iJRASET
International Journal For Research in Applied Science and Engineering Technology

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

**Volume:** 13 **Issue:** V **Month of publication:** May 2025

**DOI:** https://doi.org/10.22214/ijraset.2025.71177

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

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 V May 2025- Available at www.ijraset.com*

# Structural Similarity Index Measure Based Signature Verification Using Python

Himanshu Yadav[1], Aaryan Sharma[2], Vinay Krishna[3], Ayush Chaturvedi[4]
*Department of ECE, NIET, Greater Noida, India*

***Abstract:*** ***This paper presents a method for digital signature verification utilizing a combination of optical character recognition (OCR) and string similarity metrics, implemented in Python. The proposed method involves preprocessing input images, extracting text, and comparing the text for similarity using the SequenceMatcher algorithm. Additionally, the Structural Similarity Index (SSIM) is employed to assess the visual similarity between the images. This paper presents a method for signature verification without delving much into machine learning or deep learning. Experimental results demonstrate the effectiveness of this approach in verifying digital signatures, offering a robust solution for enhancing security in electronic documents.***

***Keywords: OCR, SSIM, gaussian blur, grayscale, thresholding***

## I. INTRODUCTION

Digital signatures have become a cornerstone of electronic document management, providing a reliable means to ensure authenticity, integrity, and non-repudiation in digital communications. As electronic transactions proliferate across sectors, the need for effective and efficient digital signature verification methods has become increasingly critical. Traditional verification approaches often depend on cryptographic techniques, which can be robust but may not adequately address variations in the visual representation of signatures.

This paper introduces an innovative approach that integrates optical character recognition (OCR) and string similarity metrics, specifically utilizing Python programming. By leveraging Python's extensive libraries and capabilities, this method aims to provide a reliable verification mechanism that accounts for the natural variations in handwritten signatures.

## II. BACKGROUND AND RELATED WORK

Digital signatures have become a cornerstone of electronic document management, providing a reliable means to ensure authenticity, integrity, and non-repudiation in digital communications. As electronic transactions proliferate across sectors, the need for effective and efficient digital signature verification methods has become increasingly critical. Traditional verification approaches often depend on cryptographic techniques, which can be robust but may not adequately address variations in the visual representation of signatures.

This paper introduces an innovative approach that integrates optical character recognition (OCR) and string similarity metrics, specifically utilizing Python programming. By leveraging Python's extensive libraries and capabilities, this method aims to provide a reliable verification mechanism that accounts for the natural variations in handwritten signatures.

## III. METHODOLOGY

### *A. Image Preprocessing*

The first and foremost step would definitely be preprocessing the input images in order to enhance the text extraction quality. The preprocessing which is done in python programming language can have many operations such as:

- *Conversion to Grayscale:* In this we convert the input images to the different shades of grey in order to reduce the colour noise which can impact the OCR performance severely. It is done using the OpenCV library.
- *Gaussian Blur*: Herein, we utilize the Gaussian blur which in turn smoothens the image in order to minimize the noise and improve text clarity.
- *Adaptive Thresholding:* This operation helps to convert the grayscale image to binary image. Binary image having only two colours helps to enhance the contrast between the background and the text.
- *Image Resizing:* To optimize the OCR recognition capabilities, we would be resizing the binary image obtained into a larger dimension.

*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 13 Issue V May 2025- Available at www.ijraset.com*

*B. Text Extraction*

We have defined a function as text_extractor using the EasyOCR library which extracts the text from the pre-processed images. If no text is found, the function returns an empty string which indicates that the OCR process was unsuccessful.

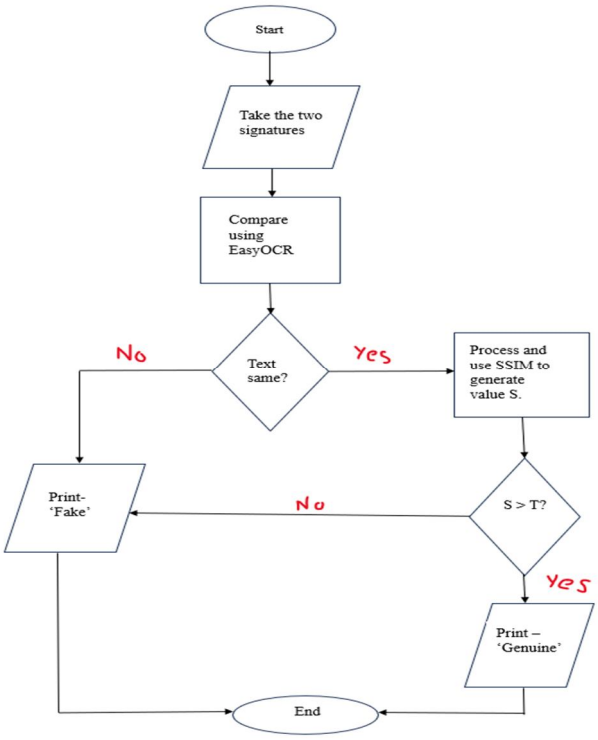
*C. Text Comparison*

Once the text is extracted from both signatures, the compare_text function employs the SequenceMatcher algorithm from Python's difflib library to compare the two strings. The algorithm calculates a similarity ratio, which measures the degree of similarity between the two text strings. A predefined threshold (default: 0.7) determines whether the texts are deemed similar enough to proceed to the next stage.

*D. Visual Similarity Assessment*

The match function orchestrates the entire process. It first preprocesses both input images, extracts the corresponding texts, and compares them. Say, if the similarity of texts is above the threshold, then only SSIM is performed and returns the similarity value otherwise it will return 0 indicating that the signature is fraud.

The following flowchart demonstrates the main working of the process:




## IV. EXPERIMENTAL SETUP

In order to check the effectiveness and accuracy of our proposed method, we did trial of it using a variety of signatures which included both genuine and forged signatures. It involved:

1) *Image Acquisition*: We collected signatures from internet, friends and other sources in order to have dynamic signatures and a large test set.
2) *Preprocessing:* Every collected image was preprocessed using the steps mentioned in Section III. After this they were ready for text extraction.
3) *Text Extraction and Comparison:* Each and every image underwent text extraction and comparison.
4) *SSIM Calculation:* Images with similar texts had their SSIM values calculated in order to check them.

The overall effectiveness of the method is based on its accuracy to identify the genuine and forged signatures and also the text extraction.

## V. RESULTS AND DISCUSSIONS

### A. Performance Metrices

The results were on the basis of two parameters:

1) *Text Similarity Scores:* In case of genuine pairs, the similarity averaged 0.8 and 0.53 for forged pairs which in turn showcases text dissimilarity.
2) *SSIM Values:* SSIM value further analysed the structural behaviour of the images which in turn also helped to determine whether they were done by same person or other.

### B. Case Studies

The results were on the basis of two parameters:

*1) Genuine Signature Match:*

- Extracted Text: "Jin Woo"
- Similarity Score: 0.88
- SSIM: 90%
- Conclusion: Strong match verified.

*2) Forged Signature:*

- Extracted Text: "Jin Woe"
- Similarity Score: 0.71
- SSIM: 73%
- Conclusion: Insufficient similarity; flagged for forgery.

### C. Limitations

Despite showcasing promising results, the proposed methodology has following limitations:

*1) Image Quality*: If the image quality is poor then the OCR accuracy is severely reduced which in turn affects the similarity scores.
*2) Threshold Sensitivity:* Even the choice of text similarity threshold can impact the verification output.
*3) Dynamic Signatures:* The method may struggle with signatures that exhibit significant dynamic characteristics, such as varying pressure or speed.

### D. Future Work

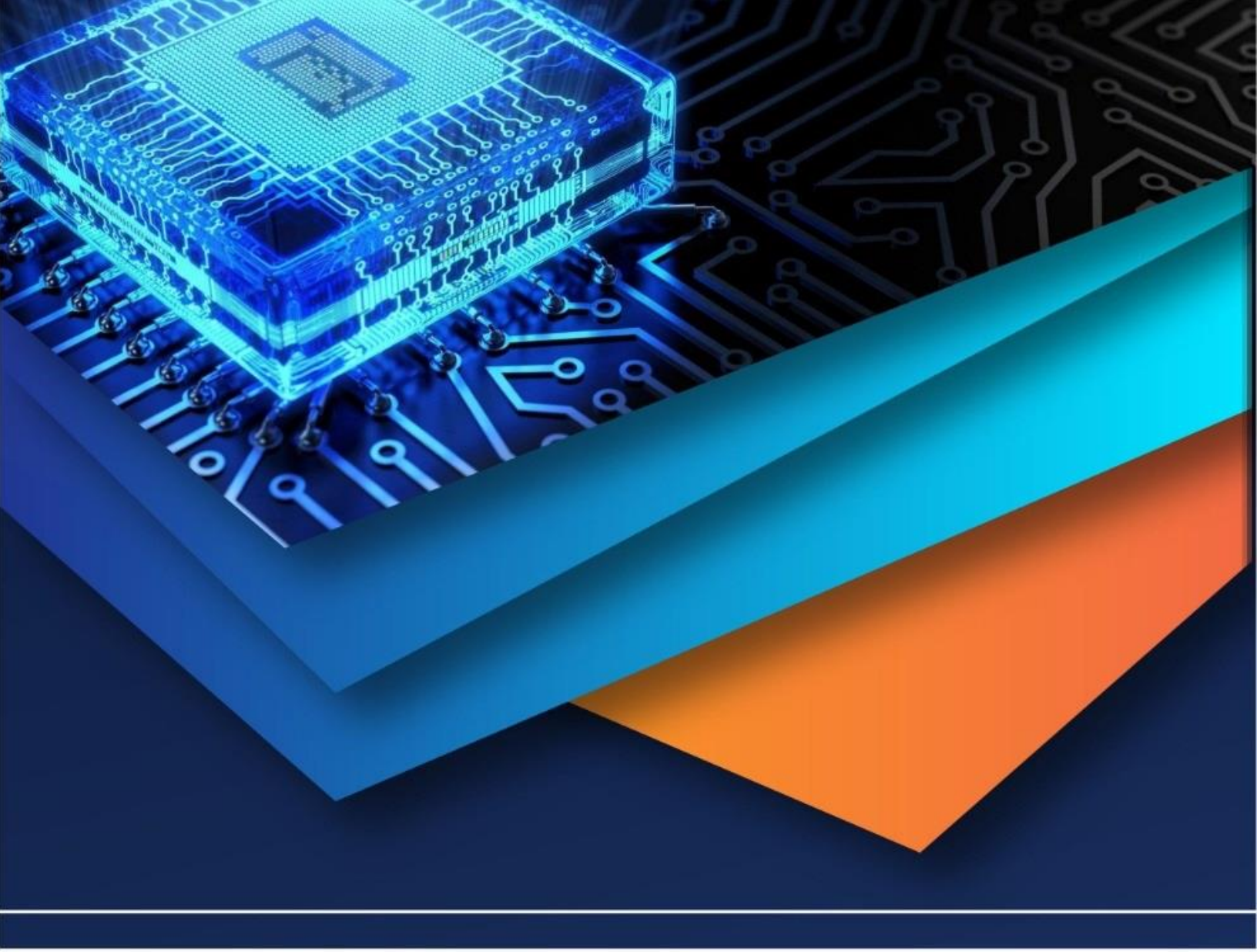
Our goal for future is to further enhance our methodology by overcoming its limitations stated above and develop new techniques and provide contributions in the field of image processing.

## VI. CONCLUSION

The paper showcases an innovative method of digital signature verification utilizing SSIM, text extraction using OCR, text comparison and image preprocessing in Python. This method demonstrates tremendous potential and a reliable method for signature verification. Experiment outcomes showcase that this method can easily identify legit and fraud signatures which in turn can be used to verify integrity of digital signatures.

## REFERENCES

[1] D. Swapna, P. Vikram, A. Sri Krishna and V. Sesha Srinivas, “A Survey on Local Patterns for Signature Verification”,2016.
[2] N. Venkateswara Rao, Dr. A. Srikrishna, Dr. B. Raveendra Babu and G. Rama Mohan Babu ”An Efficient Feature extraction and Classification of Handwritten digits using Neural Networks” in(IJCSEA) Vol.1, No.5, October 2011.
[3] Hsin-Hsiung Kao and Che-Yen Wen, “An Offline Signature Verification and Forgery Detection Method Based on a Single Known Sample and an Explainable Deep Learning Approach”, Appl. Sci. 2020, 10, 3716; doi:10.3390/app10113716.
[4] Gopichand G et.al., “Digital Signature Verification Using Artificial Neural Networks”, International Journal of Recent Technology and Engineering (IJRTE), ISSN: 2277-3878, Volume-7 Issue-5S2, January 2019.
[5] Harish Srinivasan et. al., “Machine Learning for Signature Verification”, ICVGIP 2006, LNCS 4338, pp. 761–775, 2006 © Springer-Verlag Berlin Heidelberg 2006.
[6] Jivesh Poddar et.al., “Offline Signature Recognition and Forgery Detection using Deep Learning”, The 3rd International Conference on Emerging Data and Industry 4.0 (EDI40), Procedia Computer Science 170 (2020) 610–617, Warsaw, Poland, April 6 - 9, 2020.
[7] Moises Diaz, Andreas Fischer, Miguel A. Ferar and Rejean Plamondom, “Dynamic Signature Verification System based on real Signature”, *IEEE*,2016.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY