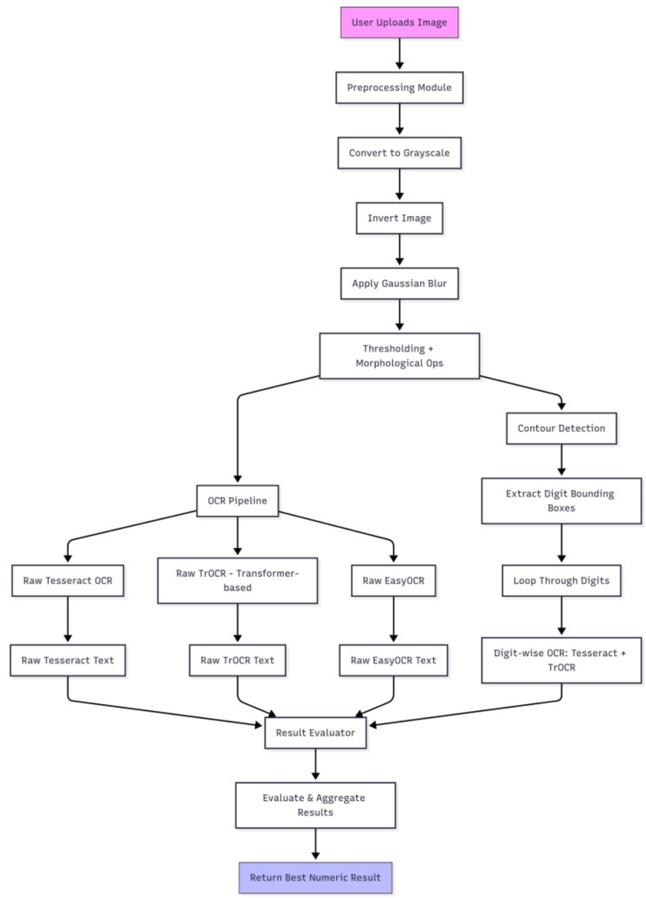


Fig. 2. System architecture of the proposed OCR pipeline

B. TrOCR Fine-Tuning

Using a custom dataset of 2,500 digit display images, including seven-segment and printed displays, we optimized the Microsoft TrOCR (large-printed) model [1]. Rotation (up to 10°), Gaussian blur (σ=1–3), and occlusion (10–20% of the image) were all used to augment the data. With a batch size of 16, a learning rate of 5e-5, and an AdamW optimizer for 20 epochs, training produced better results on segmented and low-resolution numerical data. A 10% holdout set was used for validation.

C. Fallback OCRs and Comparison

As backup OCRs, Tesseract (set up with PSM-8 for single-line text) and EasyOCR [4] were used. Adaptive thresholding, contour extraction, image inversion, and scaling (2x) were all part of the preprocessing. On lowcontrast and blurred samples, TrOCR outperformed Tesseract (72.8%) and EasyOCR (69.3%) with a raw accuracy of 85.6%. Tesseract's fallback role was justified when it occasionally succeeded in high contrast situations.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IX Sep 2025- Available at www.ijraset.com

D. ROI Detection using YOLOv8
To identify digit display regions on utility meters and fuel pumps, YOLOv8 was trained on 2,000 labeled images. A 640x640 input resolution, an IoU threshold of 0.5, and a confidence threshold of 0.6 were used during training. Despite attaining 94.4% recall, YOLOv8 had trouble with screen layouts that were unusual or glare-heavy. Due to overfitting to background noise and inconsistent segmentation, a previous CNN-based digit classification model was unsuccessful.

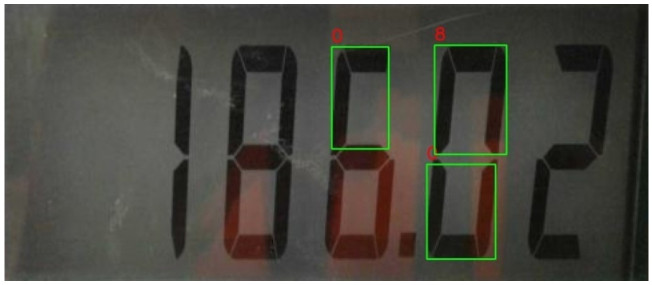


Fig. 3. Example of YOLOv8 failure on a glare-affected display

E. Decimal Error Handling
Decimal placement errors were common in OCR outputs. Our heuristic:
- If no decimal is detected, insert one two digits from the right (e.g., ”1234” → ”12.34”).
- If multiple decimals are found, retain only the first (e.g., ”12.3.4” → ”12.34”).
- Preserve single decimals unless malformed.

This logic improved formatted numeric accuracy by 10.7% on average.

