© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1723075

Artificial Bee Colony Hyperparameter Optimization of Convolutional Neural Networks for Multimodal Biometric Authentication in Smart-Home IoT Environments

AKANDE, O. V.[1], AFOLABI, A. O.[2], SAMUEL, O. D.[3], LAWAL, S. K.[4], IDOWU, A. I.[5]
[1, 3, 6] *Department of Computer Science, Ladoke Akintola University of Technology, Ogbomoso, Nigeria*
[2] *Department of Cybersecurity Science, Ladoke Akintola University of Technology, Ogbomoso, Nigeria*
[4, 5, 7] *Department of Computer Science, Federal Polytechnic Ayede, Oyo State, Nigeria*

Abstract- ***Convolutional neural network (CNN) performance is highly dependent on model hyperparameters, yet manually selected configurations may not provide the best balance of recognition accuracy, error rate and computational cost for multimodal biometric authentication. This study evaluated the effect of Artificial Bee Colony (ABC) hyperparameter optimization on a CNN classifier for face-palm-iris recognition in a smart-home Internet of Things setting. The MULB dataset contained 10,266 biometric samples from 219 identity classes. Images were standardized to 128 × 128 pixels, transformed into a common 150-dimensional fused PCA-LDA representation, and divided into 8,212 training and 2,054 testing samples. A conventional CNN served as the baseline, while ABC searched convolutional filters, kernel size, dense units, dropout rate and learning rate. Each classifier was executed four times and assessed using accuracy, precision, recall, F1-score, false positive rate (FPR), false negative rate (FNR) and classifier-level recognition time. The ABC-CNN increased mean accuracy from 80.92 ± 0.30% to 84.81 ± 0.37% and F1-score from 80.66 ± 0.29% to 84.59 ± 0.38%. FPR fell from 0.0878 ± 0.0011% to 0.0694 ± 0.0012%, while FNR fell from 14.747 ± 0.12% to 10.85 ± 0.25%. Recognition time increased by 0.000062 s/sample. Paired-samples t-tests reported significant differences for all seven metrics (p < 0.001). The findings show that ABC-based optimization improved recognition effectiveness and reduced classification errors, with a small increase in classifier-level processing time.***

Index Terms *— Artificial Bee Colony, convolutional neural network, hyperparameter optimization, multimodal biometrics, smart home, Internet of Things*

I. INTRODUCTION

Smart-home Internet of Things (IoT) environments increasingly support access control, surveillance, connected appliances and other services that expose sensitive personal and operational information. Authentication in such environments must therefore distinguish legitimate residents from unauthorized users while maintaining response times that are compatible with interactive access control. Biometric recognition is attractive because the authentication evidence is linked directly to the user; however, the reliability of a single biometric trait can decrease when acquisition conditions, sensor quality or user presentation vary [7]–[11].

Multimodal biometric systems reduce dependence on one trait by combining complementary identity evidence. In the present framework, face, palm and iris information were represented through a common PCA-LDA feature pipeline before CNN classification. Because the same fused representation was supplied to the two classifiers considered in this paper, the central experimental question is not whether fusion is useful, but whether automated ABC-based CNN configuration can improve recognition performance relative to a conventional CNN under the reported experimental conditions.

CNN behaviour is sensitive to architectural and training hyperparameters such as convolutional filter counts, kernel size, dense-layer capacity, dropout and learning rate. These parameters interact and define the capacity, regularization and convergence behaviour of the classifier. Manual tuning can therefore be inefficient and difficult to reproduce when many candidate combinations are possible [2], [3]. Metaheuristic optimization offers an alternative by

searching the configuration space according to a measurable fitness criterion.

Artificial Bee Colony (ABC) optimization is a population-based swarm technique in which candidate solutions are represented as food sources. Employed bees explore neighbouring candidates, onlooker bees allocate more search effort to promising candidates, and scout behaviour reintroduces diversity when a candidate ceases to improve [1]. In CNN hyperparameter optimization, each food source can encode one model configuration and validation performance can be used to rank candidate solutions. This study applies that principle to multimodal biometric recognition and evaluates whether the resulting ABC-CNN provides a measurable advantage over the ordinary CNN baseline.

II. RELATED WORK

Hyperparameter optimization has become an important component of machine-learning model development because the mapping between a configuration and its generalization performance is typically nonlinear and expensive to evaluate. Yang and Shami [2] reviewed manual, grid, random, Bayesian and evolutionary strategies, while Bischl et al. [3] emphasized reproducible search spaces, objective functions and validation procedures. Swarm-based techniques are particularly useful when gradients with respect to hyperparameters are unavailable or when a search space contains mixed discrete and continuous choices.

Manual search is simple but depends heavily on researcher experience and becomes inefficient when several CNN hyperparameters interact. Grid search is reproducible because it evaluates predetermined combinations, yet the number of trials increases rapidly as more candidate values are introduced. Random search reduces the cost by sampling only part of the space, but it can still spend evaluations in weak regions. Bayesian optimization is more sample-efficient because it uses previous evaluations to guide the next trial, although the surrogate model and acquisition process introduce additional modelling decisions [2], [3].

Evolutionary and swarm-based methods provide another route for difficult black-box search spaces. Genetic algorithms maintain a population and use selection, crossover and mutation; particle swarm optimization moves candidate configurations according to individual and collective search experience; and ABC updates food-source solutions through employed, onlooker and scout behaviour. These methods do not require gradients with respect to the hyperparameters, making them suitable for discrete architectural choices such as filter counts and kernel sizes together with continuous-like training choices such as learning rate [2]–[4].

For biometric authentication, optimization should be assessed with more than accuracy alone. A configuration that improves overall identification but increases false assignments or produces an impractical processing cost may be unsuitable for access-control use. This motivates the present use of precision, recall, F1-score, FPR, FNR and recognition time together with accuracy. The use of repeated runs is also important because CNN training is stochastic and a single execution may overstate or understate the effect of a chosen configuration.

ABC has been applied to CNN configuration because its employed-, onlooker- and scout-bee phases provide a simple balance between exploitation of promising settings and continued exploration. Inik [4] demonstrated this principle in SwarmCNN, where ABC and particle swarm optimization were used to identify effective CNN hyperparameters. That work supports the use of ABC for model configuration, although it was not designed specifically for multimodal face-palm-iris authentication.

Recent multimodal biometric studies also show that classifier design materially affects recognition performance. Alay and Al-Baity [5] used deep learning with iris, face and finger-vein fusion, while Wang et al. [6] employed CNN-based multimodal fusion. More recent systems have investigated CNN architecture and fusion design in fingerprint, finger-vein and smart-environment authentication [7], [8]. SecureSense [9] demonstrated multimodal person verification using face, fingerprint and iris, and Kadhim and Abdulameer [10] studied face, palm and iris identification on the MULB dataset. These studies

establish the relevance of multimodal recognition but do not isolate the effect of ABC hyperparameter optimization on the same fused face-palm-iris CNN pipeline.

Accordingly, the present comparison keeps the dataset, preprocessing, fused feature input and identity classes common to the two classifiers. The intended experimental contribution is the direct assessment of the optimized ABC-CNN against the conventional CNN using repeated runs, multiple classification/error measures, training and validation curves, recognition-time measurement and a paired statistical comparison.

III. METHODOLOGY

A. Dataset, preprocessing and common classifier input

The study used the MULB Multimodal Biometric Dataset containing face, palm and iris images. A total of 10,266 samples belonging to 219 identity classes were loaded. All images were standardized to 128 × 128 pixels before feature extraction. Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) each produced a 150-dimensional representation, after which the normalized outputs were combined through weighted-sum feature fusion. The fused 150-dimensional vector was used as the common input for both the conventional CNN and ABC-CNN. This common input is important because it keeps the representation stage fixed while the classifier configuration is evaluated.

Table 1
Experimental Setup and Implementation Configuration

Parameter	Value
Dataset	MULB Multimodal Biometric Dataset
Biometric modalities used	Face, Palm, and Iris
Actual samples loaded in implementation	10,266
Number of identity classes	219
Image size used	128 × 128 pixels
Train-test split	80:20
Training samples	8,212
Testing samples	2,054

The dataset was partitioned using an 80:20 train-test ratio. Where validation was required during model development and ABC search. Validation data were obtained from the training portion so that the final test set remained outside hyperparameter selection. The same preprocessing and fused feature representation were used for the two classifier variants.

B. Conventional CNN baseline

The conventional CNN was used as the baseline against which the optimized model was evaluated. It processed the fused feature vector after reshaping it for convolutional operations. The baseline contained two convolutional blocks, a dense layer, dropout and a softmax output for 219 identity classes. It was trained for 30 epochs with a batch size of 32. The baseline was intentionally retained because it provides a direct reference for determining whether the ABC-guided model-development procedure produced higher recognition performance than a fixed CNN configuration.

C. Artificial Bee Colony hyperparameter search

ABC was used to search the number of filters in the first and second convolutional layers, kernel size, dense-layer units, dropout rate and learning rate. The optimization was initialized with eight food sources and eight employed bees, while eight onlooker bees explored promising candidates. The maximum search length was 15 iterations, the abandonment limit was 3, and each candidate CNN was trained for five epochs before its validation performance was assessed. A random seed of 42 was used in the implementation.

The search began by generating candidate CNN configurations from the predefined parameter sets. During the employed-bee phase, a current configuration was perturbed to create a neighbouring candidate and the new candidate was retained when its fitness improved. Onlooker bees then selected food sources according to relative fitness so that stronger configurations received additional search effort. When a food source exceeded the abandonment limit without improvement, scout behaviour replaced it with a newly sampled configuration. The sequence of employed, onlooker and scout phases was repeated

until the maximum number of optimization cycles was reached.

Each candidate represented a complete combination of the searched hyperparameters. The first- and second-layer filter counts controlled the number of learned convolutional maps; kernel size controlled the receptive pattern examined at each convolution; dense-layer units determined the capacity of the final nonlinear classifier; dropout provided regularization; and learning rate controlled the magnitude of gradient-based weight updates. Searching the parameters jointly is important because a value that performs well in one configuration may behave differently when paired with another filter count, dense size or learning rate.

Table 2
Artificial Bee Colony Optimization Configuration

Parameter	Value
Initial candidate solutions / food sources	8
Employed bees	8
Onlooker bees	8
Maximum iterations	15
Abandonment limit	3
Candidate evaluation epochs	5
Random seed	42

Table 3
ABC-CNN Hyperparameter Search Space and Reported Resulting Values

Hyperparameter	Search Space	Resulting Value
Convolutional layer 1 filters	16, 32, 64, 128	32
Convolutional layer 2 filters	32, 64, 128, 256	64
Kernel size	2, 3, 5	3
Dense-layer units	128, 256, 512	256
Dropout rate	0.2, 0.3, 0.5, 0.7	0.5
Learning rate	0.0001, 0.0005, 0.001, 0.005	0.001

Table 4
Reported ABC-CNN Architecture

Layer / Parameter	Configuration
Input	Fused feature vector reshaped as (n, 1, 1)
Convolutional layer 1	32 filters, kernel size (3,1), ReLU
Pooling layer 1	MaxPooling2D (2,1)
Convolutional layer 2	64 filters, kernel size (3,1), ReLU
Pooling layer 2	MaxPooling2D (2,1)
Flatten layer	Applied
Dense layer	256 units, ReLU
Dropout rate	0.5
Output layer	219 units, Softmax
Optimizer	Adam
Learning rate	0.001
Loss function	Categorical cross-entropy
Batch size	32

D. Interpretation of the ABC search outcome

Taken together, Tables 2–4 define the optimization as a bounded discrete hyperparameter-search problem rather than an unrestricted redesign of the CNN. The Cartesian product of the listed choices contains 2,304 possible configurations: four alternatives for each convolutional filter count, three kernel sizes, three dense-layer sizes, four dropout values and four learning rates. An exhaustive grid would therefore require a large number of candidate-model evaluations. The ABC procedure instead used a population of candidate food sources and iteratively concentrated search effort on configurations that produced stronger validation fitness, while the scout mechanism preserved the ability to introduce new configurations after stagnation.

The reported solution selected 32 filters in the first convolutional layer, 64 in the second, a kernel size of 3, 256 dense units, dropout of 0.5 and a learning rate of 0.001. Notably, the reported solution did not simply choose the largest network capacity available in the search space. The selected filter counts and dense-layer size were intermediate values, while the dropout rate was relatively strong. This indicates that the

optimization outcome should be interpreted as a joint configuration chosen according to validation behaviour rather than as evidence that larger layers are inherently superior. Because the hyperparameters interact, the effect of any one selected value cannot be isolated from this experiment without a separate ablation study.

Candidate models were evaluated for five epochs during the search, whereas the final optimized model was trained after the search had identified a preferred configuration. These stages should not be conflated: the five-epoch candidate runs were a search-time approximation used to compare configurations, not the final ABC-CNN training duration. The final training procedure also employed validation-based stopping and learning-rate reduction, while the conventional CNN was reported with a fixed 30-epoch schedule. Accordingly, the subsequent comparison is best read as an evaluation of the reported ABC-guided CNN development procedure against the reported conventional CNN procedure. This distinction is revisited under the study limitations.

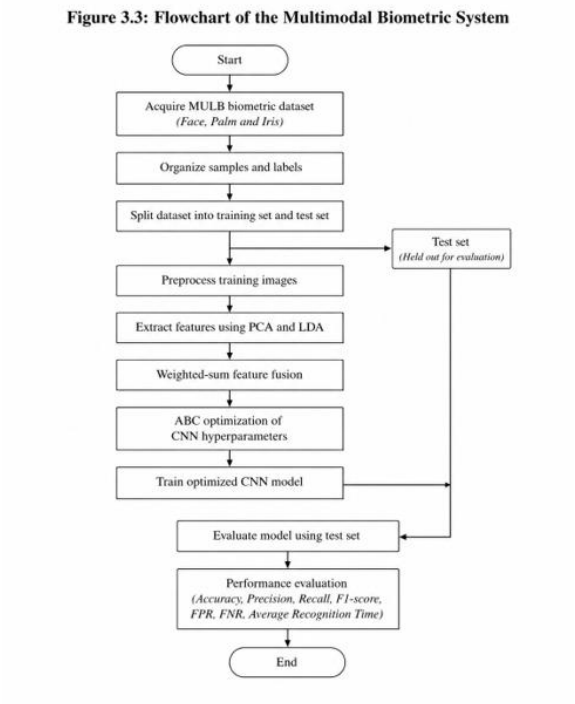

Figure 1. Workflow showing ABC optimization before final model evaluation.

The ABC mechanism iteratively explored the candidate space and retained stronger configurations according to validation performance. The final reported architecture in Table 4 was subsequently trained as the ABC-CNN classifier. The optimization process therefore operates before final testing; it does not directly modify individual test predictions but instead determines the CNN configuration used to learn from the fused biometric features.

D. Performance evaluation

Performance was evaluated using accuracy, weighted precision, weighted recall, weighted F1-score, false positive rate (FPR), false negative rate (FNR) and average recognition time. Accuracy measured the overall proportion of correctly assigned identity samples; precision measured the reliability of predicted identities; recall measured the proportion of actual identity instances correctly recognized; and F1-score summarized the balance between precision and recall. FPR and FNR were included because wrong positive assignments and missed genuine identities are operationally important in authentication systems. Recognition time was measured as the average classifier inference time per test sample, not as total end-to-end authentication latency.

To reduce dependence on a single random initialization, each classifier was independently executed four times under the same reported experimental conditions. Mean and standard deviation were therefore used to summarize performance across the repeated executions.

E. Paired statistical comparison

Corresponding conventional-CNN and ABC-CNN runs were treated as paired observations. For each metric, the paired difference was defined as $d_i = XABC_{,i} - XCNN_{,i}$. With $n = 4$ paired runs, the mean difference and the variability of the differences were used to compute the paired-samples t statistic. The test used three degrees of freedom ($df = n - 1 = 3$) and a significance level of $\alpha = 0.05$.

$$\bar{d} = (1/n) \Sigma d_i, \qquad t = \bar{d} / (s\_d / \sqrt{n}), \qquad df = n - 1$$

The null hypothesis for each metric was H_0: $\mu d = 0$, indicating no mean performance difference between the classifiers. The alternative hypothesis was H_1: μd

≠ 0. A p-value below 0.05 was interpreted as evidence of a statistically significant paired difference. Direction was considered separately: positive differences are desirable for accuracy, precision, recall and F1-score, whereas negative differences are desirable for FPR and FNR. For recognition time, a positive difference indicates that the optimized model is slower.

IV. RESULTS AND DISCUSSION

A. Conventional CNN training and validation behaviour

The conventional CNN was trained for 30 epochs using a batch size of 32. Final training accuracy reached 95.89%, whereas final validation accuracy was 80.92%, producing a final training-validation gap of approximately 14.97 percentage points. The highest validation accuracy was 82.04%. Training loss declined to 0.1320, but validation loss ended at 1.1291 after reaching a minimum of 0.7966. The pattern indicates that the model continued fitting the training samples after validation performance had largely stabilized.

Table 5
Training and Validation Results of the Conventional CNN

Performance Measure	Result
Number of training epochs	30
Batch size	32
Final training accuracy (%)	95.89
Final validation accuracy (%)	80.92
Final training loss	0.1320
Final validation loss	1.1291
Highest validation accuracy (%)	82.04
Lowest validation loss	0.7966

The early-epoch values make this pattern clearer. Training accuracy increased from 2.20% at epoch 1 to 28.71% at epoch 2 and 59.48% at epoch 3, while validation accuracy rose from 31.94% to 66.46% and 75.61% over the same epochs. Training accuracy reached 78.66% at epoch 5 and 86.26% at epoch 7, after which it continued increasing toward the mid-90% range. Validation accuracy, in contrast, changed much less after the early learning phase and reached its highest value of 82.04% at epochs 26 and 29 before ending at 80.92%. This divergence indicates that later epochs benefited fitting of the training data more than validation recognition.

The loss curves show the same transition. Training loss declined from 5.2333 at epoch 1 to 3.2682 at epoch 2, 1.6171 at epoch 3 and 0.7486 at epoch 5. Validation loss fell from 3.6375 to 1.6738 and 1.0710 over the first three epochs and reached its minimum of 0.7966 at epoch 7. After that point, training loss continued downward to 0.1320, whereas validation loss gradually increased and finished at 1.1291. The lowest validation loss therefore occurred substantially earlier than the final epoch.

Table 6
Performance Summary of the Conventional CNN

Performance Metric	Result
Accuracy (%)	80.92 ± 0.30
Precision (%)	82.26 ± 0.30
Recall (%)	80.92 ± 0.30
F1-score (%)	80.66 ± 0.29
False positive rate (%)	0.0878 ± 0.0011
False negative rate (%)	14.747 ± 0.12
Average recognition time (s/sample)	0.000098 ± 0.000003

Across four runs, the conventional model produced 80.92 ± 0.30% mean accuracy and 80.66 ± 0.29% F1-score. The small standard deviations show that the baseline performance was relatively stable across repeated executions. Its FPR was very low, but the FNR of 14.747% indicates that missed genuine identity instances remained a more substantial source of error than false positive assignments.

B. ABC-CNN training and validation behaviour

The ABC-CNN was configured for a maximum of 50 epochs but completed 21 epochs under the reported early-stopping condition. Final training accuracy was 99.44% and final validation accuracy was 84.66%. The highest validation accuracy, 85.20%, occurred at epoch 20, while the lowest validation loss, 0.7259, occurred at epoch 11. Final training and validation losses were 0.0156 and 0.7895, respectively. The final training-validation accuracy gap was approximately

14.78 percentage points, slightly smaller than the conventional model gap.

Table 7
Training and Validation Results of the ABC-CNN

Performance Measure	Result
Maximum configured epochs	50
Actual epochs completed	21
Initial learning rate (reported)	0.0005
Final training accuracy (%)	99.44
Final validation accuracy (%)	84.66
Final training loss	0.0156
Final validation loss	0.7895
Highest validation accuracy (%)	85.20
Epoch of highest validation accuracy	20
Lowest validation loss	0.7259
Epoch of lowest validation loss	11

The optimized model learned useful identity patterns rapidly. Validation accuracy exceeded 80% by the third epoch and remained above approximately 82% for most of the subsequent training period. Although the training trajectory still approached near-perfect fit, the optimized model maintained higher validation accuracy and lower validation loss than the baseline at the end of training.

Validation accuracy increased from 50.39% at epoch 1 to 76.73% at epoch 2 and 80.87% at epoch 3. By epoch 4 it had reached 82.47%, while training accuracy had already reached 87.28%. Training accuracy rose to 91.61% by epoch 5 and 97.97% by epoch 11. Validation accuracy increased more gradually after the initial surge, reaching 84.81% at epoch 11, 85.05% at epoch 19 and the maximum of 85.20% at epoch 20 before ending at 84.66% at epoch 21. Thus, most of the validation gain occurred early, but the optimized model sustained a validation level above the conventional CNN plateau.

ABC-CNN loss declined sharply during the same period. Training loss decreased from 5.1553 at epoch 1 to 2.3564 at epoch 2, 0.9114 at epoch 3 and 0.2960 at epoch 5. Validation loss fell from 2.7446 to 1.0305 and 0.8202 over the first three epochs and dropped below 0.75 by epoch 4. Its minimum value of 0.7259 occurred at epoch 11. Later validation loss fluctuated within a narrow band and ended at 0.7895 while training loss continued toward 0.0156. This behaviour supports the use of validation-based stopping rather than continued training solely to improve training loss.

Table 8
Performance Summary of the ABC-CNN

Performance Metric	Result
Accuracy (%)	84.81 ± 0.37
Precision (%)	85.53 ± 0.38
Recall (%)	84.81 ± 0.37
F1-score (%)	84.59 ± 0.38
False positive rate (%)	0.0694 ± 0.0012
False negative rate (%)	10.85 ± 0.25
Average recognition time (s/sample)	0.000160 ± 0.000006

The ABC-CNN achieved 84.81 ± 0.37% mean accuracy, 85.53 ± 0.38% precision, 84.81 ± 0.37% recall and 84.59 ± 0.38% F1-score. These values were higher than the corresponding baseline results. FPR and FNR were also lower, while recognition time remained below one millisecond per sample at classifier level. The reported standard deviations remained small relative to the mean values, indicating that the improvement was observed consistently rather than being driven by one unusually favourable run.

C. Side-by-Side Comparison of Validation Accuracy and Validation Loss

For direct visual comparison, the corresponding learning curves are arranged by metric rather than by model. The conventional-CNN and ABC-CNN accuracy curves are placed side by side first, followed by the corresponding loss curves. This makes the effect of optimization on validation behaviour easier to examine before the final classification metrics and statistical tests are considered.

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1723075

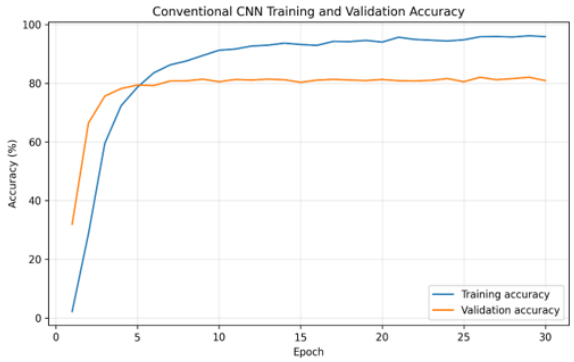

Figure 2(a). Conventional CNN training and validation accuracy.

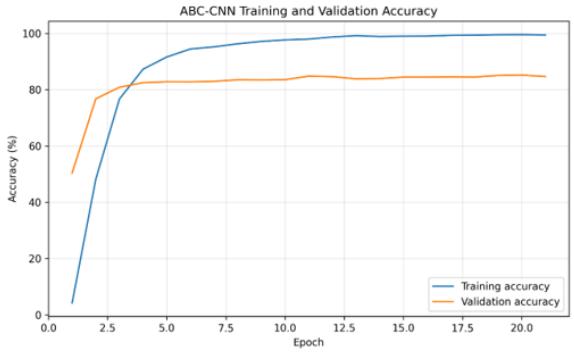

Figure 2(b). ABC-CNN training and validation accuracy.

The validation-accuracy comparison shows a clear upward shift after ABC-guided optimization. The conventional CNN achieved a highest validation accuracy of 82.04%, whereas the ABC-CNN reached 85.20%. After the rapid early learning stage, the conventional validation curve remained mainly around 80–82%, while the optimized model stabilized mostly above 82% and later approached 85%. The improvement is therefore not confined to the final test metric; it is also visible in the validation trajectory. Nevertheless, both training curves continue to rise above their validation curves, so optimization improved validation performance without completely eliminating overfitting.

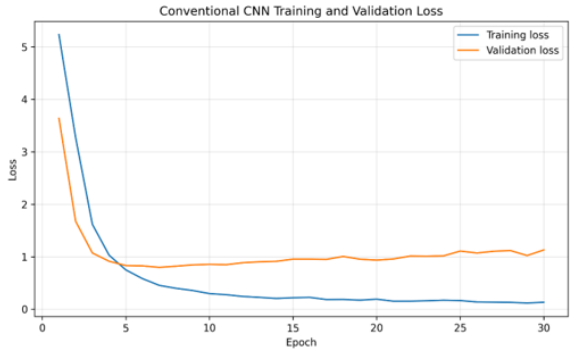

Figure 3(a). Conventional CNN training and validation loss.

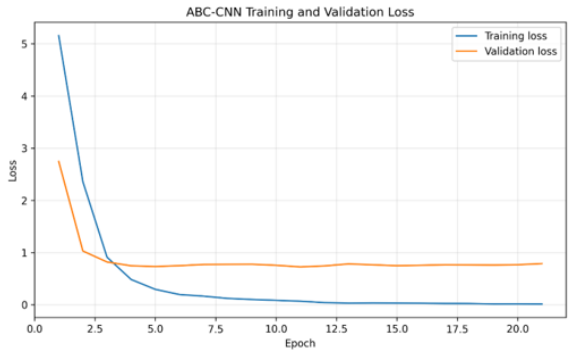

Figure 3(b). ABC-CNN training and validation loss.

The loss curves support the same interpretation. The conventional CNN reached its minimum validation loss of 0.7966 at epoch 7 and then showed a gradual increase while training loss continued to decline. The ABC-CNN achieved a lower minimum validation loss of 0.7259 at epoch 11 and subsequently fluctuated within a narrower, lower range before stopping. Thus, ABC-guided optimization produced both a higher validation-accuracy ceiling and a lower minimum validation loss. The continued decline in training loss for both models confirms that the major benefit is improved validation behaviour rather than complete removal of the training-validation gap.

D. Direct performance comparison

Table 9 brings the two models together under the same evaluation measures. ABC optimization increased accuracy and recall by 3.89 percentage points, precision by 3.27 percentage points and F1-score by 3.93 percentage points. These improvements are practically relevant because they affect both the overall proportion of correctly recognized samples and

the balance between reliable positive assignments and successful recovery of genuine identity instances.

Table 9
Comparison of Conventional CNN and ABC-CNN Performance

Performance Metric	Conventional CNN (Mean ± SD)	ABC-CNN (Mean ± SD)	Difference
Accuracy (%)	80.92 ± 0.30	84.81 ± 0.37	+3.89
Precision (%)	82.26 ± 0.30	85.53 ± 0.38	+3.27
Recall (%)	80.92 ± 0.30	84.81 ± 0.37	+3.89
F1-score (%)	80.66 ± 0.29	84.59 ± 0.38	+3.93
False Positive Rate (%)	0.0878 ± 0.0011	0.0694 ± 0.0012	−0.0184
False Negative Rate (%)	14.747 ± 0.12	10.85 ± 0.25	−3.897
Recognition Time (s/sample)	0.000098 ± 0.000003	0.000160 ± 0.000006	+0.000062

The error-rate changes strengthen the classification results. FPR fell by 0.0184 percentage points, from 0.0878% to 0.0694%, showing that incorrect positive class assignments became less frequent. FNR fell by 3.897 percentage points, from 14.747% to 10.85%. Because FNR was much larger than FPR for both models, this reduction represents an important improvement in genuine-user recognition. Nevertheless, an FNR above 10% also indicates that missed genuine identities remain a limitation of the current software implementation.

The gain in recognition effectiveness was accompanied by an increase in inference time from 0.000098 to 0.000160 s/sample. The absolute difference is 0.000062 s, or 0.062 ms, per sample. Relative to the baseline, the increase is sizeable in percentage terms because the starting value is extremely small, but the absolute classifier-level delay remains very small. Importantly, this measurement does not include image acquisition, preprocessing, feature extraction, network communication or smart-home device actuation; it should therefore not be interpreted as complete end-to-end authentication latency.

E. Paired-samples statistical test

Descriptive improvements can arise from training variability, so the four corresponding executions were also analysed as paired observations. Table 10 presents the reported t-statistics and p-values. All seven metrics had p-values below 0.001, which are lower than the 0.05 significance threshold used in the study. The null hypothesis of no paired mean difference was therefore rejected for every metric.

Table 10
Paired-Samples t-Test Comparison of Conventional CNN and ABC-CNN

Performance Metric	Mean Difference	t-value	p-value	Decision
Accuracy (%)	+3.89	75.10	<0.001	Significant
Precision (%)	+3.27	67.94	<0.001	Significant
Recall (%)	+3.89	75.10	<0.001	Significant
F1-score (%)	+3.93	73.40	<0.001	Significant
FPR (%)	−0.0184	−109.31	<0.001	Significant
FNR (%)	−3.897	−58.96	<0.001	Significant
Recognition Time (s/sample)	+0.000062	48.02	<0.001	Significant

Note: n = 4 paired runs, df = 3, α = 0.05.

For accuracy, recall and F1-score, the large positive t-values correspond to consistently higher ABC-CNN values across the paired runs. Precision shows the same direction. In contrast, FPR and FNR have negative mean differences and negative t-values because lower error values are favourable; the negative sign therefore indicates improvement rather than deterioration. Recognition time has a positive mean difference and a positive t-value, meaning that

the optimized model was consistently slower than the baseline even though the absolute increase was small. The inferential results support the descriptive comparison, but the statistical evidence should be interpreted with the sample size in mind. Only four paired runs were available, giving three degrees of freedom. With such a small number of pairs, future experiments should increase the number of independent repetitions and report confidence intervals or non-parametric sensitivity analyses in addition to the paired t-test. This would provide a more robust estimate of the magnitude and stability of the optimization effect.

F. Interpretation in relation to prior studies

The direction of the present findings agrees with the wider hyperparameter-optimization literature. Inik [4] showed that swarm-based search can discover competitive CNN configurations without relying entirely on manual tuning. The present work extends that idea to a face-palm-iris multimodal framework and demonstrates gains not only in accuracy but also in precision, recall, F1-score and biometric error measures. The improvement is therefore broader than a single headline accuracy value.

The results are also consistent with multimodal biometric studies showing that CNN configuration affects recognition quality [5]–[8]. However, direct numerical ranking against published accuracies should be avoided because datasets, feature representations, fusion strategies, identity counts and evaluation protocols differ. For example, some recent multimodal systems use end-to-end deep feature learning or adaptive fusion, while the present framework classifies a fixed 150-dimensional fused PCA-LDA representation. The current result is therefore most informative as an internal controlled comparison between the conventional and optimized CNN variants.

From a smart-home perspective, the reduction in FNR is especially relevant to usability because fewer genuine identity instances are missed. The very low FPR is also desirable because false positive assignments create security concerns. The optimized model therefore provides a better recognition-error balance than the baseline in the reported software experiment. The classifier-level speed penalty is small enough to justify further deployment-oriented testing, but practical feasibility cannot be inferred from inference time alone.

G. Magnitude of the optimization effect

The absolute percentage-point differences in Table 9 can also be expressed relative to the conventional CNN baseline to clarify their scale. Accuracy and recall increased by approximately 4.81% relative to baseline, precision by 3.98%, and F1-score by 4.87%. In contrast, FPR decreased by approximately 20.96% and FNR decreased by 26.43% relative to their baseline values. Recognition time increased by approximately 63.27% relative to the very small baseline inference time. These relative values should be interpreted together with the absolute values because a large percentage change in recognition time corresponds to only 0.062 ms/sample in absolute terms.

Table 11
Derived Relative Change from Conventional CNN to ABC-CNN

Metric	Absolute Difference	Relative Change
Accuracy	+3.89 percentage points	+4.81%
Precision	+3.27 percentage points	+3.98%
Recall	+3.89 percentage points	+4.81%
F1-score	+3.93 percentage points	+4.87%
False positive rate	−0.0184 percentage points	−20.96%
False negative rate	−3.897 percentage points	−26.43%
Recognition time	+0.000062 s/sample	+63.27%

Note: Relative changes are calculated from the reported mean values in Table 9 and are not additional experimental measurements.

The largest practically favourable relative change occurred in FNR. A reduction of about one quarter relative to the baseline indicates that the optimization procedure had a stronger proportional effect on missed

genuine identity instances than on the already very low FPR. This distinction matters because an authentication model may show only a modest increase in overall accuracy while producing a more meaningful improvement in a specific error type.

The training curves provide a complementary view of the optimization effect. The ABC-CNN reached a higher maximum validation accuracy (85.20% versus 82.04%) and a lower minimum validation loss (0.7259 versus 0.7966). It also completed 21 epochs under the reported early-stopping procedure compared with 30 fixed epochs for the baseline. These observations suggest stronger validation behaviour, although they should not be interpreted as a pure convergence-speed advantage because the final training protocols were not identical.

Taken together, the optimization results are strongest when interpreted as a multi-metric improvement rather than an isolated accuracy gain. The optimized model improved the four principal classification measures, reduced both reported biometric error measures, retained low run-to-run variability and achieved statistically significant paired differences. The only consistent cost was the small increase in classifier-level recognition time.

V. PRACTICAL IMPLICATIONS, ROBUSTNESS AND REPRODUCIBILITY

A. Security and usability interpretation of the error measures

The reduction in FPR and FNR provides information that is not captured by overall accuracy alone. The optimized classifier reduced FPR from 0.0878% to 0.0694%, while FNR declined from 14.747% to 10.85%. Because the FPR was already very small for the conventional CNN, the absolute room for improvement in that measure was limited. By contrast, FNR was substantially larger, and its reduction of 3.897 percentage points represents a more noticeable practical change. Within the reported multiclass evaluation, the optimization therefore improved both the reliability of positive identity assignments and the ability to recover genuine class instances.

The difference in scale between FPR and FNR is also important when interpreting smart-home authentication behaviour. A system may achieve a low average false positive rate while still rejecting or missing a non-trivial proportion of genuine identity instances. In an access-control setting, this creates a usability burden even when the security-oriented error measure appears strong. The ABC-CNN reduced that burden relative to the baseline, but the remaining FNR above 10% indicates that additional improvement would still be desirable before the model is treated as a mature access-control solution.

The reported FPR and FNR should not be equated automatically with threshold-based false acceptance rate and false rejection rate used in some biometric verification systems. The present experiment is a 219-class identification problem and the research computes its error measures from the multiclass confusion matrix. Consequently, the safest interpretation is as class-averaged classification error behaviour under the reported identification protocol. Future deployment studies can complement these measures with verification-oriented FAR, FRR and equal-error-rate analysis when a decision threshold and claimed identity are explicitly defined.

B. Recognition-time trade-off

The recognition-time result illustrates why absolute and relative changes should be considered together. ABC-CNN inference increased from 0.000098 to 0.000160 s/sample. Relative to the baseline, this corresponds to an increase of approximately 63.27%, yet the absolute change is only 0.000062 s, or 0.062 ms, per sample. The percentage increase therefore appears large primarily because the baseline classifier-level time is extremely small. For the software experiment reported here, the improvement in recognition metrics was obtained without introducing a large absolute inference delay.

This result nevertheless cannot establish real-time smart-home performance on its own. The measured interval covers classifier inference after the feature vector has been prepared. A practical authentication transaction would additionally include sensor capture, image quality checks, preprocessing, PCA-LDA transformation, feature fusion, communication between devices or services and the actuation of the protected smart-home resource. The study therefore provides evidence about classifier efficiency, not a

complete latency budget. This distinction should be maintained in both the conclusion and any deployment claim.

C. Robustness of the observed improvement

The mean values and standard deviations indicate that the two classifiers were reasonably stable across the four reported executions. Standard deviations remained below 0.4 percentage points for accuracy, precision, recall and F1-score in both models. The ABC-CNN showed slightly larger variability than the conventional CNN on some measures, but its mean remained higher for every principal classification metric. The repeated-run design is therefore preferable to reporting one favourable execution because it demonstrates that the observed advantage persisted across multiple independent training runs.

The paired statistical analysis provides an additional layer of evidence because it evaluates corresponding runs rather than comparing only the two grand means. The reported p-values are below 0.001 for all seven measures. For the classification metrics, the direction of the paired differences favours ABC-CNN; for FPR and FNR, negative differences are favourable because lower error is desired; and for recognition time, the positive difference confirms a consistent cost. Statistical significance therefore supports the consistency of the differences, while the effect sizes in Tables 9 and 11 describe their practical magnitude.

At the same time, the inferential evidence should be interpreted cautiously because n = 4 provides only three degrees of freedom. With so few paired observations, the estimated variance of the differences is based on a very small sample. The highly significant reported t-values indicate strong consistency in these four runs, but future studies should repeat the experiment with more random seeds and report confidence intervals around the mean differences. A non-parametric or resampling-based sensitivity analysis could also be used to check whether the conclusion remains stable without relying entirely on the assumptions of the paired t-test.

D. Reproducibility of the optimization procedure

The optimization procedure contains several details that support reproducibility. The search space is explicitly bounded; the study reports eight employed bees, eight onlooker bees, 15 maximum iterations, an abandonment limit of 3, five candidate-evaluation epochs and a random seed of 42. The final architecture and training configuration are also reported in Tables 3 and 4. These details allow another researcher to reconstruct the intended search and understand which design decisions were optimized rather than manually fixed.

The search-space specification is particularly important. The six optimized hyperparameters yield 2,304 possible combinations if every listed choice were evaluated exhaustively. ABC was used to avoid an exhaustive grid and to concentrate evaluations on more promising configurations. The reported solution selected intermediate filter and dense-layer sizes rather than the largest available values, reinforcing the idea that the search objective was validation performance rather than model size. This also means that the selected configuration should be interpreted jointly: the experiment does not establish the independent causal effect of any single hyperparameter.

Reproducibility would be strengthened further by retaining the optimization log for each run, including the initial food sources, candidate fitness values, best solution after every iteration and the exact final best-parameter dictionary. Such records would make it possible to distinguish whether different ABC runs converge to the same region of the search space or reach different configurations with similar validation performance. They would also resolve discrepancies if summary text and implementation tables report different parameter values.

E. Implications for future smart-home deployment

The present findings justify deployment-oriented evaluation rather than immediate claims of deployment readiness. The ABC-CNN improved the recognition-error trade-off on a software implementation, but a practical smart-home platform would need to assess memory use, processor load, energy consumption, sensor-to-decision latency and behaviour under variable acquisition conditions. These measurements are particularly relevant if the classifier is moved from a desktop development environment to an edge device with more limited computational resources.

A useful next step would therefore keep the same trained classifier while measuring the complete authentication pipeline on representative hardware. Face, palm and iris samples could be acquired through the intended sensors, passed through the same preprocessing and fusion stages, and timed from acquisition to final access decision. Such an experiment would determine whether the 0.062 ms classifier-level increase remains negligible once all stages are included and whether the accuracy advantage is preserved under realistic lighting, pose, sensor noise and communication conditions.

F. Model-selection interpretation of the optimization result

The conventional and ABC-CNN results also illustrate why model selection should be based on validation behaviour rather than training fit alone. The baseline reached 95.89% final training accuracy, while the ABC-CNN reached 99.44%. If only training accuracy were considered, both models would appear highly successful. Their validation results tell a different story: the conventional CNN ended at 80.92% validation accuracy and the ABC-CNN at 84.66%, with maximum values of 82.04% and 85.20%, respectively. The optimization benefit is therefore visible primarily in how well the learned representation transfers beyond the training samples.

The same principle appears in the loss values. The baseline achieved a very low final training loss of 0.1320, but its validation loss rose to 1.1291 after having reached a lower value earlier in training. ABC-CNN reduced training loss further to 0.0156, yet the more important result is that its validation loss remained lower, ending at 0.7895 with a minimum of 0.7259. Thus, the optimized configuration did not merely fit the training data more strongly; it also improved the validation objective used to judge generalization during model development.

This distinction is important for automated hyperparameter search. An optimizer that rewards only training accuracy could favour increasingly expressive configurations that memorize the development data. In this study, candidate solutions were compared through validation performance, and the implementation additionally applied a penalty based on the training-validation gap during fitness evaluation. That design is consistent with the aim of selecting a configuration that balances fit and generalization rather than one that simply maximizes training accuracy.

