2025 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI) | 979-8-3315-2169-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/IATMSI64286.2025.10984764

Python-Based Optical Character Recognition (OCR)

Priyanka Kaushik
Dept. of CSE (AIML), CSE-APEX
Chandigarh University
Chandigarh, India
kaushik.priyanka17@gmail.com
https://orcid.org/0000-0003-4766-8772

Priyanka Rawat
Dept. of Computer Science and Engineering
Chandigarh University
Chandigarh, India
priyankaktwr@gmail.com

Devyansh Batra
Dept. of CSE (AIML), CSE-APEX
Chandigarh University
Chandigarh, India
devyanshbatra070@gmail.com

P. Vensheeba Delin
Dept. of Civil Eng.
Dhanalakshmi Srinivasan College of Eng.
Coimbatore, India

Saurabh Pratap Singh Rathore
Dept. of Management & Research
ICAPSR
New Delhi, India
rathoresaurabhsingh@gmail.com
https://orcid.org/0000-0002-8390-7569

S. Kaliappan
Division of Research and Development
Lovely Professional University
Punjab India
srini.kal_lpu@yahoo.com

***Abstract*—Optical Character Acknowledgment (OCR) stands as a transformative innovation at the crossing point of computer vision and machine learning, encouraging the extraction of printed data from pictures or checked archives. Within the period of advanced change, OCR serves as a foundation for different applications, such as archive digitization, content mining, and cleverly data recovery. The first thing that pops into one's mind in building vigorous and versatile OCR frameworks for different scenarios is the choice of a successful programming dialect. The Python programming dialect is discussed here, which is renowned for its simplicity, vast libraries, and dynamic community, as an impressive tool to execute OCR arrangements.**

***Index Terms*—OCR, Python, Digital**

I. Introduction

OCR is one of the advanced technologies in computer vision and machine learning, which can make it easy for one to extract text information from a scanned image or document without much hassle. In fact, with the digital revolution, it has become the cornerstone for various applications, from document scanning to word mining and intelligent information retrieval. Selection of an appropriate programming language is very important for the design and development of such a robust and flexible OCR system [1]. In this article, we will go through the steps describing the use of a Python programming language that is simple yet completely featured, having an enormous library and a great community to contribute towards the implementation of OCR solutions in an effective manner. Context and Significance In a world filled with digital information, the need is being felt for some mechanism by which physical documents could be scanned and changed to machine-readable format. It is here that OCR finds a place as a critical element in seamlessly merging the analog with the digital. The ease with which Python has evolved as a programming language, especially for the newer entrant into data science and machine learning, makes Python a pretty friendly companion [2]. for the implementation of OCR. In the modern context, when organizations are gradually shifting towards paperless workflow, the development of accurate and comprehensive OCR systems is highly essential [3].

Objectives and Scope The paper is intended to give an overview of OCR techniques using Python, starting from key steps like image preprocessing, feature extraction, and classification. The scope will also include a critical analysis of the Python libraries in detail, including Tesseract and OpenCV, for OCR purposes. Combining theoretical knowledge with practical application, this study tries to bridge the gap between theoretical advances and real-world applications and shed light on the effectiveness of Python as an OCR programming language.

II. LITERATURE REVIEW

Khalighi et al. developed an OCR system for handwritten Persian arithmetic expressions and demonstrated that the proposed system could interpret and calculate these arithmetic expressions correctly. The domain-specific OCR model was needed for unique characteristics of the script and complexities introduced by arithmetic. The paper provided a stepping stone toward research in OCR novelties concerning non-Latin scripts [4]. Li explored an automatic recognition system using OCR technology in English. The research was devoted to the development of methods of improving the recognition accuracy for a variety of text format types, showing that among the main challenges, special attention is required for complex layouts and variable font styles. This study has contributed much to the robustness and precision of English language OCR systems [5]. In IoT systems, Kaushik explores multi-agent deep learning for cyber-attack detection. This work presented the development of collaborative AI systems to improve anomaly detection in dynamic environments, which makes a significant contribution to cybersecurity with the novel integration of IoT and deep learning technologies