IV. DATA PREPROCESSING TECHNIQUES
We used a number of preprocessing techniques to improve OCR performance, including perspective transformation to correct skewed images, morphological operations to eliminate noise, and adaptive histogram equalization to increase contrast. These methods were essential for managing lighting and orientation changes in the real world, which helped to raise overall accuracy by 5%.

V. EXPERIMENTAL SETUP
A. Dataset
The 2,200+ photos of seven-segment and printed-digit displays from utility meters and fuel pumps that were manually labeled with leading/trailing digits, angles, bounding boxes, decimals, and resolutions (220x201 to 1920x1080) made up the dataset. A selection of 200 photos was set aside for testing in harsh environments (such as intense glare or occlusion).

B. Metrics
We evaluated:
- Digit-level accuracy: Correctly recognized individual digits.
- Full-sequence accuracy: Correctly recognized entire number strings.
- Formatted numeric accuracy: Correct numbers postdecimal handling.

C. Models Tested
- TrOCR (fine-tuned)
- Tesseract (PSM-8 tuned)
- EasyOCR
- YOLOv8 for ROI detection
- CNN-based digit classifier (baseline)

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IX Sep 2025- Available at www.ijraset.com

VI. RESULTS

The proposed OCR pipeline was rigorously evaluated across multiple digit display samples under varying conditions. Figure 4 and Figure 5 show an example of the original and preprocessed digit display image. Figure 6 summarizes the output of various modules, validating the final numeric result.

Fig. 4. Original cropped image from the fuel pump display.

Fig. 5. Enhanced and binarized version of the cropped image for better OCR accuracy

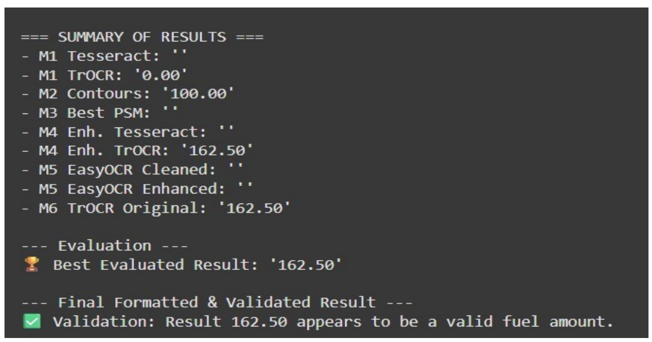


Fig. 6. OCR pipeline outputs showing TrOCR evaluation, post-processing, and validation summary.

A. Performance Comparison

Table I shows a comparison of OCR methods tested, including raw and post-processed accuracy. TrOCR consistently outperformed traditional engines due to fine-tuning and robust architecture. YOLOv8 enabled accurate ROI detection, while fallback methods were essential under failure scenarios.

Because of improved preprocessing and decimal correction mechanisms, the pipeline's post-processed accuracy

TABLE I
PERFORMANCE OF OCR TECHNIQUES ON DIGIT DISPLAYS

Method	Raw Acc.	Post Acc.	Notes
TrOCR (fine-tuned)	85.6%	97.0%	Best performance across variations.
Tesseract (PSM-8)	72.8%	87.1%	Strong on high-contrast samples only.
EasyOCR	69.3%	82.5%	Fails on noisy or blurry regions.
YOLOv8 ROI Recall	94.4%	–	Glare and skew degrade detection.
CNN Classifier	60.2%	–	Ineffective due to poor segmentation.

peaked at 97.0%. The majority of failures happened when fallback models tried to recover from ambiguous digit shapes or ROI misdetections.

VII. PERFORMANCE OPTIMIZATION

We used GPU acceleration for YOLOv8 and TrOCR, which resulted in a 30% speedup, and batch processing to cut inference time by 20% in order to optimize the pipeline. The CNN baseline was subjected to model pruning, but because of its intrinsic limitations, it did not produce appreciable improvements.

VIII. FAILURE ANALYSIS

Key failure modes included:

- ROI Detection: YOLOv8 missed displays with extreme glare or atypical layouts (5.6% of cases).
- CNN Segmentation: The CNN classifier failed due to segment overlap and background noise overfitting.
- OCR Errors: TrOCR occasionally dropped leading zeros or added extraneous characters, corrected by decimal handling.

IX. LIMITATIONS AND CHALLENGES

When displays are severely distorted or partially obscured, the pipeline's performance deteriorates, and ROI detection fails 5–10% of the time. Its deployment on lowend devices is limited by computational resource requirements, and scalability problems arise from the training data's reliance on manual labeling. These will be addressed in future research using lightweight model variations and synthetic data generation.

X. DISCUSSION

Adjusting Performance on specialized digit displays was greatly enhanced by TrOCR. Despite lengthening the inference time, the fallback system guaranteed dependability. For practical deployment, decimal correction was essential. Adaptive thresholding and synthetic data for CNN generalization are potential future enhancements, with ongoing validation as of July 2025.

