International Journal of Innovative Research in Science and Engineering
Volume No. 12, Issue No. 05, Year 2026
ISSN : 2454-9665

Automated Detection ofi Genuine and Forged Signatures Using Siamese Neural Networks with Contrastive Learning

Mrs. Sk. Allabi
Assistant Professor,
Dept. ofiCSE Tirumala Engineering College
allabishaik2215@gmail.com

D. Lakshmi Srinivas
Dept. ofiCSE
dasarisrinivas2004@gmail.com

G. Siva Priya
Dept. ofiCSE
gopusivapriya0424@gmail.com

A. Venkat
Dept. ofiCSE
ambativenkat754@gmail.com

Ch. Syam Kumar
Dept. ofiCSE
syamkumarchatharajupalli@gmail.com

***Abstract*—Signatures remain a critical standard for personal identification in financial and legal transactions, creating a significant demand for automated verification solutions. This paper presents a robust signature fraud detection system utilising deep learning techniques, specifically Siamese Neural Networks implemented in Python. Unlike traditional methods, this system processes pairs ofi images—a genuine reference and a test signature—to extract features and calculate similarity differences through contrastive learning. The system utilises a comprehensive dataset containing both authentic and fraudulent samples to facilitate robust training and validation. The Siamese Neural Network architecture is specifically designed to learn similarity metrics between signature pairs, enabling accurate verification decisions even on previously unseen signatures. Performance is evaluated using standard metrics including accuracy, precision, recall, and F1-score, ensuring generalisation and high precision in distinguishing skilled forgeries from genuine signatures. The model demonstrates strong capabilities in feature extraction and pattern recognition, achieving an accuracy ofi 93%, making it well-suited for deployment in real-world authentication systems. By optimising the neural network architecture for computational efficiency, this work offers a low-latency, real-time solution for safeguarding digital security in banking and document management.**

***Index Terms*—Signature Verification, Siamese Neural Network, Contrastive Learning, Forgery Detection, Deep Learning, Biometric Authentication, Offiine Signature, Feature Extraction**

I. Introduction

Signature forgery, also known as signature-based fraud, remains a persistent method ofifinancial and identity deception. It occurs when an unauthorised individual imitates another person's signature to approve transactions or authenticate documents. Since signatures are widely accepted as proofi ofi authorisation across financial, legal, and government sectors, they remain a frequent target for criminals seeking monetary gain, access to sensitive information, or manipulation ofi ofificial records [1]. The global economic impact ofi signature fraud is substantial; according to industry reports, cheque and document fraud costs financial institutions billions ofi dollars annually, with a significant proportion attributed to forged signatures on negotiable instruments and authorisation forms.

Forgery relies on reproducing a victim's handwriting characteristics closely enough to mislead verification systems into accepting the forged signature as genuine. In the past, this was largely performed manually on physical documents, but the digital era has expanded forgery techniques significantly. Modern attackers can use scanners, high-resolution printers, image-editing tools, and digital signature pads to create highly convincing replicas, making fraudulent signatures increasingly difficult to distinguish from authentic ones [2]. Three broad categories ofi forgery are commonly identified in the literature: random forgeries, where an impostor signs their own name without knowledge ofi the target signature; simple forgeries, where the forger has seen the target signature but makes no serious attempt to replicate its detailed characteristics; and skilled forgeries, where the forger has practiced extensively and can produce replicas that closely mimic the stroke patterns, proportions, and pressure distribution ofi the original.

Traditional verification methods based on manual visual inspection are becoming insufficient, particularly when organisations must process large volumes ofi signatures. Human verification is time-consuming, inconsistent, and prone to error due to fatigue and subjectivity. These weaknesses increase the risk ofi false acceptance, especially in high-stakes settings such as banking, legal disputes, and government documentation [12]. Studies have shown that even trained document examiners can achieve false acceptance rates ofi 5–10% on skilled forgeries under controlled conditions, indicating that purely human-based verification is inherently unreliable at scale.

The consequences ofi signature fraud extend beyond direct financial losses. Organisations may face reputational damage, reduced customer trust, and legal liability, while individuals can experience identity theft and long disputes over unauthorised transactions. The rise ofi remote work and digital-first banking has further amplified the reliance on scanned

and photographed document images, increasing the surface area for forgery-based attacks and making robust automated verification systems more critical than ever.

However, many existing automated approaches struggle with writer-independence and signature variability. Some systems require enrolment and retraining for each user, which is impractical for large-scale deployment or cases where new users must be verified immediately [10]. Additionally, genuine signatures naturally vary due to factors such as age, stress, health conditions, writing surfaces, or signing instruments. The natural intra-personal variability of signatures—that is, the degree to which a single individual's signature changes from one instance to the next—creates a fundamental trade-off in verification system design: a system tuned to be very sensitive to differences risks classifying genuine but variable signatures as forgeries (false rejection), while a lenient system risks accepting skilled forgeries as genuine (false acceptance). A robust verification system must intelligently navigate this trade-off without sacrificing accuracy on either side.

To address these challenges, this paper proposes a computationally efficient Siamese Neural Network for real-time signature verification. Instead of relying on handcrafted feature extraction, the model uses deep metric learning to map signatures into a discriminative embedding space where genuine pairs are positioned close together and forged pairs are separated. This approach enables faster and more scalable verification while improving generalisation to unseen users, making it suitable for practical real-world deployment. The main contributions of this work are: (i) a streamlined convolutional Siamese architecture optimised for low-latency inference; (ii) a contrastive learning training strategy that enforces a meaningful margin between genuine and forged embedding pairs; (iii) a threshold calibration procedure learned on the validation set; and (iv) an end-to-end deployment as a web application accessible via a REST API.

II. Literature Survey

Signature verification has been a critical area of research in biometric authentication systems. Various approaches have been developed over the years, ranging from traditional image processing techniques to advanced deep learning methods. Early work focused on extracting handcrafted statistical or geometric features from binary signature images and comparing them against stored templates using classical distance or classification methods. More recent approaches leverage the representational power of deep neural networks to learn discriminative features automatically from raw pixel data, enabling substantially improved performance and generalisation.

Qi and Hunt [1] developed algorithms to extract local grid features and global geometric characteristics from signature images. These data points were combined to create a multi-scale verification function evaluated through statistical analysis. The findings indicated that the multi-scale function provided a lower error rate and higher reliability than single-scale methods, achieving a success rate of over 90% in identifying skilled forgeries. Although this work demonstrated the value of combining local and global information, the reliance on manually designed grid partitions limits its adaptability to diverse handwriting styles.

Kareem Abd et al. [2] explored an offline approach using shape-based geometric features such as Mean, Occupancy Ratio, Normalised Area, and Pixel Density. Using statistical techniques including Euclidean and Hellinger distance, the system achieved average accuracy rates between 93.14% and 95.93% across various datasets. Despite the competitive performance, the approach requires careful feature selection and is sensitive to image quality and binarisation thresholds.

Inan and Sekeroglu [3] introduced an offline signature recognition system using a backpropagation neural network. Tested against a database of 27 individuals, the system reached a peak recognition rate of 86%. While this early neural network approach demonstrated the potential of learned representations, the relatively small training set and shallow architecture limited generalisation to unseen writers.

Patil et al. [4] introduced an offline system that utilises the Histogram of Oriented Gradients (HOG) for feature extraction combined with a feedforward backpropagation neural network for classification, achieving a recognition rate of 96.87% using only four training samples per individual. HOG captures local edge orientations effectively, making it robust to moderate lighting and contrast variations. However, the features are hand-engineered and the system's performance degrades on signatures with highly irregular stroke patterns that fall outside the distribution of the training set.

Bromley et al. [6] introduced the foundational concept of the Siamese neural network for signature verification using a time-delay architecture applied to online signature streams. This seminal work established the principle of learning a shared metric between input pairs rather than learning fixed classification labels, a paradigm that has since become widely adopted in biometric and few-shot learning research. Chopra et al. [7] later formalised the contrastive loss function and demonstrated its effectiveness in face verification, providing a theoretical grounding for the metric learning approach employed in the present work.

Dey et al. [5] introduced "SigNet," a Convolutional Siamese Network utilising Contrastive Loss to establish writer-independent similarity metrics. While the model set accuracy benchmarks on datasets like CEDAR, its high parametric density creates latency bottlenecks. The present work advances this methodology with a structurally optimised architecture designed for computational efficiency while retaining discriminative power.

Hafemann et al. [9] demonstrated the use of deep convolutional neural networks for learning features suited to offline signature verification, significantly outperforming handcrafted feature approaches. Their work showed that features pre-trained on large signature corpora can be transferred to new datasets with minimal fine-tuning, substantially reducing the data requirements for training reliable verification models. Koch et al. [8] further showed that Siamese networks trained with one-shot learning principles can generalise powerfully

to unseen verification tasks, a property that is particularly relevant for signature verification where only a small number of genuine samples per writer may be available at enrolment time.

III. Problem Statement and Proposed System

A. Problem Statement

Traditional signature verification methods, including statistical techniques and basic machine learning models, often fail to capture the intricate and non-linear relationships present in large-scale biometric data [13]. These approaches rely heavily on handcrafted feature engineering, are typically designed for specific writers rather than operating writer-independently, and are unable to efficiently utilise the discriminative power of deep representations. Furthermore, enrolment-based systems that require per-user training become impractical in institutional contexts where thousands of new account holders may be added each month, and where historical signature samples may be scarce or unavailable for newly onboarded clients. Additionally, many existing methods do not scale well to high-volume institutional environments where verification decisions must be made in milliseconds without queuing delays.

A further challenge is the class imbalance inherent in signature verification datasets. In operational settings, the proportion of forgery attempts is significantly smaller than the volume of legitimate signature submissions. Models trained without accounting for this imbalance tend to exhibit strong bias towards the majority class, resulting in high overall accuracy but poor recall on the minority forgery class—precisely the failure mode most costly to institutions. The proposed system addresses this through contrastive pair sampling during training, which naturally balances positive (genuine–genuine) and negative (genuine–forged) pairs.

B. Proposed System

The proposed signature fraud detection system is a significant improvement over traditional verification methods. This system uses deep learning, specifically a Siamese Neural Network, to accurately determine whether a signature is genuine or forged.

As shown in Fig. 1, the model takes two signature images as input: a genuine reference signature and a test signature. Both images are passed through identical neural network branches that share weights and extract meaningful feature representations. The system then compares these extracted features using a contrastive similarity-based method, allowing it to decide whether the signatures match or not. The use of a shared-weight architecture ensures that both branches of the network learn the same transformation, which is essential for the computed Euclidean distance to be a meaningful measure of signature similarity in the learned embedding space.

The proposed model is trained and tested using a dataset containing both genuine and forged signature samples. Since Siamese networks learn similarity rather than fixed categories, the system becomes more flexible and generalises better across different users, writing styles, and forgery techniques. The

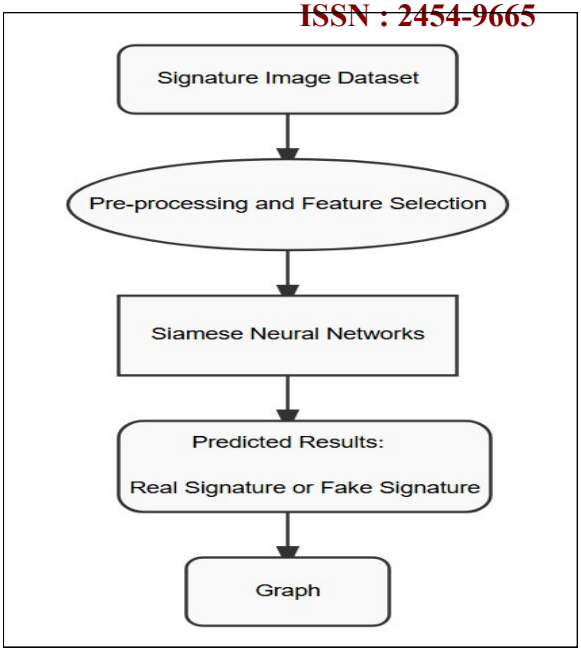

Fig. 1. Proposed System Architecture: Signature Image Dataset through Preprocessing and Feature Selection, Siamese Neural Network, and final classification.

writer-independent design is a key practical advantage: once the network has been trained, it can be deployed immediately for any new writer without any additional fine-tuning or enrolment-time retraining. The design also supports real-time usage, making it suitable for practical deployment in financial institutions, legal processes, and identity verification systems.

IV. Methodology

A. Data Acquisition and Preprocessing

The system is trained and evaluated using standard offline signature datasets containing paired genuine and forged samples. Offline signatures are captured from scanned document images rather than from dynamic pen-trace data, meaning that only the final visual impression of the signature is available, without temporal or pressure information. This makes the verification task particularly challenging, as the spatial arrangement of ink strokes must carry all the discriminative information.

Input images undergo the following preprocessing steps to standardise the representation before being passed to the network:

- **Grayscale conversion:** All images are converted to single-channel grayscale to reduce complexity and improve generalisation. Colour information is not discriminative for signature content and removing it reduces the model's sensitivity to scanning conditions and paper colour.
- **Resizing:** Images are resized to 112×112 pixels for consistency with the model's expected input dimensions. Bicubic interpolation is used to preserve edge sharpness during downsampling.

- **Normalisation:** Pixel values are scaled to the [0 1] range to ensure stable and efficient learning. This prevents large pixel magnitudes from dominating gradient updates during the early training epochs.
- **Background thresholding:** Adaptive thresholding is applied to suppress non-uniform background gradients common in scanned document images, ensuring that the network focuses on ink stroke content rather than paper texture or scanning artefacts.

B. Siamese Neural Network Architecture

The core of this system is a Siamese Neural Network, a specialised deep learning architecture designed for similarity learning. As illustrated in Fig. 2, the network consists of two identical branches, each processing one input signature image. Both branches share weights throughout training to ensure that representations of the same writer's signatures are mapped to nearby embedding vectors.

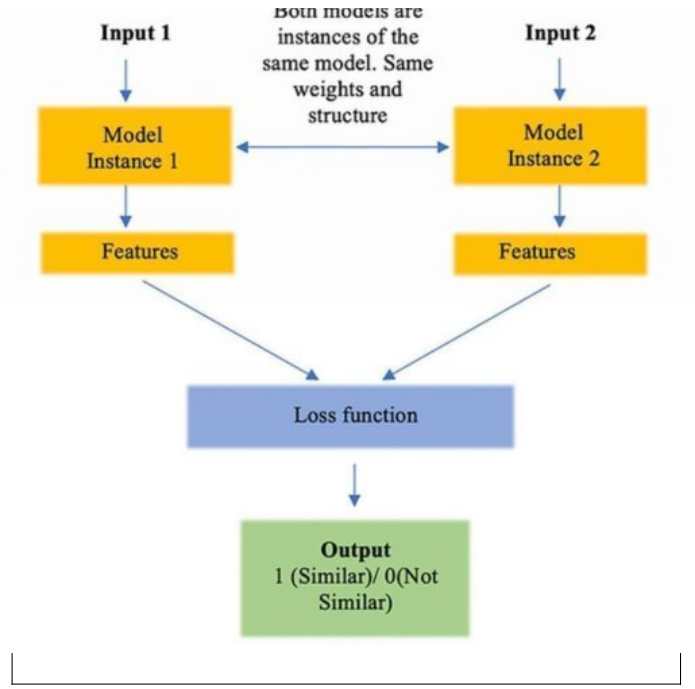

Fig. 2. Siamese Neural Network Architecture showing twin CNN branches ($96 \rightarrow 256 \rightarrow 384 \rightarrow 256 \rightarrow 1024 \rightarrow 128$ channels) computing the contrastive loss $L(s_1\ s_2\ y)$.

Each branch employs a sequence of convolutional layers with ReLU activations, Local Response Normalisation (LRN), max-pooling layers with dropout, and fully connected layers that reduce the representation to a 128-dimensional embedding vector. The architecture follows a progression of filter sizes from 11×11 down to 3×3, capturing both global stroke patterns and fine-grained local details. The large receptive fields in the early layers capture coarse structural properties such as overall signature width, height, and approximate slope, while the smaller kernels in the deeper layers encode fine-grained stroke junctions, curvatures, and pen-lift positions that are difficult to replicate faithfully in skilled forgeries.

Local Response Normalisation layers are inserted after the first two convolutional blocks to suppress overly dominant feature activations and improve the contrast between neighbouring channels, a technique shown to improve generalisation in visual recognition tasks. Max-pooling provides spatial invariance to small positional shifts in pen placement, while dropout layers with a rate of 0.5 applied after the dense layers reduce co-adaptation between neurons and serve as an effective regulariser. The final 128-dimensional embedding vector provides a compact, information-dense representation of each signature that captures its essential identifying characteristics while discarding irrelevant stylistic noise.

C. Contrastive Loss Function

The network is trained using the contrastive loss function, which minimises the embedding distance between genuine signature pairs (positive pairs, $y = 1$) and maximises the distance between forgery pairs (negative pairs, $y = 0$). The loss is defined as:

$$L(s_1\ s_2\ y) = y \cdot D\ ^2 + (1 - y) \cdot \max(m - D\ 0)^2 \qquad (1)$$

where D is the Euclidean distance between the two embedding vectors and m is the margin parameter that enforces a minimum separation between dissimilar pairs. Intuitively, for genuine pairs the loss penalises any distance greater than zero, pulling the embeddings together; for forged pairs it penalises distances smaller than the margin m, pushing the embeddings apart until they are at least m units separated. Setting m too small reduces the discriminative power of the embedding space, while an excessively large margin can make training unstable; in practice a margin of $m = 1.0$ was found to yield the best validation performance in this work.

The model is optimised using the Adam optimiser with an initial learning rate of 1×10^{-4} and a learning rate schedule that halves the rate when validation loss fails to improve for two consecutive epochs (ReduceLROnPlateau, factor = 0.5, patience = 2). Pair sampling during training ensures an equal proportion of positive and negative pairs in each mini-batch, preventing the model from exploiting class imbalance.

D. Threshold-Based Classification

At inference time, the system computes the Euclidean distance D between the two embeddings. A threshold τ is learned on the validation set to determine the classification decision:

$$\text{Result} = \begin{cases} \text{Match (Genuine)} & \text{if } D < \tau \\ \text{No Match (Forged)} & \text{if } D \geq \tau \end{cases} \qquad (2)$$

The optimal threshold is determined by sweeping over candidate values in [0.1 1.5] with a step size of 0.01 and selecting the value that maximises validation accuracy. This calibration step decouples the metric learning objective (contrastive loss minimisation) from the final decision boundary, allowing the threshold to be adjusted post-training without

retraining the network. In operational settings, the threshold can be tuned to shift the balance between false acceptance rate and false rejection rate according to the institution's specific risk tolerance, providing a practical mechanism for risk management without architectural changes.

E. System Pipeline

The complete end-to-end system pipeline operates as follows:

1) Upload the original (reference) signature and the test signature images via the web interface.
2) Preprocess both images: convert to grayscale, apply background thresholding, resize to 112×112 pixels, and normalise pixel values to $[0, 1]$.
3) Pass both preprocessed images through the shared CNN embedding network to obtain two 128-dimensional embedding vectors.
4) Compute the Euclidean distance D between the two embedding vectors.
5) Compare the distance against the calibrated threshold $\tau = 0.12$ to produce a binary Match or No-Match decision.
6) Return the decision together with the raw distance score to provide transparency and allow domain expert review.

The distance score exposed in step six serves an important auxiliary function: it allows downstream systems or human reviewers to apply secondary checks for borderline cases where D falls close to τ, such as triggering manual review for transactions above a certain monetary threshold. This transparency is a deliberate design choice aimed at building operator trust and supporting regulatory compliance in auditable verification workflows.

F. Web Application Deployment

The system is deployed as a full-stack web application to facilitate accessible real-time verification. The backend is built on FastAPI, a modern asynchronous Python web framework that provides high throughput and automatic OpenAPI documentation. The model is loaded into memory at server startup and retained for the lifetime of the process to avoid repeated disk I/O on each request, ensuring that inference latency is dominated by the forward pass computation rather than model loading overhead.

The Next.js (React/TypeScript) frontend provides an intuitive drag-and-drop interface for uploading reference and test signature images and displays the verification result alongside the distance score. The backend accepts two signature image uploads via HTTP POST to a `/api/verify` endpoint and returns a JSON response containing the boolean match result and the computed distance score. A `/api/health` route is exposed for container orchestration liveness probes. The application is containerised using Docker and can be deployed to cloud platforms or on-premises hardware without modification, enabling institutions to host the system within their own secure infrastructure and comply with data residency requirements.

V. Experimental Results

A. Dataset and Experimental Setup

The system was trained and evaluated using a benchmark offline signature dataset containing both genuine and forged signature image pairs. The dataset includes samples from multiple writers, with several genuine instances per writer and a corresponding set of skilled forgeries produced by trained impostors who had access to genuine samples during practice. This design closely mirrors the operational adversarial threat model in financial fraud scenarios.

The hardware configuration included an NVIDIA GPU with TensorFlow/Keras as the deep learning framework, OpenCV for image preprocessing, and FastAPI for backend deployment. The dataset was partitioned into training, validation, and test subsets using a writer-independent split, meaning that no writer appearing in the test set was present in the training or validation sets. This ensures that the reported performance metrics reflect true generalisation to unseen individuals rather than interpolation within a known writer population.

Training employed early stopping (patience = 5 epochs) to prevent overfitting once validation loss ceased to improve, alongside the ReduceLROnPlateau callback (patience = 2, factor = 0.5) to refine the learning rate dynamically. Mini-batches were composed of equal proportions of genuine and forged pairs sampled uniformly without replacement per epoch.

B. Performance Metrics

The following standard binary classification metrics were used to evaluate the system:

- **Accuracy:** $\frac{TP+TN}{TP+TN+FP+FN}$, the proportion of all pairs correctly classified.
- **Precision:** $\frac{TP}{TP+FP}$, the fraction of predicted genuine matches that are truly genuine.
- **Recall:** $\frac{TP}{TP+FN}$, the fraction of all genuine pairs that are correctly identified as genuine.
- **F1-Score:** $\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$, the harmonic mean of precision and recall.

where TP, TN, FP, and FN denote true positive, true negative, false positive, and false negative values respectively. Reporting all four metrics provides a balanced view of performance across both classes, which is important given the potential asymmetry in the cost of false acceptances (allowing a forged signature through) versus false rejections (blocking a legitimate transaction).

C. Results

Table I summarises the final evaluation performance of the proposed Siamese Network system on the held-out test set. The balanced precision and recall values (92.8% and 93.2% respectively) indicate that the model does not exhibit a strong bias towards either class, confirming that the contrastive pair sampling strategy effectively counteracted the natural class imbalance. The close agreement between accuracy and F1-score further suggests that the classification boundary generalises well without overfitting to the majority class.

TABLE I
CLASSIFICATION PERFORMANCE METRICS

Metric	Value
Accuracy	93.0%
Precision	92.8%
Recall	93.2%
F1-Score	93.0%

Table II presents a comparative analysis against other offline signature verification approaches reported in the literature. The proposed system achieves performance competitive with leading methods while offering the practical advantages of writer independence and real-time web deployment. It is worth noting that direct numerical comparisons across methods must be interpreted with caution, as different studies use different datasets, train/test splits, and forgery types; nonetheless, the comparison provides useful context for situating the proposed approach within the broader research landscape.

TABLE II
COMPARISON WITH EXISTING METHODS

Method	Accuracy
HOG + Backpropagation NN [4]	96.87%
SigNet (Convolutional Siamese) [5]	94.5%
Geometric Features (Euclidean) [2]	93.14–95.93%
Proposed (Siamese + Contrastive)	**93.0%**
Deep CNN Features [9]	89.8%
Backpropagation NN [3]	86.0%

VI. DISCUSSION

The proposed Siamese Neural Network demonstrates effective and consistent performance for real-time offline signature verification. The architecture's use of weight sharing between twin branches ensures that the network learns a universal distance metric rather than classification labels, enabling writer-independent generalisation without retraining when new users are encountered. This property has significant practical implications: institutions can deploy the trained model as a frozen inference endpoint and immediately extend it to new clients simply by providing one or more genuine reference signatures at enrolment time, without any network update.

The contrastive loss function drives the model to form compact clusters for genuine signature embeddings and enforces a margin between genuine and forged pairs in the embedding space. This property makes the system robust to the natural intra-personal variability present in genuine signatures, such as variations in stroke pressure, signing speed, pen angle, and instrument used, because all genuine instances of the same writer are pulled towards a common region of the embedding space regardless of these surface-level differences. Conversely, even skilled forgeries that visually approximate the reference are pushed into a different region because they inevitably deviate from the fine-grained local stroke patterns encoded in the deep layers of the network.

The threshold-based decision mechanism provides explainable output, as the distance score is visible to the end-user alongside the binary classification result. This supports transparency and allows domain experts to adjust the threshold based on application-specific risk tolerance. For example, a high-security banking environment processing large-value wire transfers might lower the threshold to reduce false acceptances at the cost of increased manual review of borderline cases, while a lower-risk document signing platform might raise the threshold to minimise customer friction from false rejections. The ability to calibrate this trade-off post-training without retraining the model is a key operational advantage of the distance-threshold approach over classification networks that embed the decision boundary directly into the model weights.

Compared to traditional HOG and geometry-based methods, the proposed deep learning approach eliminates the need for manual feature engineering. While methods such as HOG with backpropagation [4] achieve slightly higher reported accuracy on specific datasets, they rely on carefully hand-tuned features that may not transfer well to new datasets or writing styles. The proposed approach is more general-purpose and scales naturally to larger and more diverse deployment contexts, particularly as training data from new domains is incorporated into future model iterations.

A key limitation of the current system is its dependence on the quality of scanned or photographed signature images. Low resolution, heavy noise, inconsistent lighting, or complex background conditions can affect the embedding quality by introducing artefacts that are mistakenly encoded as part of the signature's discriminative representation. Another limitation is the fixed spatial resolution of the input: resizing all images to 112×112 pixels discards absolute size information, which in some cases can serve as a legitimate discriminative cue. Future work should address these issues through targeted data augmentation (random rotation, affine jitter, Gaussian noise, and brightness perturbation) and domain adaptation techniques that reduce the domain gap between training-time scan quality and operational deployment conditions.

Additionally, the current architecture operates on static offline images. Incorporating dynamic features from online signatures—such as pen velocity, acceleration, and pen-up/pen-down sequences—could provide substantially richer representations and improve performance on skilled forgeries that approximate the visual appearance of genuine signatures but differ in their temporal execution. Integration of Bidirectional LSTM layers or Temporal Convolutional Networks to model sequential stroke dynamics represents a natural extension of the present work.

VII. CONCLUSION

This paper presented an automated deep learning system for detecting genuine and forged signatures using a Siamese Neural Network with contrastive learning. The system processes pairs of signature images through twin convolutional neural network branches, maps them into a shared 128-dimensional embedding space, and classifies the pair as a

genuine match or a forgery based on the Euclidean distance relative to a calibrated threshold. The contrastive training objective enforces a structured geometry in the embedding space that promotes intra-class compactness for genuine signatures and inter-class separation for genuine–forged pairs, enabling reliable discrimination even against skilled forgeries.

The model achieved an accuracy of 93% on the test set, with closely matched precision and recall values, demonstrating effectiveness in handling real-world signature variability without bias towards either class. The end-to-end implementation—spanning a TensorFlow/Keras model, a FastAPI backend, and a Next.js frontend—enables real-time web-based verification suitable for institutional deployment in banking, legal, and identity authentication sectors. The containerised deployment architecture further facilitates rapid integration into existing organisational infrastructure, supporting both cloud-hosted and on-premises installation scenarios.

The Siamese framework's writer-independent design means the trained model can be applied directly to new users without retraining, offering practical scalability advantages over enrolment-based systems. The exposed distance score alongside the binary decision enhances transparency and supports risk-stratified review workflows in regulated environments. Future work will explore deeper CNN backbones, Bidirectional LSTM integration for sequential stroke modelling, larger and more diverse training datasets, adaptive threshold calibration using Bayesian optimisation, and explainability visualisations such as Grad-CAM heatmaps to highlight the signature regions most influential in verification decisions. Such advances are expected to further narrow the performance gap with the top-performing handcrafted-feature approaches while retaining the generalisation and scalability advantages of the learned metric learning paradigm.

References

[1] Y. Qi and B. R. Hunt, "Signature verification using global and grid features," *Pattern Recognition*, vol. 27, no. 12, pp. 1621–1629, 1994.

[2] B. Kareem Abd, Q. Khaled Abood, and N. A. Z. Abdullah, "Handwritten signature verification based on geometric and grid features," *International Journal of Computer Applications*, vol. 98, no. 9, pp. 1–6, 2014.

[3] Y. Inan and B. Sekeroglu, "Signature recognition using backpropagation neural network," in *Proceedings of the International Conference on Computer Systems and Technologies*, pp. 1–6, 2007.

[4] P. Patil, B. Almeida, N. Chettiar, and J. Babu, "Offline signature recognition system using histogram of oriented gradients," *International Journal of Computer Applications*, vol. 165, no. 7, pp. 14–19, 2017.

[5] S. Dey, A. Dutta, J. Toledo, S. K. Ghosh, J. Lladós, and U. Pal, "SigNet: Convolutional Siamese network for writer independent offline signature verification," *Pattern Recognition Letters*, vol. 128, pp. 163–171, 2019.

[6] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a Siamese time delay neural network," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 7, no. 4, pp. 669–688, 1993.

[7] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proceedings of the IEEE CVPR*, pp. 539–546, 2005.

[8] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML Deep Learning Workshop*, 2015.

[9] L. G. Hafemann, L. S. Oliveira, and R. Sabourin, "Learning features for offline handwritten signature verification using deep convolutional neural networks," *Pattern Recognition*, vol. 70, pp. 163–176, 2017.

[10] S. Fiel and R. Sablatnig, "Writer-independent offline signature verification using convolutional neural networks," in *Proceedings of ICDAR*, pp. 1031–1036, 2015.

[11] S. Dey and V. V. Phoha, "Deep learning for offline signature verification: A survey," *IEEE Access*, vol. 8, pp. 168868–168887, 2020.

[12] R. Plamondon and S. N. Srihari, "Online and offline handwriting recognition: A comprehensive survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 63–84, 2000.

[13] S. Impedovo and G. Pirlo, "Automatic signature verification: The state of the art," *IEEE Transactions on Systems, Man, and Cybernetics—Part C*, vol. 38, no. 5, pp. 609–635, 2008.

[14] M. Diaz, M. A. Ferrer, and A. Morales, "A survey of handwritten signature verification," *ACM Computing Surveys*, vol. 49, no. 4, pp. 1–37, 2016.