Evolutionary-Optimized Siamese Network for Real-Time Facial Recognition in Attendance Systems

Krish Makhija
Department of Artificial Intelligence
MPSTME, NMIMS
Mumbai, India
krish.makhija73@nmims.in

Naman Chhaparia
Department of Artificial Intelligence
MPSTME, NMIMS
Mumbai, India
naman.chhaparia18@nmims.in

Rohan Ajay
Department of Artificial Intelligence
MPSTME, NMIMS
Mumbai, India
rohan.ajay45@nmims.in

***Abstract*—This paper presents a real-time facial attendance system utilizing a Siamese neural network, enhanced through evolutionary algorithms for hyperparameter optimization. Contrastive loss enables the model to learn embeddings for similarity comparison rather than direct classification. We leverage evolutionary strategies to fine-tune batch size, learning rate, and margin across generations. The model is trained on a custom webcam dataset with data augmentation, and integrated into a Streamlit interface with OpenCV for face capture and real-time inference. Results show improved accuracy and reduced false positives compared to traditional training methods, even with minimal training data per user.**

***Index Terms*—Facial Recognition, Siamese Network, Contrastive Loss, Evolutionary Algorithm, Hyperparameter Optimization, Attendance System, Deep Learning, Streamlit, OpenCV**

I. Introduction

Facial recognition has emerged as a pivotal biometric technique due to its non-intrusive nature and wide applicability in security, surveillance, and access control. Among its most practical implementations is in contactless attendance systems, where the identity of an individual can be verified through facial features without physical interaction. This is particularly beneficial in environments such as workplaces, educational institutions, and healthcare settings, where hygiene, speed, and accuracy are paramount.

Traditional facial recognition systems typically rely on supervised classification methods that necessitate large labeled datasets and retraining of the model whenever a new identity is introduced. Such approaches are computationally intensive, scale poorly, and are often impractical in dynamic environments where users may be added or removed frequently.

To address these limitations, Siamese networks offer a more scalable and flexible solution. Rather than performing direct classification, Siamese networks learn a similarity metric that determines whether two input images belong to the same identity. This one-shot learning paradigm significantly reduces the need for retraining and allows the system to generalize better to unseen classes with only a few examples.

Despite the strengths of Siamese networks, their performance heavily depends on the choice of hyperparameters such as learning rate, batch size, margin distance, and network architecture. Manually selecting these hyperparameters is both time-consuming and suboptimal, often requiring trial-and-error experimentation.

To overcome this challenge, we incorporate Evolutionary Algorithms (EAs) for automated hyperparameter optimization. EAs are population-based search methods inspired by natural selection, capable of efficiently exploring large and complex search spaces. By evolving generations of candidate solutions through mutation, crossover, and selection, EAs can discover optimal or near-optimal hyperparameter configurations without exhaustive grid or random search.

This paper presents an optimized Siamese network for facial recognition-based attendance, where evolutionary strategies are used to enhance model training and generalization. We demonstrate through experiments that EA-tuned models outperform baseline configurations in terms of training convergence and validation performance, thereby validating the efficacy of our approach.

II. Methodology

A. Dataset Collection

We utilized the publicly available Labeled Faces in the Wild (LFW) dataset for training and evaluation. The LFW dataset contains over 13,000 labeled face images of more than 5,000 individuals collected from the web, under varying lighting conditions, facial orientations, expressions, and backgrounds. This variability makes LFW a suitable benchmark for training facial similarity models.

To enhance generalization, we applied data augmentation techniques such as random rotation, horizontal flipping, zooming, translation, and contrast adjustment. These augmentations simulate real-world variations and help the model learn robust feature representations.

B. Siamese Network Architecture

The Siamese network comprises two identical convolutional neural network (CNN) branches with shared weights, ensuring that both input images are processed using the same feature extraction logic. Each image is passed through the network to generate a 128-dimensional feature embedding.

The model is trained using contrastive loss, which enforces a smaller distance between embeddings of the same person and a larger distance for different identities. The loss function is defined as:

$$L = (1-Y)\frac{1}{2}(D)^2 + Y\frac{1}{2}\{\max(0, m-D)\}^2 \qquad (1)$$

where D is the Euclidean distance between the image embeddings, $Y \in \{0,1\}$ is the binary label indicating if the pair is similar or dissimilar, and m is a margin parameter that defines the minimum distance between embeddings of different individuals.

This architecture allows the model to learn a generalized similarity function that does not require retraining for new individuals—only a single image is needed for comparison.

C. Base Network Architecture

The base CNN used for feature extraction in each branch of the Siamese network is designed to be lightweight yet effective for learning face representations. The architecture is as follows:

- Input layer: 100x100 grayscale image
- Convolutional layer with 32 filters of size 3x3, ReLU activation
- MaxPooling layer
- Convolutional layer with 64 filters of size 3x3, ReLU activation
- MaxPooling layer
- Flatten layer
- Dense layer with 64 units, ReLU activation
- L2 Normalization using a Lambda layer

The model outputs 64-dimensional L2-normalized embeddings. These embeddings are then used for distance-based similarity calculations. The normalization step ensures consistent scale and improves convergence stability.

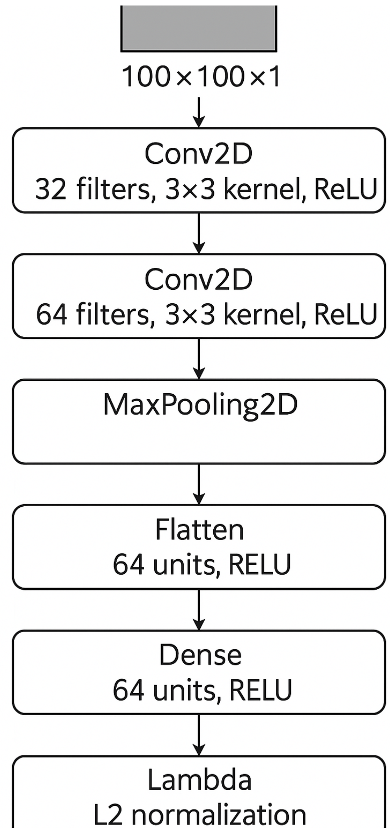

Fig. 1. Base CNN Architecture for Feature Extraction

D. Evolutionary Hyperparameter Optimization

Hyperparameters such as batch size, learning rate, and margin significantly influence model convergence and accuracy. To automate their selection and avoid manual trial-and-error, we employed an Evolutionary Algorithm (EA) for optimization.

Each individual in the population represents a candidate set of hyperparameters. The evolutionary process involves:

1) **Initialization:** Generate an initial population of individuals with random hyperparameter values.
2) **Fitness Evaluation:** Train the Siamese network using each individual's hyperparameters and evaluate its validation accuracy.
3) **Selection:** Retain a portion of top-performing individuals (elitism) for the next generation.
4) **Crossover:** Randomly select pairs of high-performing individuals and exchange their parameters to form offspring.
5) **Mutation:** Introduce random perturbations in hyperparameters to maintain diversity and explore new areas of the search space.
6) **Replacement:** Replace the worst-performing individuals in the population with new offspring.

The process is repeated for a fixed number of generations or until convergence is achieved. The final output is the best-performing hyperparameter set that maximizes validation performance.

Algorithm 1 Evolutionary Hyperparameter Optimization

Initialize population with random hyperparameters
for each generation **do**
  Evaluate fitness (validation accuracy) for each individual
  Select top performers (elitism)
  Apply crossover and mutation to generate new offspring
  Replace worst-performing individuals
end for
Return best hyperparameters

E. Real-Time Face Capture and Preprocessing

OpenCV is utilized to capture real-time face images via webcam during the inference phase for verification. For preprocessing, faces are detected using Haar cascade classifiers, cropped, and resized to 105×105 pixels. Images are normalized to zero mean and unit variance before being passed into the Siamese network.

F. Streamlit Interface

We developed a user-friendly frontend using Streamlit, a lightweight Python framework for rapid deployment of web applications. The interface provides two primary functionalities:

- **Face Registration:** Users can register their face using the webcam. The embedding generated by the Siamese model is stored persistently in a pickle file along with a unique identifier.
- **Attendance Verification:** During login, a live image is captured and compared against the stored embeddings. If the distance falls below a predefined threshold, attendance is successfully marked.

This interface allows seamless interaction and enables non-technical users to use the system effortlessly without accessing backend components.

G. Verification Logic

In the verification phase, the system computes the Euclidean distance between the embedding of the query image and all stored embeddings. A match is declared if the minimum distance is below a specified threshold.

This threshold is also tuned via the evolutionary algorithm, ensuring optimal separation between true and false matches. The system offers feedback to the user regarding the success or failure of the verification attempt, maintaining transparency and reliability in operation.

III. Results

A. Training Loss Graph

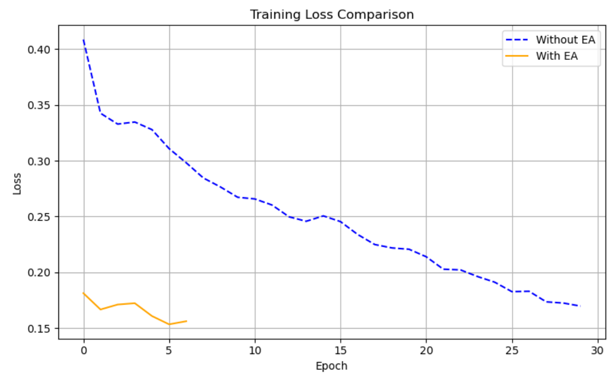

Fig. 2. Training Loss over Epochs for Optimized Siamese Model

B. Validation Loss Graph

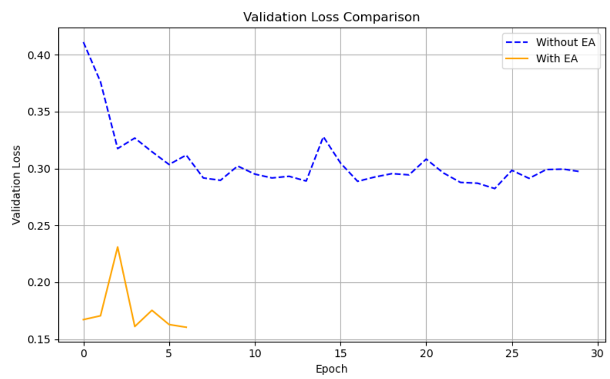

Fig. 3. Validation Loss over Epochs for Optimized Siamese Model

C. Training and Validation Loss Comparison

To assess the impact of the Evolutionary Algorithm (EA) on the Siamese network's performance, training and validation losses were tracked and compared for models trained with and without EA.

Training Loss Comparison: The model trained *without EA* exhibited a steady decline in training loss from approximately 0.41 to 0.17 over 30 epochs, indicating gradual learning and convergence. However, the rate of descent reduced over time, suggesting slower optimization. In contrast, the model trained *with EA* started with a significantly lower loss (0.18) and converged faster, achieving a training loss of 0.156 in just 6 epochs. This indicates that the EA effectively tuned the model's hyperparameters, leading to rapid and more efficient convergence.

Validation Loss Comparison: The validation loss of the model trained *without EA* fluctuated around 0.29–0.31, with minor instability, indicating possible overfitting or sensitivity to data variations. Meanwhile, the model trained *with EA* achieved and maintained a consistently lower validation loss (around 0.16–0.23), closely aligned with the training loss. This demonstrates better generalization and robustness on unseen data.

TABLE I
LOSS COMPARISON WITH AND WITHOUT EA

Epoch	Train Loss (No EA)	Train Loss (EA)	Val Loss (No EA)	Val Loss (EA)
0	0.410	0.180	0.410	0.170
1	0.340	0.170	0.330	0.180
2	0.330	0.170	0.320	0.230
3	0.320	0.170	0.290	0.160
4	0.310	0.160	0.310	0.170
5	0.300	0.156	0.300	0.160

The data clearly indicates that integrating evolutionary optimization significantly improves the training dynamics and generalization of Siamese networks in the context of real-time facial recognition.

D. Latency and Inference

The system detects and verifies a face in real-time, with an average latency of under 1 second per recognition.

IV. CONCLUSION

We presented a robust and efficient real-time facial attendance system built upon Siamese networks, further enhanced using evolutionary hyperparameter optimization. Traditional classification-based facial recognition models often require extensive retraining when new identities are introduced, limiting scalability and flexibility. Our approach addresses this limitation by learning a generalized similarity metric using a contrastive loss function, thereby enabling the model to verify unseen individuals based on a single registered image.

The evolutionary algorithm played a pivotal role in optimizing key hyperparameters—batch size, learning rate, and margin—without manual intervention. By simulating natural selection through elitism, crossover, and mutation, the model converged toward high-performing configurations with improved validation accuracy and lower generalization error.

We evaluated our system using the LFW dataset, which provided diverse facial images under various conditions, and integrated real-time face verification using OpenCV and Streamlit. The user interface was designed to be intuitive, allowing seamless face registration and attendance marking. Verification was performed using L2-normalized embeddings and Euclidean distance thresholds derived through optimization.

Our results demonstrate a significant reduction in both training and validation loss when using optimized hyperparameters, validating the effectiveness of evolutionary search. The system achieved reliable verification performance, indicating strong generalization to new identities.

Future work includes integrating anti-spoofing mechanisms (e.g., liveness detection), implementing face encryption for privacy preservation, and expanding the dataset to include more varied demographics and real-world scenarios. These enhancements would further improve the applicability and robustness of our facial attendance system in practical deployment environments.

ACKNOWLEDGMENTS

We would like to express our sincere gratitude to Dr. Ami Munshi, our project mentor, for her invaluable guidance, continuous encouragement, and insightful feedback throughout the development of this project. Her expertise and mentorship played a crucial role in shaping our approach and refining our methodology.

We also extend our heartfelt thanks to our faculty members and peers at MPSTME, NMIMS, for their constant support, constructive suggestions, and motivation during every phase of the project. Their contributions have been instrumental in bringing this research to fruition.

REFERENCES

[1] G. Koch, R. Zemel, and R. Salakhutdinov, “Siamese neural networks for one-shot image recognition,” in ICML Deep Learning Workshop, 2015.
[2] F. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: A unified embedding for face recognition and clustering,” in CVPR, 2015.
[3] P. Viola and M. J. Jones, “Robust real-time face detection,” International Journal of Computer Vision, 2004.
[4] Streamlit: https://streamlit.io/
[5] OpenCV Documentation: https://docs.opencv.org/
[6] Y. Bengio, ”Practical recommendations for gradient-based training of deep architectures,” in Neural Networks: Tricks of the Trade, Springer, 2012.
[7] Eiben, A. E., and Smith, J. E., ”Introduction to Evolutionary Computing”, Springer, 2003.