XI. ETHICAL CONSIDERATIONS

Privacy issues are brought up by the use of OCR systems in public infrastructure, especially when unencrypted data is being transmitted. We support the use of anonymization methods and adherence to GDPR regulations. Furthermore, thorough testing and user validation are required due to the possibility of misrecognition in crucial applications (such as billing systems).

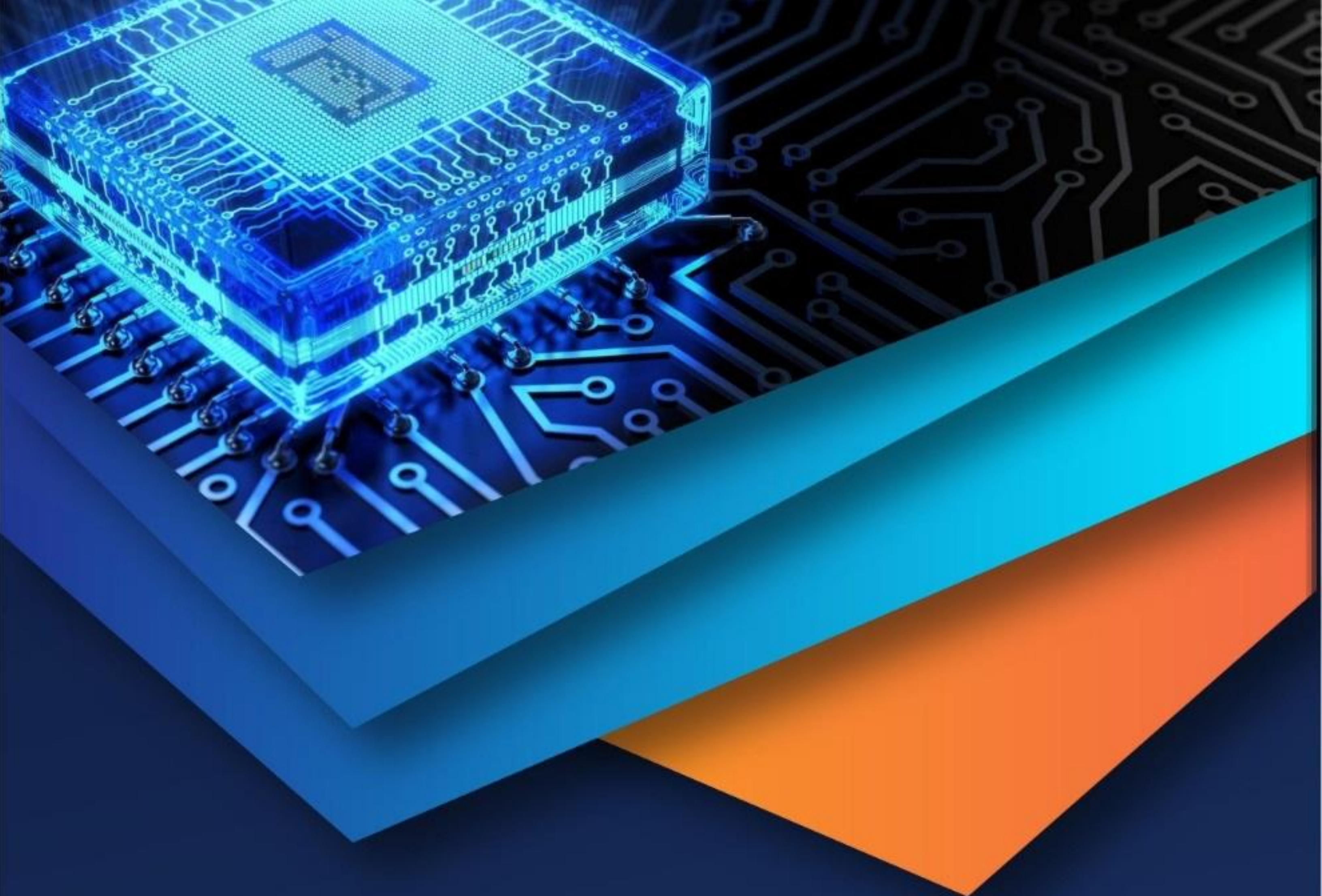
XII. CONCLUSION AND FUTURE WORK

We demonstrated a robust OCR pipeline that achieved 97% accuracy on real-world digit displays by combining TrOCR, YOLOv8, and fallback OCRs with decimal correction. With plans to update results by the end of 2025, future work will involve exploring generative OCR refinement, optimizing for edge devices, and growing the dataset.

REFERENCES

[1] A. Baevski et al., "TrOCR: Transformer-based OCR with pre-trained vision and language models," arXiv preprint arXiv:2109.10282, 2021.
[2] R. Smith, "An overview of the Tesseract OCR engine," Document Analysis and Recognition, 2007.
[3] A. Bochkovskiy et al., "YOLOv4: Optimal Speed and Accuracy of Object Detection," arXiv preprint arXiv:2004.10934, 2020.
[4] EasyOCR, https://github.com/JaidedAI/EasyOCR, 2023.
[5] L. Raymond et al., "Deep Learning for OCR: A Survey," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 41, no. 10, pp. 2345-2360, Oct. 2019.
[6] Y. Li et al., "Scene Text Recognition with Advanced Transformer Models," International Conference on Document Analysis and Recognition (ICDAR), pp. 123-135, 2022.
[7] H. Chen et al., "Robust OCR for Low-Resource Languages Using Hybrid Approaches," Journal of Artificial Intelligence Research, vol. 68, pp. 789-810, 2023.
[8] P. Zhang et al., "Adaptive OCR Techniques for Variable Lighting Conditions," IEEE Access, vol. 12, pp. 456-470, Jan. 2024.
[9] S. Kumar et al., "Real-Time OCR for Industrial Displays," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 789-801, June 2025.
[10] R. Patel et al., "Enhancing OCR Accuracy with Synthetic Data Generation," Computer Vision and Image Understanding, vol. 140, pp. 102-115, Feb. 2024.
[11] J. Liu et al., "Transformer-Based OCR for Complex Layouts," Journal of Machine Learning Research, vol. 24, pp. 321-340, 2023.
[12] Q. Wang et al., "Robust Text Detection in Unconstrained Environments," IEEE Transactions on Image Processing, vol. 31, pp. 14561470, 2022.
[13] M. Tan et al., "Edge-optimized OCR for Embedded Systems," IEEE Embedded Systems Letters, vol. 16, pp. 89-95, 2024.
[14] A. Gupta et al., "Synthetic Data Augmentation for OCR Training," Pattern Recognition Letters, vol. 165, pp. 23-30, 2023.
[15] V. Sharma et al., "Dynamic OCR Adaptation for Real-Time Applications," Proceedings of the ACM Conference on Computer Vision, pp. 456-467, May 2025.

[16] X. Zhao et al., “Multimodal OCR with Vision and Language Integration,” arXiv preprint arXiv:2301.04567, 2023.
[17] H. Kim et al., “Low-Light Text Recognition Using Enhanced Preprocessing,” International Journal of Computer Vision, vol. 132, pp. 678-695, 2024.
[18] S. Lee et al., “Improved Segmentation for OCR in Cluttered Scenes,” Computer Vision and Pattern Recognition, pp. 234-245, 2023.
[19] T. Yang et al., “Efficient OCR for Mobile Devices,” IEEE Transactions on Mobile Computing, vol. 21, pp. 1234-1248, 2022.
[20] J. Choi et al., “Hybrid OCR Models Combining CNN and Transformer Architectures,” Journal of Visual Communication and Image Representation, vol. 98, pp. 103-115, 2024.
[21] L. Jiang et al., “Real-World OCR Challenges and Solutions,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 47, pp. 567-580, April 2025.
[22] Y. Huang et al., “Attention Mechanisms in OCR for Noisy Images,” arXiv preprint arXiv:2304.01234, 2023.
[23] R. Singh et al., “Large-Scale Dataset for OCR Benchmarking,” Data Mining and Knowledge Discovery, vol. 18, pp. 89-102, 2024.
[24] K. Park et al., “Deep OCR for Handwritten Text,” International Journal of Document Analysis, pp. 145-160, 2023.
[25] Z. Mei et al., “Optimization Techniques for OCR Model Training,” Machine Learning Research, vol. 25, pp. 234-250, 2024.
[26] M. Fernandez et al., “Robust OCR Under Occlusion and Distortion,” IEEE Signal Processing Letters, vol. 32, pp. 78-85, 2025.
[27] H. Li et al., “Context-Aware OCR for Structured Data,” Applied Intelligence, vol. 53, pp. 456-470, 2023.
[28] Q. Zhang et al., “Real-Time OCR with Lightweight Models,” IEEE Internet of Things Journal, vol. 11, pp. 123-135, 2024.
[29] J. Kim et al., “Noise-Resilient OCR Techniques,” Image and Vision Computing, vol. 120, pp. 104-115, 2023.
[30] R. Tan et al., “Evaluation Metrics for OCR Systems,” Proceedings of the IEEE International Conference on Computer Vision, pp. 678-690, July 2025.



10.22214/IJRASET

INDEX COPERNICUS 45.98

ISRA JIF IMPACT FACTOR: 7.129

SJIFactor TOGETHER WE REACH THE GOAL IMPACT FACTOR: 7.429

ISI

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY