International Journal for Modern Trends in Science and Technology
Volume 11, Issue 10, pages 19-28.
ISSN: 2455-3778 online
Available online at: http://www.ijmtst.com/vol11issue10.html
DOI: https://doi.org/10.5281/zenodo.17248555

Forgery Detection - Detect Forged Signatures using AI

SK. Naga Rehmathunnisa[1] | Y. Madhuri[2] | D. Mahalakshmi[2] | E. Anvitha[2] | M. Lakshmi Sowmya[2]

[1]*Associate Professor, Department of CSE, Vijaya Institute of Technology for Women, Enikepadu, AP, INDIA.*
[2]*Department of CSE, Vijaya Institute of Technology for Women, Enikepadu, AP, INDIA.*

To Cite this Article
SK. Naga Rehmathunnisa, Y. Madhuri, D. Mahalakshmi, E. Anvitha & M. Lakshmi Sowmya (2025). Forgery Detection - Detect Forged Signatures using AI. International Journal for Modern Trends in Science and Technology, 11(10), 19-28. https://doi.org/10.5281/zenodo.17248555

Article Info
Received: 04 September 2025; Accepted: 28 September 2025.; Published: 01 October 2025.

KEYWORDS

Convolutional Neural Networks (CNN),
Siamese Neural Network,
Deep Learning,
Supervised Learning,
Feature Extraction,
Classification,
Model Training,
Data Augmentation,
Transfer Learning.

ABSTRACT

Signature forgery poses significant risks across multiple sectors, including finance, legal, and government institutions. This project focuses on developing a robust signature forgery detection system utilizing a Siamese neural network model. The system is designed to analyze and compare two signature images an original signature and a suspected forged signature. By leveraging deep learning techniques, the system aims to provide accurate and reliable results, assisting users in identifying potential forgeries. And this project introduces an AI-powered forgery detection system that uses Artificial Intelligence and Machine Learning techniques to automate signature authentication. Withdeep learning models, particularly Convolutional Neural Networks (CNNs), the system analyzes document images, extracts intricate features such as signature strokes, text structures, and pixel nconsistencies, and accurately classifies signatures as genuine or forged. Overall, the Signature Forgery Detection project aims to enhance the security and reliability of signature verification, ensuring accurate authentication and preventing fraud across various sectors in an ever-evolving digital landscape

INTRODUCTION

The project aims to develop a Signature Forgery Detection system using advanced AI and machine learning techniques to provide accurate and reliable authentication for documents. In today's digital age, document verification is crucial across various industries, as the risk of fraud through forged signatures is ever-increasing. Our Signature Forgery Detection system seeks to address this challenge by offering a comprehensive solution to identify and prevent signature forgery in an efficient and secure manner.

The rise of digital transactions, online contracts, and e-signatures has made the need for robust forgery detection systems more urgent than ever. Both individuals and organizations face the risk of document manipulation and fraudulent activity. Our system aims

to bridge this gap by providing an intuitive and powerful platform that ensures the authenticity of signatures across various types of documents.

This documentation serves as a guide for understanding the architecture, features, and functionalities of the Signature Forgery Detection system. Whether you are a developer interested in contributing to the project, a business looking to integrate signature verification into your workflow, or a user seeking to verify the authenticity of signatures, this documentation provides detailed insights into the workings of our platform.

By leveraging AI and machine learning algorithms, we have created a robust and scalable system capable of detecting signature forgeries with high accuracy. Features such as dynamic threshold adjustment, image preprocessing, and multi-modal verification ensure the system's effectiveness in identifying forged signatures. Our goal is to provide users with a reliable tool that simplifies document verification while maintaining high standards of security and accuracy. We invite you to explore this documentation to gain a deeper understanding of the Signature Forgery Detection system and how it can transform the way signatures are verified, ensuring greater trust and security in document transactions. Thank you for your interest and support in our project.

EXISTING SYSTEM:

Manual Verification:

o Experts visually inspect signatures to detect inconsistencies.

o Time-consuming and prone to human errors, making it inefficient for large-scale verification, especially when dealing with high volumes of documents.

Traditional Image Processing Techniques:

Optical Character Recognition (OCR):

Extracts text from scanned documents but struggles with forged handwriting or manipulated text, making it ineffective for signature verification.

Edge Detection & Watermark Analysis:

Detects alterations based on structural inconsistencies but lacks adaptability to new forgery techniques, limiting its effectiveness in modern fraud detection.

Feature Extraction-Based Approaches:

Handcrafted Features:

Analyzes specific features like texture, ink variation, and signature verification patterns.

While effective for some types of forgery, this method has limited success in detecting more advanced forgeries, such as deep fake signatures and manipulated documents, which use sophisticated techniques to mimic real signatures. These existing methods often fall short when dealing with high-quality forgeries, making them inadequate for modern and real-world applications in signature verification.

Disadvantages:

1. Time-Consuming:Many traditional methods require significant manual effort, making them slow and impractical for large-scale document verification.
2. Prone to Human Error: Human inspection can result in oversights or misjudgements, leading to false positives or false negatives in detecting forgeries.
3. Inefficient for High Volumes: As the scale of document verification increases, traditional methods become less efficient and harder to maintain consistently, especially in industries with high document throughput.
4. Limited to Simple Forgery Detection: Traditional methods often rely on basic techniques that struggle to detect sophisticated forgeries, such as deepfake signatures or carefully manipulated documents.
5. Lack of Adaptability: Many traditional systems cannot evolve to handle new or complex forgery techniques. They may be outdated and unable to detect modern, advanced fraud methods.
6. Inaccurate in Handling Subtle Forgery Variations: Systems that focus on feature extraction or image processing may miss subtle but significant differences between genuine and forged signatures, leading to inaccuracies.
7. Limited Scalability: Traditional systems struggle to scale effectively, especially when large datasets are involved, limiting their use in real-world applications that require high throughput.

PROPOSED SYSTEM:

The proposed method leverages AI and deep learning models to improve forgery detection accuracy and automation. The process starts with preprocessing document images using techniques like grayscale conversion, denoising, and segmentation to enhance quality. Convolutional Neural Networks (CNNs) are

then used to automatically extract relevant features such as stroke consistency, signature curves, and pixel intensity variations. The CNN-based classification model is trained on labelled datasets of real and forged documents, utilizing Transfer Learning and Fine-Tuning to boost accuracy. The system is validated using benchmark datasets and synthetic forgeries, with performance evaluated using metrics like Accuracy, Precision, Recall, and F1-score. False positives and negatives are analyzed to refine the model.

This approach offers improved scalability, robustness, and reliability compared to traditional methods.

Advantages:

1. Accuracy: By leveraging deep learning models, the system automatically learns complex features, improving the accuracy of forgery detection compared to traditional methods.
2. Automation: The use of Convolutional Neural Networks (CNNs) automates the feature extraction and classification process, reducing human error and improving efficiency.
3. Scalability: The system can handle large datasets, making it scalable for high-volume environments, unlike traditional methods that are time-consuming and inefficient at scale.
4. Adaptability: With techniques like Transfer Learning and Fine-Tuning, the model can be updated and adapted to detect new types of forgeries, ensuring it stays relevant as
fraud techniques evolve.
5. Comprehensive Evaluation: The system uses benchmark datasets and synthetic forgeries for validation, ensuring reliable performance through rigorous evaluation with metrics such as Accuracy, Precision, Recall, and F1-score.
6. Robustness: The model's ability to handle subtle variations in signatures, such as stroke consistency and pixel intensity, makes it more effective in detecting sophisticated forgeries like deepfake signatures.
7. Reduced False Positives/Negatives: By analyzing and refining the model based on false positives and negatives, the system continuously improves its accuracy and reduces errors over time.
8. Improved Efficiency: Preprocessing and automated analysis save time compared to manual verification, enabling faster document verification processes.

1.2. Objectives:

1. Automated Forgery Detection: Replace manual signature verification with an automated system. Manual verification is slow, subjective, and prone to human error. AI provides consistent and fast assessments.
2. High Accuracy & Precision:
Minimize false positives (authentic signatures flagged as fake) and false negatives (forgeries marked as genuine). Ensures trust in financial transactions, legal processes, and secure document verification.
3. Learning from Data: Train AI models on genuine and forged signatures so they can learn distinguishing patterns. Why it matters: AI can detect subtle inconsistencies that humans might miss, especially in skilled forgeries.
4. Generalization Across Signers: Develop models that can verify signatures from a wide variety of individuals. In real-world applications (banks, contracts, ID verification), systems must handle diverse handwriting styles.
5. Enhance Security: Detect and prevent fraud involving forged documents or impersonation. Critical for identity verification, financial security, and legal compliance.

1.3. Importance of forged signatures using AI:

In an increasingly digital and fast-paced world, the authenticity of signatures remains a critical component in validating identities and securing agreements. From banking and legal transactions to government documents and academic records, signatures act as a personal stamp of approval and identity verification. However, with the rising sophistication of forgery techniques, traditional methods of verifying signatures are no longer sufficient. This is where Artificial Intelligence (AI) steps in, offering advanced, efficient, and highly accurate methods for detecting signature forgery. The integration of AI into signature verification processes holds immense importance for enhancing security, reducing fraud, and promoting trust in both physical and digital transactions.1.3 Multiple-Disease-Predictor-ML-Flask-WebApp

One of the most compelling reasons for utilizing AI in signature forgery detection is its ability to significantly improve security and reduce fraud. Forgery, whether it be in financial documents, legal contracts, or identity cards, can lead to serious financial losses and legal consequences. AI systems are trained on large datasets of

genuine and forged signatures, enabling them to learn and detect minute differences that human eyes might overlook. This includes variations in stroke pressure, speed, direction, and pattern consistency. As a result, AI-based systems can effectively identify not just basic forgeries, but also highly skilled imitations that are crafted to deceive traditional verification methods.

1.4Architecture and Mechanism

The architecture of the Signatures forgery detection system is designed to facilitate seamless interaction between the user interface, backend server, and the Siamese Neural Network model. The architecture can be divided into three main components.

Key Stages in detection of forged signatures

1. Web Interface: The web interface serves as the front end of the application, allowing users to interact with the system. It is built using HTML, CSS, and JavaScript, providing a user-friendly experience for uploading Signatures and viewing results.

Key features of the web interface include:

File Upload: Users can upload two Signatures images (an original and a suspected forgery) through file input fields. Image Preview: JavaScript is used to provide real-time previews of the uploaded images, enhancing user experience by allowing users to confirm their selections before submission. o Result Display: After processing, the results of the forgery detection are displayed on the same page, showing whether the suspected Signatures is forged or real, along with the images for comparison.

2. Backend Server: The backend server is implemented using Flask, a lightweight web framework for Python. It handles incoming requests from the web interface, processes the uploaded images, and interacts with the machine learning model.

Key functionalities of the backend server include:

Image Handling: The server receives the uploaded images, saves them temporarily, and prepares them for processing. Image Preprocessing: Before passing the images to the model, the server performs necessary preprocessing steps, such as resizing, normalization, and conversion to grayscale.

Model Inference: The server loads the trained Siamese Neural Network model and uses it to perform forgery detection by comparing the two images.

Response Generation: After processing, the server generates a response that includes the detection result and the paths to the uploaded images for display on the web interface.

3.Siamese Neural Network Model:

The Siamese Neural Network is the core component responsible for performing the actual forgery detection. It consists of two identical subnetworks that share weights and are designed to learn a similarity function between the two input images.

Key characteristics of the model include:

Feature Extraction: The model extracts features from both images using convolutional layers, enabling it to learn complex patterns and differences.

Distance Calculation: The model computes the Euclidean distance between the feature vectors of the two images, which serves as a measure of similarity. A lower distance indicates that the images are likely to be similar (i.e., the suspected Signatures is real), while a higher distance suggests forgery.

Training and Evaluation: The model is trained on a dataset of genuine and forged Signatures, allowing it to learn the distinguishing features that indicate forgery.

EXPERIMENTAL METHODOLOGY:

1. Data Collection

Objective: Obtain a dataset of genuine and forged signatures.

Sources:

o Public datasets like CEDAR, GPDS, or MCYT.

o Custom datasets collected via signature pads or mobile/tablet apps.

Types of signatures:

o Offline: Scanned images of handwritten signatures.

o Online: Digitally captured with time-series data like stroke pressure, speed, angle, and coordinates.

2. Data Preprocessing

Objective: Prepare the signature data for analysis and training.

Steps:

o Resizing & normalization of images.

o Noise reduction using filters (Gaussian, Median).

o Binarization (thresholding) for offline signatures.

o Feature extraction for online signatures (e.g., stroke order, velocity, pressure).

o Data augmentation (rotation, scaling, distortion) to improve model robustness.

3. Feature Extraction (for traditional ML approach)

Objective: Extract meaningful features from the signatures.
Techniques:
o Geometric features: height, width, aspect ratio.
o Statistical features: pixel distribution, histogram.
o Texture features: Local Binary Patterns (LBP), Gabor filters.
o Dynamic features (online): pressure, time, speed between points.
4. Model Selection
Objective: Choose appropriate AI/ML models for classification.
Approaches:
o Traditional Machine Learning:
• Support Vector Machine (SVM)
• Random Forest
• K-Nearest Neighbors (KNN)
o Deep Learning:
• Convolutional Neural Networks (CNN) for image-based (offline) signatures.
• Recurrent Neural Networks (RNN) or LSTM for online time-sequence signatures.
• Siamese Networks for signature verification (pairwise comparison).
• Autoencoders for anomaly detection in signature patterns.
5. Model Training
Objective: Train the AI model to distinguish between genuine and forged signatures.
Process:
o Split dataset into training, validation, and testing sets (e.g., 70/15/15).
o Use cross-validation to improve generalization.
o Optimize using techniques like Adam, SGD, or RMSprop.
o Use loss functions appropriate to the task:
• Binary Cross-Entropy (for classification)
• Contrastive Loss (for Siamese networks)
6. Evaluation Metrics
Objective: Assess the performance of the model.
Metrics:
o Accuracy
o Precision, Recall, and F1-Score
o False Acceptance Rate (FAR)
o False Rejection Rate (FRR)
o Equal Error Rate (EER) – where FAR and FRR are equal.
o ROC-AUC for classification performance.
7. Experimental Setup
Environment:
o Python with libraries like TensorFlow, PyTorch, Scikit-learn.
o Hardware: GPU for deep learning models.
Tools:
o Jupyter Notebooks for experimentation and visualization.
o OpenCV for image preprocessing.
o Matplotlib or Seaborn for performance plots.
8. Result Analysis
Compare performance across different models and feature sets.
Identify failure cases (e.g., skilled forgeries, poor-quality scans).
Analyze confusion matrices and ROC curves.
Evaluate performance on both seen and unseen users.
9. Conclusion and Future Work
Summarize key findings and best-performing models.
Discuss limitations (e.g., dataset size, generalizability).
Propose future improvements:
o Larger datasets
o Multimodal verification (signature + biometrics)
o Real-world deployment considerations

2.2. Related work

Digital forgery detection has been an ongoing topic in the research field for many years. Numerous solutions for the same have been proposed addressing different kinds of forgery detection. The existing document forgery detection methods can be broadly classified into main categories i.e active and passive methods. The paper [1] proposes a system architecture based on the inspection of probed documents with the analysis of ink. The paper produced a new method to find any mismatch of ink color in the HSD images. The approach is based on an NMF model with orthogonal as well as graph regularization. The assumption made here is that under some attainment protocols, some of the latent actors present in the HSD images can be forced to be orthogonal. The author of this paper also has proposed an efficient algorithm that is multiplier based to incorporate in the method The paper [2] proposes an efficient method to detect signature forgery which is

based on the Siamese neural network. The method uses CNN for data preprocessing and evaluation is done using a Siamese network working with the CNN model. Unique features of the implementation include a contrastive loss function. A high recall was achieved and loss was minimized to 0.43. The paper [3] proposes a robust system to detect digital forgery using CNN architecture for compressed images. The CNN architecture consists of multiple layers such as a pooling layer, a convolutional layer, and fully connected layers. The authors in [4] offer a shallow convolutional neural network (CNN) that can identify manufactured region boundaries from original edges in low-resolution images. SCNN was created to make use of chroma and saturation data. Two techniques based on SCNN, term sliding windows detection (SWD) and fast SCNN have been developed to detect and identify image forgery regions. The paper [5] provides a new deep learning-based image fraud detection system for automatically learning hierarchical representation from input RGB color photographs. Image splicing and copy-move forgeries can be detected using the suggested CNN, the fundamental high pass filter set employed in the spatial rich model (SRM) is utilized to establish the weights at the first layer of our network, which serves as a regularizer to efficiently suppress the effect of picture contents and capture the subtle artifacts created by the tampering operations. The pre-trained CNN is used as a patch descriptor to retrieve dense features from the test images, and the final discriminative features for SVM classification are obtained via a feature fusion technique.

Dataset:

Finding an appropriate dataset for training and testing the model is one of the challenging tasks while creating an ML model. Since training of the ML model almost depends on the dataset that has been used. Forged document images dataset is not available easily. We planned to create the dataset by collecting real document images from the web and forging the images to create the forged image dataset. The dataset consists of two types of data, one is for training copy move forgery detection model and the other is for training the signature forgery detection model. Since the dataset is created manually, it requires preprocessing. The data preprocessing is done using the OpenCV technique. Since the noise in the image has a great effect on the result of the models. Preprocessing includes slant correction and denoising. Since We have approached the forgery detection problem by creating different models for different types of intrinsic features, extraction of that particular feature is also needed. The signature forgery detection model accepts the extracted signature from the uploaded image.

2.3. Proposed Method

In this paper, we proposed a novel method for signature recognition and forgery detection. The proposed system architecture is shown in Fig. 6, where, test signature is recognized with the given input training set using both CNN and Crest-Trough method. Then forgery detection algorithms (Harris Algorithmic followed by Surf Algorithm) are enforced on this classified image

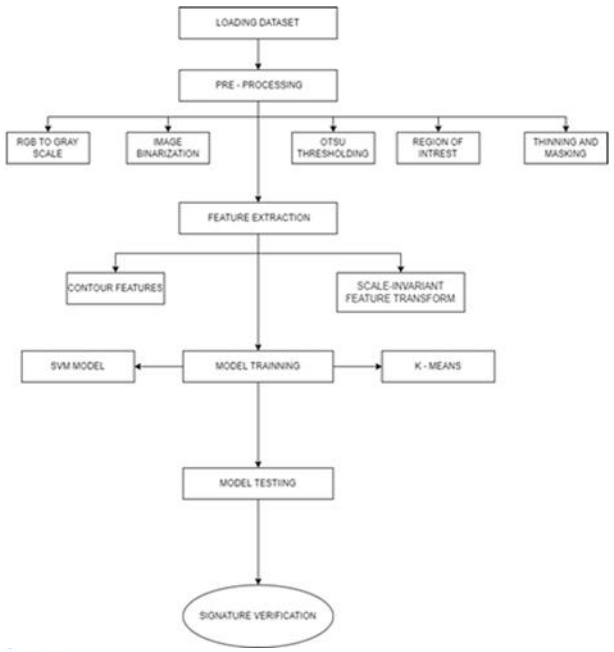


FIG: PROPOSED MODEL

3.1 Objectives for input and output design

Signature Recognition Convolutional Neural Networks (CNNs)[4] have tested no-hit in recent years at an outsized variety of image processing-based machine learning tasks. Several different strategies of playacting such tasks as shown in Fig. 7 revolve around a method of feature extraction, during which hand-chosen options extracted from a picture fed into a classifier to make a classification call. Such processes solely as sturdy because of the chosen options, which regularly take giant amounts of care and energy to construct. Against this, in CNN, the options fed into the ultimate linear classifier all learned from the dataset. A CNN consists of a variety of layers as shown in Fig. 7, beginning at the raw image pixels, that each performs an easy computation and feeds the result to the successive layer, with the ultimate

result being fed to a linear classifier. The layers computation area unit supports a variety of parameters that learned through the method of backpropagation, during which for every parameter, the gradient of the classification loss with relation to that parameter is computed and therefore the parameter is updated to minimize the loss perform. The look of any signature verification system typically needs the answer of 5 sub-issues: data retrieval, pre-processing, feature extraction, identification method, and performance analysis. Off-line signature verification just deals with pictures non-heritable by a scanner or a photographic camera. In associate degree off-line signature verification system, a signature is non-heritable as a picture. This picture depicts a private sort of human. The method needs neither be too sensitive nor too rough. It should have a proper balance between an occasional False Acceptance Rate (FAR) and an occasional False RejectionRate(FRR).

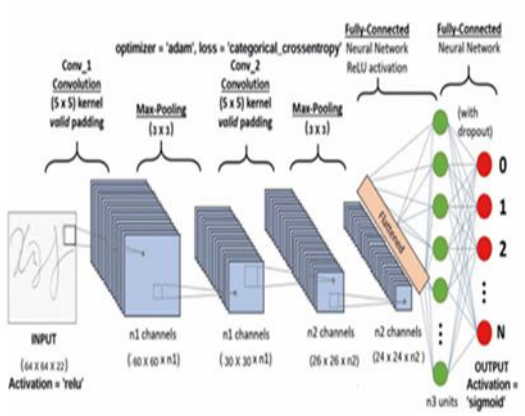


Fig: CNN Architechture

Signature Forgery Detection When the image is finally classified into one among the present classes of the topics. The henceforward system proposes a technique for the detection of the forgery within the same by checking the chosen options against the on the market set of pictures for the signature and produces a binary result if it's cast or not. Here two Algorithms are used for Forgery Detection. 4.2.1. Harris Algorithm Avoid hyphenation at the end of a line. Symbols denoting vectors and matrices should be indicated in bold type. Scalar variable names should normally be expressed using italics. Weights and measures should be expressed in SI units. All non-standard abbreviations or symbols must be defined when first mentioned, or a glossary provided [13]. The Harris corner detection algorithm is based on formula

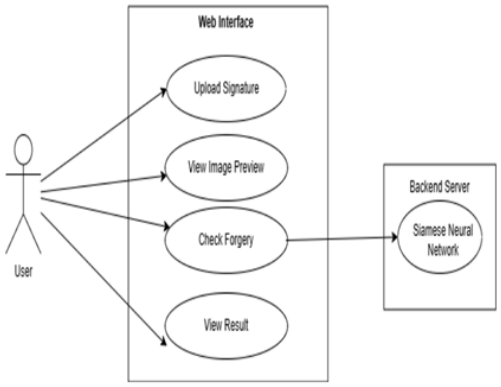
Where: • E is the separation between the first and affected window. • u, v is the displacement of the frame within the x-direction and y-direction respectively. • w (x, y) is the frame at point (x, y). This acts sort of a mask which assures that solely the marked window is working. that the intensity of the image at a point (x, y). • I (x+u, y+v) is the intensity of the considered frame. • I (x, y) is the intensity of the first. Features: • Corner points detected from training and take a look at knowledge. • Corner points extracted from training and take a look at knowledge. • Take a look at knowledge compared with each training knowledge. 4.2.2. Surf Algorithm Speeded up robust features (SURF) [14] uses square-shaped filters for approximation of Gaussian smoothing. (The SIFT approach uses cascaded filters to observe scale-invariant characteristic points, wherever the Difference of Gaussians (DoG) is calculated on rescaled pictures more and more.) Filtering the image with a sq. is far quicker if the integral image is used. The SURF algorithm is based on formula

$$S(x,y) = \sum_{i=0}^{x}\sum_{j=0}^{y} I(i,j)$$

Features: • Index points detected from training and take a look at knowledge. • Index points extracted from training and take a look at knowledge. • Take a look at points compared with each training knowledge

Use case diagram:

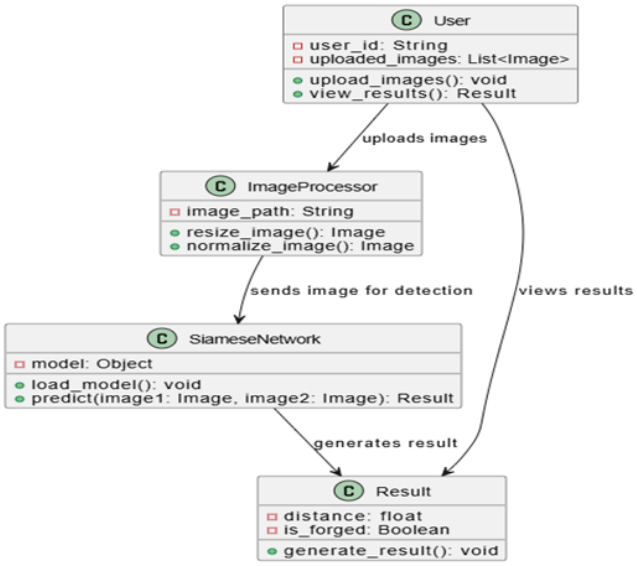
A use case diagram is a visual representation of how users (or actors) interact with a system to achieve specific goals. It's a high-level overview that shows the system's functionality and how it's used by different actors, making it easier for stakeholders to understand the system's behavior.



Class diagram:

A class diagram is a visual representation that shows the classes within a system, their attributes, methods, and the relationships between them. It's a type of static

structure diagram in the Unified Modeling Language (UML). Class diagrams are essential for understanding and documenting the structure of object-oriented systems.

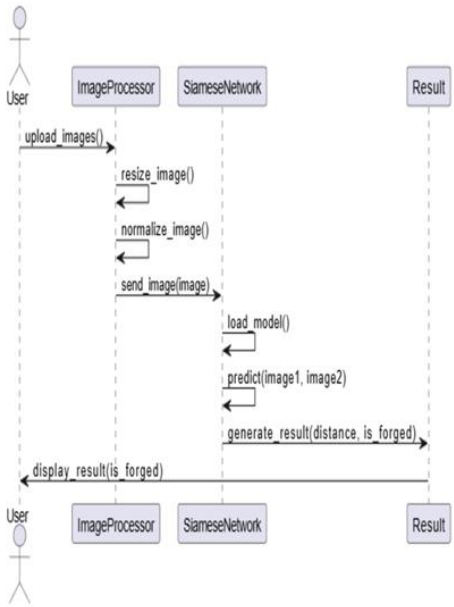


Activity diagram:

An activity diagram is a type of Unified Modeling Language (UML) flowchart that visually represents the flow of actions or processes within a system. It's essentially a way to map out the sequence of steps involved in a task or process, highlighting decision points, parallel activities, and loops.

Sequence Diagram:

A sequence diagram, a type of interaction diagram in Unified Modeling Language (UML), visually depicts the sequence of interactions between objects in a system over time. It shows the order in which messages are exchanged between objects, essentially illustrating how objects communicate and cooperate to achieve a specific task or process.



3.2. RESULTS & DISCUSSION

The training images for the copy-move model, CASIA-2 and MICC dataset have been considered. The dataset consists of images from these two datasets which are split into training and testing datasets and then passed onto our model to classify them into two classes i.e authenticate and forged.

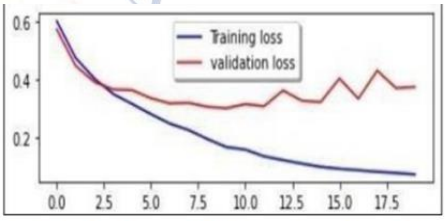


Fig 2: Training loss and Validation Loss of Copy Move Forgery Detection Model

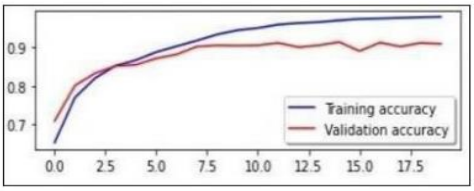


Fig 3: Training Accuracy and Validation Accuracy of Copy Move Forgery Detection Model

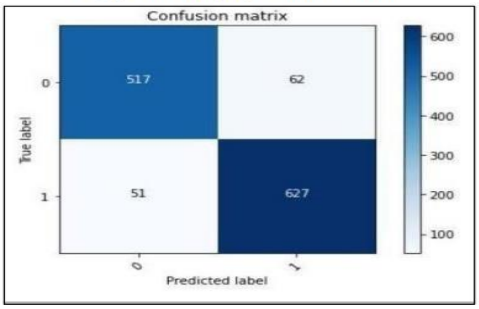


Fig 4: Confusion Matrix of Copy Move Forgery Detection Model

4.RESULTS

Firstly, the test signature is recognized with the given input training set using both CNN and Crest-Trough method. Then forgery detection algorithms (Harris Algorithmic followed by Surf Algorithm) are enforced on this classified image. The Results from each the algorithms are then compared as shown in Fig. 8 and Fig. 9 respectively. The popularity associated with identification with neural networks yields an accuracy of 94%. The latter planned forgery detection works with associate accuracy of 85-89%. The sole skilled and closely solid signatures typically don't seem to be captured, else it properly identifies all the forgeries within the signature.

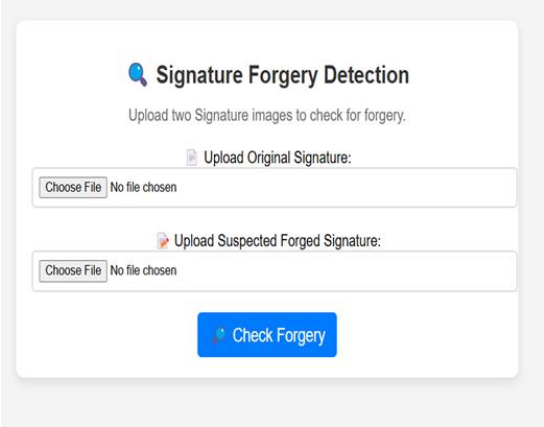


Fig: initial page

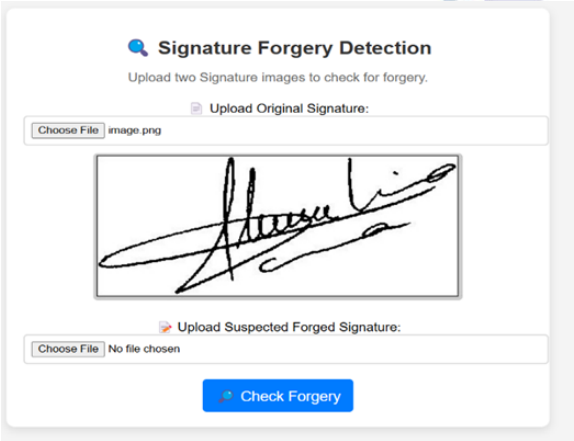


Fig: Upload signatures files

After the user uploads the signature files, the system processes the images to determine whether the signature is genuine or forged. The uploaded signature undergoes preprocessing steps such as resizing, grayscale conversion, and noise removal to enhance image quality. Key features of the signature, including stroke patterns, curves, and pressure points, are then extracted using image processing techniques or deep learning models like Convolutional Neural Networks (CNNs). These features are compared with reference signatures stored in the system to assess similarity. Based on this analysis, the system classifies the signature as either "Genuine" or "Forged" and displays the result to the user, often along with a confidence score that indicates the accuracy of the prediction.

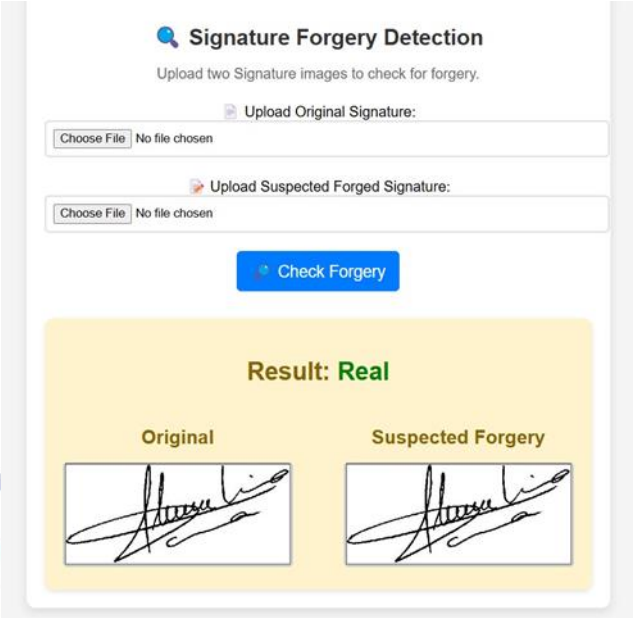


Fig: Result Real

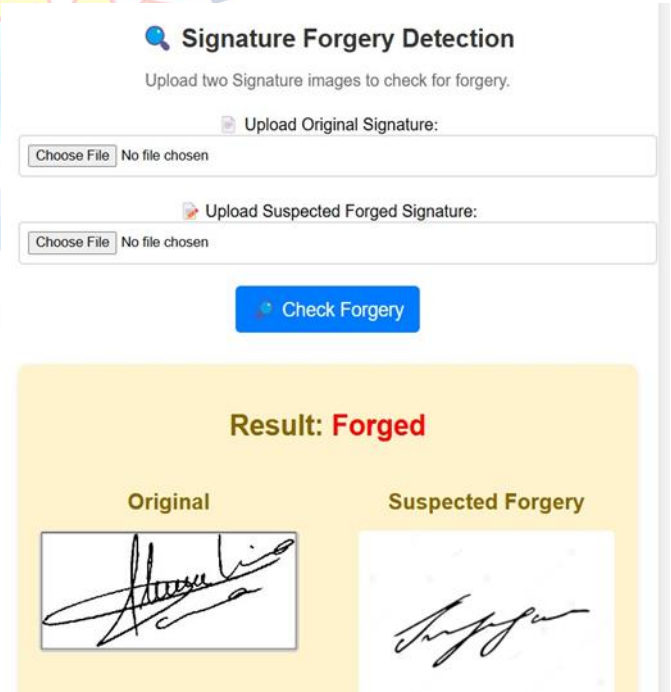


Fig: Forgery Detected

5.SUMMARY AND CONCLUSIONS

In conclusion, Signatures forgery detection is an essential task across various sectors, including banking, legal, and academic fields. The integrity of Signatures is crucial for maintaining trust and security in transactions and communications. Traditional methods of forgery

detection often rely on manual inspection by trained professionals, which can be time-consuming, subjective, and prone to human error. As a result, there is a growing need for automated systems that can efficiently and accurately detect forged Signatures. Furthermore, Signature forgery detection methods suffer from limitations in accuracy, scalability, and adaptability to advanced fraud techniques. These methods are often unreliable when dealing with high-quality forgeries, making them inadequate for real-world applications. In essence, these systems leverage algorithms that can analyze visual features in Signatures, making them capable of identifying subtle discrepancies that may indicate forgery. This project focuses on developing a robust Signatures forgery detection system utilizing a Siamese neural network model. The system is designed to analyze and compare two Signatures images: an original Signatures and a suspected forged Signatures. By leveraging deep learning techniques, the system aims to provide accurate and reliable results, assisting users in identifying potential forgery.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] Alameri, M.A.A.; Ciylan, B.; Mahmood, B. Computational Methods for Forgery Detection in Printed Official Documents. In Proceedings of the 2022 ASU International Conference in Emerging Technologies for Sustainability and Intelligent Systems (ICETSIS), Virtual, 22–23 June 2022; pp. 307–331.

[2] Montasari, R.; Hill, R.; Parkinson, S.; Peltola, P.; Hosseinian-Far, A.; Daneshkhah, A. Digital forensics: Challenges and opportunities for future studies. Int. J. Organ. Collect. Intell. 2020, 10, 37–53.

[3] Dyer, A.G.; Found, B.; Rogers, D. An insight into forensic document examiner expertise for discriminating between forged and disguised signatures. J. Forensic Sci. 2008, 53, 1154–1159.

[4] Parkinson, A.; Colella, M.; Evans, T. The development and evaluation of radiological decontamination procedures for documents, document inks, and latent fingermarks on porous surfaces. J. Forensic Sci. 2010, 55, 728–734.

[5] Ragai, J. Scientist And The Forger, The: Insights Into The Scientific Detection Of Forgery In Paintings; World Scientific: Singapore, 2015.

[6] Warif, N.B.A.; Wahab, A.W.A.; Idris, M.Y.I.; Ramli, R.; Salleh, R.; Shamshirband, S.; Choo, K.-K.R. Copy-move forgery detection: Survey, challenges, and future directions. J. Netw. Comput. Appl. 2016, 75, 259–278.

[7] Valderrama, L.; Março, P.H.; Valderrama, P. Model precision in partial least squares with discriminant analysis: A case study in document forgery through crossing lines. J. Chemom. 2020, 34, e3265.

[8] Niu, P.; Wang, C.; Chen, W.; Yang, H.; Wang, X. Fast and effective Keypoint-based image copy-move forgery detection using complex-valued moment invariants. J. Vis. Commun. Image Represent. 2021, 77, 103068.

[9] Muthukrishnan, R.; Radha, M. Edge detection techniques for image segmentation. Int. J. Comput. Sci. Inf. Technol. 2011, 3, 259.

[10] Gorai, A.; Pal, R.; Gupta, P. Document fraud detection by ink analysis using texture features and histogram matching. In Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July 2016.

[11] Markiewicz-Keszycka, M.; Cama-Moncunill, X.; Casado-Gavalda, M.P.; Dixit, Y.; Cama-Moncunill, R.; Cullen, P.J.; Sullivan, C. Laser-induced breakdown spectroscopy (LIBS) for food analysis: A review. Trends Food Sci. Technol. 2017, 65, 80–93.

[12] Elsherbiny, N.; Nassef, O.A. Wavelength dependence of laser-induced breakdown spectroscopy (LIBS) on questioned document investigation. Sci. Justice 2015, 55, 254–263.

[13] Raman Spectroscopy of Two-Dimensional Materials; Tan, P.H. (Ed.) Springer: Berlin/Heidelberg, Germany, 2018.