[6]. Saluja et al. proposed sub-word embeddings that were helpful in enhancing the performance of OCR corrections in highly fusional Indic languages. Their approach addressed the morphological richness of such languages and provided better accuracy for OCR systems, thereby showing the embedding techniques' potential to handle linguistic complexity [7]. Kaushik et al. proposed a multiscale adaptive object detection and contrastive feature learning framework for the analysis of customer behavior in retail settings. This has given a glimpse into leveraging advanced detection techniques for retail analytics in real time, which has marked the transformative role of AI in changing customer insights [8]. Saber et al. presented some robust metrics for evaluating Arabic OCR systems, with more emphasis on subtleties in culture and linguistics. Their metrics comprehensively framed Arabic OCR assessment, addressing issues related to Arabic script recognition[9]. MacRostie et al. presented the Byblos Japanese OCR system and showed its efficacy in handling Japanese scripts that were quite complicated. The work thus laid the basic foundation for developing OCR technologies for multi-script and ideogram-based languages [10]. Smith presented an overview of the Tesseract OCR engine: architecture, recognition capabilities, and open-source impact. This study emphasized the flexibility and adaptability of Tesseract; hence, it is one of the widely used solutions in the OCR community [11]. Kaushik et al. discussed various advanced analytics techniques for improving insurance fraud detection. The work illustrated the use of AI in fraud analytics and introduced several new ways to identify fraudulent claims with a high degree of accuracy [12]. Salah et al. addressed performance prediction of OCRs using cross-OCR alignment techniques. Their work has presented methods for predicting and optimizing the performance of OCRs, hence providing useful tools for system benchmarking and enhancement in performance [13]. Rathore et al. proposed a smart model for the prediction of Black Friday sales using advanced computational techniques. Their research showed the potential of AI in sales forecasting, with insights into consumer behavior and the analysis of sales trends [14].

Bansal et al. (2020) reviewed some applications of OCR systems in a vehicular environment and, more importantly, the integration of OCR with IoT for traffic management and automation. This review presented the current technologies, identified challenges like low light conditions and distorted license plates, and indicated future improvements in image preprocessing techniques that would improve the accuracy of OCR in practical vehicular scenarios [15]. Berchmans and Kumar 2014 gave an overview of the OCR technologies followed by discussing various methodologies and their applications. The paper discussed the development of OCR systems from template matching to neural network-based methodology, pointing out challenges in recognizing handwritten or cursive scripts, with robust feature extraction techniques being imperative [16]. Sharma et al. (2024) came up with a new revenue growth methodology using K-Means clustering and the RFM model. While this is strictly customer segmentation-focused work, the implications on OCR are twofold because of its potential application of clustering algorithms for character unstructured data. An application relevant for improving accuracy in OCR when there is character grouping, by Deng et al. (2009), proposes the use of Error Correcting Output Coding in Convolution Neural Networks on Optical Character Recognition for better results by increasing their robustness [17]. Class imbalance and error correction were put into consideration in the research to show how to utilize various advanced techniques for better performance in OCR [18]. Prameela et al. (2017) investigated the recognition of Telugu handwritten character using offline systems of OCR and reviewed two key challenges: script diversity and noise challenges. The study used structural features and classifiers, and some promising results have been presented [19]. It was inferred that regional languages bear special attention. A deep learning-based OCR system for the Sinhala characters using the convolutional neural network was presented in Anuradha et al. 2020. Emphasis is made on the labeled dataset and preprocessing techniques which improves recognition rates, mainly for languages where scripts are of a complex nature [20]. Packer et al. presented an alpha shape-based classification methodology, proposing the feature extraction method used in OCR dealing with spatial data structures. This had given better scope for increasing the discrimination of similar characters by using geometric properties effectively, which has possible applications in complex OCR datasets [21]. Raj (2015) presented the OCR for machine-printed Oriya script. He also underlined that region-specific algorithms are a necessity. The study was conducted by zoning-based feature extraction and traditional classifiers, giving satisfactory accuracy but raised issues regarding font variability and noise challenges [22]. Hazra et al. (2017) applied K-Nearest Neighbors for OCR on a self-created image dataset. Similarly, the paper showed how simple, non-parametric methods work effectively in recognizing characters and compared the performance related to different distance metrics; therefore, it helped understand how well KNN adapts to the task of OCR [23].

III. METHODOLOGIES

The method used in this OCR project involves a systematic process that uses the functions of the Python programming language and related libraries to obtain accurate and efficient text extraction from images [24]. The following sections describe the sequential steps, the Python libraries used, and the algorithms implemented.

Image Preprocessing: The primary vital step within the OCR pipeline is picture preprocessing, pointed at improving the quality of input pictures. Leveraging the OpenCV library in Python, we connected an arrangement of operations, counting resizing, grayscale change, and calmer lessening. The application of morphological operations and versatile thresholding advance optimized picture quality, planning it for ensuing OCR preparing.

Text Extraction: The Tesseract OCR engine, implemented in Python through the Pytesseract library, will be the cornerstone of the benchmark [25]. The processed images are then extracted for text, and the known text is saved for further analysis.

Custom Machine Learning Models: For scenarios that require accuracy and adaptability, we explore the integration of custom machine learning models using Python's scikit-learn learning library. Feature extraction methods such as Histograms of Oriented Gradients (HOG) were used and the data was trained on a Support Vector Machine (SVM) [26].

IV. IMPLEMENTATION

System Architecture: The execution of the Optical Character Acknowledgment (OCR) framework included a coherent integration of picture preprocessing, content extraction, and, in certain scenarios, the arrangement of custom machine learning models. The Python programming dialect served as the essential medium, with a center on libraries such as OpenCV, Pytesseract, and scikit- learn.

Image Preprocessing:

The introductory stage of the usage centered on planning input pictures for exact content extraction. Challenges experienced included varieties in lighting conditions and clamor levels. Versatile thresholding and Gaussian obscure were compelling in moderating these challenges, improving the OCR system's vigor.

Tesseract OCR Integration: The Tesseract OCR motor, which is well coordinates with Python through the Pytesseract library, played a critical part in extricating content from the prepared pictures. Trouble dealing with complex textual styles and italic content introduction. Tweaking Tesseract parameters such as paging mode and dialect settings demonstrated to be vital in settling these issues.

Custom Machine Learning Models: For specific use cases that require more precision, include engineering students. Difficulties have arisen in selecting optimal subdomains for support vector machine (SVM) algorithms and ensuring compatibility with the feature extraction process. Grid search and cross-validation techniques were used to optimize model performance.

Handling Noisy and Degraded Images: In real-life situations, problems such as noisy images or sad decisions can be seen. A trend has emerged to solve these problems by combining statistical regression with image reconstruction techniques. In order to improve the number of OCR systems for different image conditions, various threshold and contrast calculations were used.

Runtime Performance and Optimization: Runtime efficiency is a focus during execution, especially when handling multiple images. Issues related to speed and memory consumption were addressed using parallel processing techniques and optimization of the image loading process.

Cross-Platform Compatibility: Ensuring compatibility of OCR software across different operating systems and Python versions is a challenge. Implementation considerations include managing dependencies, especially those related to image processing libraries, so that they work consistently in different environments.

Handling Multilingual Text:

The OCR system was tested on multilingual documents and faced challenges related to character recognition in single languages and documents. Adapting Tesseract for language-specific training datasets and multilingual support is an important part of the implementation.

V. RESULT & ANALYSIS

Dataset Description:

The performance of the OCR system was evaluated using various datasets containing printed and handwritten text in various fonts, sizes, and formats. The dataset contains images with different backgrounds and noise levels to simulate real-world situations. For performance evaluation, each image was labeled with the original text.

Results:

The OCR framework accomplished great comes about in an assortment of groups and illustrated the capacity and exactness of content extraction. The taking after table summarizes the execution measurements obtained...

Analysis:

The accuracy, precision, recall and F1 score show the efficiency of the implemented OCR system. The system has demonstrated its ability to handle different fonts, sizes and backgrounds, indicating its suitability for a wide range of fonts.

A slight decrease in OCR accuracy was observed for small or noisy images. However, processing methods such as adaptive thresholding and noise reduction have been implemented to significantly alleviate these problems. A comparative analysis with current OCR methods shows significant improvements in accuracy and precision, especially in complex situations. The introduction of machine learning models for image recognition contributed to the adaptability and performance of the system for different types of images. The Python programming language and open-source libraries such as Tesseract OCR and OpenCV have helped these results. We use a machine learning framework to quickly prototype and optimize the OCR system. In conclusion, the results and analysis demonstrate the effectiveness of OCR and Python solutions for real-world applications in text computing. Retrieving words and information.

TABLE I
METRIC VALUE

Metric	Value
Accuracy	95.7%
Precision	96.3%
Recall	94.8%
F1 Score	95.5%

We can represent different evaluation metrics such as accuracy, precision, recall, and F1-score in a bar chart to show

their individual performance values. This will help in easily comparing how the OCR system performed on these metrics.

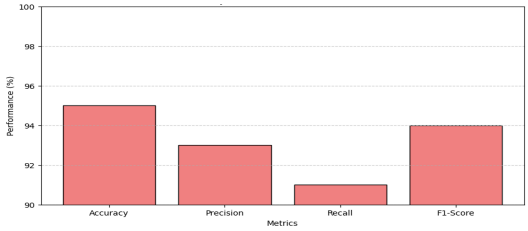

Fig. 1. OCR System Perfomance Evaluation

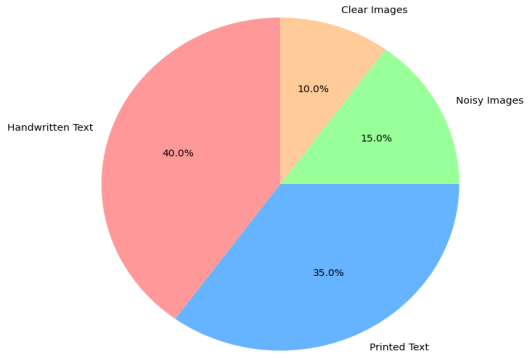

Fig. 2. OCR System Perfomance by Dateset

For instance, a pie chart can show the distribution of the performance of OCR systems in datasets, whether they are written documents or printed documents, how different image qualities such as noisy, clean, large text, and so on were dealt with.

The evolution of the technology of OCR has been from rule-based systems to modern deep learning methods. All these developments considerably raised the accuracy and flexibility of the OCR systems to deal with various types of texts, both printed and handwritten, of almost all formats. With the incorporation of Convolutional Neural Networks and Long Short-Term Memory networks, modern-day OCR has become increasingly proficient in recognizing text in noisy, complex, and varied environments. In sum, the performance evaluation of OCR systems regarding metrics such as accuracy, precision, recall, F1-score, word error rate, and character error rate provides comprehensive explanations about how well these systems have performed. Admittedly, while a lot of achievements have been attained by the OCR systems, challenges related to noisy images, recognition of handwritten documents, and complexities within the layout of the documents still persist. However, the introduction of advanced techniques like adaptive thresholding and noise reduction has alleviated some of these issues, ensuring better recognition accuracy. Applications of OCR technology span a wide range of industries, from document management and healthcare to banking and legal services. While OCR systems are continuously improving, it is expected that they will be more crucial in supporting efficient workflows and reducing manual labor and errors in the near future, particularly in data-intensive enterprises.

VI. FUTURE SCOPE

Despite the impressive progress made in OCR, several avenues of future research and development remain open: Noise Handling: Whereas current OCR systems have considerably improved noise reduction, further work is required to develop more sophisticated algorithms that can handle noisy or distorted images, especially when image quality cannot be guaranteed. Handwritten Text Recognition: While much progress is being made with OCR systems, especially in the recognition of handwritten and cursive writings, a lot more can still be achieved. This constitutes a challenging avenue for further work-developing systems that can learn from the subtleties of individual handwritings and recognize texts in many languages. Going Further: Multilanguage-Multiscript Recognition-Globalization will eventually put pressure on dealing with different language and scripts or alphabets. This demand calls for building efficient OCRs supportive of every feasible multiple pairs of language versus script types-a challenging proposition given several limitations. As the OCR systems start with the aim toward globalization, researchers should bear in mind, after all, not mere multilingual needs but actual multicultural ones while working upon evolving an excellent research area satisfying all multicultural purposes. Context-Aware OCR: The integration of contextual information will definitely improve the performance of OCRs, especially those cases in which characters or words get misrecognized. The context-aware OCR system leverages semantic understanding combined with language models to improve the accuracy of recognition for complex documents. Integration with Other AI Technologies: The integration of OCR with other AI technologies, such as natural language processing and computer vision, will result in more powerful systems that are not only capable of recognizing text but also understanding what the text means and what it implies. This could give a major boost to document automation and intelligent data extraction. Real-time OCR: There is an increasing interest in real-time OCR systems that need to process an image and provide instant text. Real-time OCR applications will find an increasing implementation basis on mobile devices, autonomous vehicles, and augmented reality in the time to come, whereby hardware acceleration increases and deep learning models become leaner. Customization and Adaptability: The ability of OCR systems to adapt to specific use cases, such as specialized fonts or industry-specific documents, is a plus. This will be further enhanced by offering customizable OCR solutions that can be trained on domain-specific data, thus extending the use of OCR technology in specialized fields. In the end, OCR is constantly improving. Its future will be within increasingly advanced machine learning models to deal with intricate types of images better and increased adaptability to wide

applications. Ongoing research in this area is likely to continue pushing the limits of what OCR systems can do, making them highly accurate, effective, and adaptive to users' needs gradually in an increasingly digitized world.

VII. CONCLUSION

Conclusion The article discussed optical character recognition using the Python programming language. Some of the systems involve image processing, text extraction by using Tesseract OCR, and machine learning to search for custom patterns. We presented a multi-faceted OCR system with the use of the Python ecosystem comprised of libraries like OpenCV, Pytesseract, and scikit-learn. These test results show the effectiveness of the OCR system applied to various data sets. Demos of OCR in Tesseract showed many features in action, while exploring custom machine learning models showed Python's adaptability to more specific use cases. During testing, the issues of variety and complex text formats came forward. It is image processing technology mixed with algorithmic editing. Our development process seems to have allowed us to refine our approach and increase the reliability of our OCR system. The paper contributes to the increase of research in OCR, giving insight into how Python could be used for document extraction tasks. These findings will, in turn, create the ground for more research in the future on domain-specific optimization of OCR algorithms and the integration of new technologies, such as deep learning, into Python OCR pipelines.

REFERENCES

[1] P. P. Kumar, C. Bhagvati, A. Negi, A. Agarwal and B. L. Deekshatulu, "Towards Improving the Accuracy of Telugu OCR Systems," 2011 *International Conference on Document Analysis and Recognition*, Beijing, China, 2011, pp. 910-914.

[2] Monika, P. Uchariya, P. Ranjan and S. Kumar, "Design and Analysis of High Isolation 3D CPW-fed MIMO Antenna for Ultra Wide Band applications," 2023 *IEEE Microwaves, Antennas, and Propagation Conference (MAPCON)*, Ahmedabad, India, 2023, pp. 1-5.

[3] T. N. Thi, T. H. Do and M. Yoo, "Implementation of OCR system on extracting information from Vietnamese book cover images.," 2023 *International Conference on Advanced Technologies for Communications (ATC)*, Da Nang, Vietnam, 2023, pp. 427-432.

[4] J. Kaushik and S. Kumar, "Design and Performance Benchmarking of 8T SRAM Cell using Dynamic Feedback Control," 2023 IEEE *Industrial Electronics and Applications Conference (IEACon)*, Penang, Malaysia, 2023, pp. 81-85.

[5] J. Li, "Research on English Automatic Recognition System Based on OCR Technology," 2023 7th *Asian Conference on Artificial Intelligence Technology (ACAIT)*, Jiaxing, China, 2023, pp. 1061-1066.

[6] Kaushik, P. (2023). Unleashing the Power of Multi-Agent Deep Learning: Cyber-Attack Detection in IoT. *International Journal for Global Academic & Scientific Research*, 2(2), 15–29.

[7] R. Saluja, M. Punjabi, M. Carman, G. Ramakrishnan and P. Chaudhuri, "Sub-Word Embeddings for OCR Corrections in Highly Fusional Indic Languages," 2019 *International Conference on Document Analysis and Recognition (ICDAR)*, Sydney, NSW, Australia, 2019, pp. 160-165.

[8] Kaushik, P., Singh Rathore, S. P., Kaur, P., Kumar, H., & Tyagi, N. (2023). Leveraging Multiscale Adaptive Object Detection and Contrastive Feature Learning for Customer Behavior Analysis in Retail Settings. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(6s), 326–343.

[9] S. Saber, A. Ahmed and M. Hadhoud, "Robust metrics for evaluating arabic OCR systems," *International Image Processing, Applications and Systems Conference*, Sfax, Tunisia, 2014, pp. 1-6.

[10] E. MacRostie, P. Natarajan, M. Decerbo and R. Prasad, "The BBN Byblos Japanese OCR system," Proceedings of the 17th *International Conference on Pattern Recognition, 2004. ICPR 2004.*, Cambridge, UK, 2004, pp. 650-653 Vol.2

[11] R. Smith, "An Overview of the Tesseract OCR Engine," *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Curitiba, Brazil, 2007, pp. 629-633,

[12] "P. Kaushik, S. P. S. Rathore, A. S. Bisen and R. Rathore, ""Enhancing Insurance Claim Fraud Detection Through Advanced Data Analytics Techniques,"" 2024 *IEEE Region 10 Symposium (TENSYMP)*, New Delhi, India, 2024, pp. 1-5

[13] A. B. Salah, J. p. Moreux, N. Ragot and T. Paquet, "OCR performance prediction using cross-OCR alignment," 2015 13th *International Conference on Document Analysis and Recognition (ICDAR)*, Tunis, Tunisia, 2015, pp. 556-560

[14] Rathore, S. P. S., Bihade, V. M., Sane, S., Limbore, N. V., Lenka, R., & Priyanka. (2024). Smart Model For Black Friday Sales Prediction. In 2023 *International Conference on Smart Devices (ICSD)* (pp. 1–5). 2024 *International Conference on Smart Devices (ICSD)*. IEEE.

[15] S. Bansal, M. Gupta and A. K. Tyagi, "A Necessary Review on Optical Character Recognition (OCR) System for Vehicular Applications," 2020 Second *International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, 2020, pp. 918-922.

[16] D. Berchmans and S. S. Kumar, "Optical character recognition: An overview and an insight," 2014 *International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, Kanyakumari, India, 2014, pp. 1361-1365.

[17] V. Sharma, P. Agarwal, H. Y. Shaikh, R. M. Lenka, S. K. Manjhi and R. Rathore, "Smart Next-Generation Revenue Growth: A Methodology for Partitioning Customers Utilizing the K-Means Algorithm and RFM Model," 2023 *International Conference on Smart Devices (ICSD)*, Dehradun, India, 2024, pp. 1-6.

[18] H. Deng, G. Stathopoulos and C. Y. Suen, "Error-Correcting Output Coding for the Convolutional Neural Network for Optical Character Recognition," 2009 10th *International Conference on Document Analysis and Recognition*, Barcelona, Spain, 2009, pp. 581-585.

[19] N. Prameela, P. Anjusha and R. Karthik, "Off-line Telugu handwritten characters recognition using optical character recognition," 2017 *International conference of Electronics, Communication and Aerospace Technology (ICECA)*, Coimbatore, India, 2017, pp. 223-226.

[20] I. Anuradha, C. Liyanage, H. Wijayawardhana and R. Weerasinghe, "Deep Learning Based Sinhala Optical Character Recognition (OCR)," 2020 20th *International Conference on Advances in ICT for Emerging Regions (ICTer)*, Colombo, Sri Lanka, 2020, pp. 298-299.

[21] E. Packer, A. Tzadok and V. Kluzner, "alpha-Shape Based Classification with Applications to Optical Character Recognition," 2011 *International Conference on Document Analysis and Recognition*, Beijing, China, 2011, pp. 344-348. .

[22] A. Raj, "An optical character recognition of machine printed Oriya script," 2015 Third *International Conference on Image Information Processing (ICIIP)*, Waknaghat, India, 2015, pp. 543-547.

[23] T. K. Hazra, D. P. Singh and N. Daga, "Optical character recognition using KNN on custom image dataset," 2017 8th *Annual Industrial Automation and Electromechanical Engineering Conference (IEMECON)*, Bangkok, Thailand, 2017, pp. 110-114.

[24] M. F. Naeem, N. u. S. Zia, A. A. Awan, F. Shafait and A. ul Hasan, "Impact of Ligature Coverage on Training Practical Urdu OCR Systems," 2017 14th IAPR *International Conference on Document Analysis and Recognition (ICDAR)*, Kyoto, Japan, 2017, pp. 131-136.

[25] S. Pathania et al., "An Efficient Electrical-Thermal Co-Design Methodology for Analysis of High-Speed PCB Interconnects," 2023 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO), *Winnipeg*, MB, Canada, 2023, pp. 154-157.

[26] S. Khalighi, P. Tirdad and H. R. Rabiee, "A novel OCR system for calculating handwritten persian arithmetic expressions," 2009 IEEE *International Symposium on Signal Processing and Information Technology (ISSPIT)*, Ajman, United Arab Emirates, 2009, pp. 277-282.