The reported outcome also shows that optimization did not remove the need for regularization and stopping criteria. The ABC-CNN still exhibited a substantial training-validation separation, and validation loss stopped improving long before training loss approached zero. The use of dropout, early stopping and learning-rate reduction therefore remains relevant after hyperparameter selection. ABC should be understood as one component of the model-development process rather than as a replacement for conventional controls against overfitting.

From a reporting perspective, the strongest evidence for the optimizer comes from convergence of several indicators: higher validation accuracy, lower validation loss, higher test accuracy, precision, recall and F1-score, and lower FPR and FNR. Agreement across these measures reduces the likelihood that the conclusion depends on one isolated metric. At the same time, the recognition-time increase demonstrates that optimization can introduce a trade-off even when most performance measures improve. A complete evaluation should therefore retain both effectiveness and computational measures.

For future comparison with other optimization methods, the same experimental structure should be preserved. Competing optimizers such as random search, Bayesian optimization, particle swarm optimization or genetic algorithms should use the same search space, candidate-training budget, validation split and final training protocol. This would make it possible to determine whether the improvement is specific to ABC or reflects the broader benefit of automated hyperparameter search over a fixed baseline. Without such a matched comparison, the present study supports ABC over the reported ordinary CNN, but it does not establish ABC as the universally best optimizer for the problem.

G. Sensitivity of the conclusion across evaluation measures

The conclusion that ABC improved the classifier does not depend on a single performance measure.

Accuracy and recall both increased by 3.89 percentage points, precision increased by 3.27 percentage points and F1-score increased by 3.93 percentage points. These four measures describe related but different aspects of multiclass recognition. Their consistent direction indicates that the optimized model did not obtain a higher accuracy by sacrificing precision or recall; instead, the improvement was distributed across the major classification measures.

F1-score is particularly useful in this context because it summarizes the balance between precision and recall. The baseline F1-score of 80.66% increased to 84.59% after optimization. This gain is slightly larger than the corresponding precision increase and is comparable to the accuracy and recall gains. The result therefore suggests that the optimized classifier improved the balance of correct identity assignments rather than producing a narrow change confined to one side of the prediction process.

Error-rate measures provide an additional check on the same conclusion. If higher accuracy had been accompanied by a worse FPR or FNR, the practical value of the optimization would be less clear for authentication. Instead, both error measures declined. The agreement between higher classification scores and lower error rates strengthens the interpretation that ABC-CNN provides a generally better recognition profile under the reported test conditions.

Recognition time is the only evaluated measure that moved in an unfavourable direction. This is valuable rather than contradictory information because it exposes the cost associated with the improved recognition profile. A model-selection decision can therefore be framed explicitly: ABC-CNN offers approximately four percentage points of improvement in the principal classification measures and a substantial reduction in FNR, while adding 0.062 ms/sample to classifier inference. Whether that trade-off is acceptable ultimately depends on the latency budget of the target deployment.

H. Relationship with recent multimodal biometric studies

The current accuracy should not be compared mechanically with higher values reported in other multimodal studies. Kadhim and Abdulameer [10], for example, also studied face, palm and iris data from MULB, but the feature representations, classifiers and evaluation procedures were different. Likewise, SecureSense [9] used face, fingerprint and iris with decision-level fusion, while Alshardan et al. [7] and El_Rahman and Alluhaidan [8] employed different deep-learning architectures and biometric combinations. Differences in reported accuracy therefore reflect both algorithmic choices and experimental design.

For this reason, the most defensible contribution of the present article is the within-study comparison. Both models used the same dataset, identity classes, preprocessing and fused PCA-LDA input. The comparison therefore asks a narrower and more controlled question: given the same biometric representation, does ABC-guided CNN configuration improve the reported classifier? The answer from the present results is yes across the major classification and error measures, with the recognition-time trade-off already described.

This framing also separates the present article from a feature-fusion study. Weighted PCA-LDA fusion is treated here as a fixed common input, not as the independent variable. The experimental emphasis is the classifier-configuration stage. This distinction is important for publication because it prevents the optimization paper from duplicating the analysis of PCA-only, LDA-only and weighted-sum feature representations. Instead, the paper uses the selected fused representation as a controlled input while isolating the change from the ordinary CNN procedure to the ABC-guided CNN procedure.

The comparison with SwarmCNN [4] is similarly conceptual rather than numerical. Inik demonstrated that ABC and PSO can be used for CNN hyperparameter optimization, providing methodological support for swarm-based search. The present study applies ABC to a different problem structure: a 219-class multimodal biometric identification task based on fused face-palm-iris features. The significance of the current result lies in showing that the optimization principle remains useful in this biometric setting, not in claiming that its absolute accuracy should match results obtained on unrelated datasets or tasks.

Overall, the results position ABC as an effective automated search mechanism within the current multimodal pipeline. The evidence supports continued investigation of swarm-based tuning, but it also points to clear next comparisons: identical-budget evaluation against random search, Bayesian optimization and other population-based optimizers; ablation of individual hyperparameters; and end-to-end testing with deep learned features or adaptive fusion. These extensions would determine whether the current gain arises mainly from the chosen ABC search strategy, from automated tuning more generally, or from the interaction between optimization and the fixed fused feature representation.

VI. LIMITATIONS AND VALIDITY CONSIDERATIONS

The study was software based and did not include physical biometric sensors, edge devices, network transmission, smart locks or other smart-home hardware. Consequently, the reported recognition time represents model inference rather than total authentication latency. Real deployments may introduce image-capture delays, preprocessing costs, communication overhead, power constraints and environmental variation.

A second limitation concerns generalization. ABC optimization raised validation accuracy and lowered validation loss, but it did not eliminate the training-validation gap. Final training accuracy was 99.44% for the ABC-CNN compared with final validation accuracy of 84.66%. This residual gap indicates that hyperparameter search improved the model configuration without fully resolving overfitting or representation limitations. Additional regularization, stronger data augmentation, deeper learned features or adaptive fusion may be required for further gains.

The comparison should also be interpreted as the effect of the reported ABC-CNN development procedure rather than as an isolated causal estimate for one hyperparameter change. The optimized model used early stopping and learning-rate reduction in the final training procedure, whereas the conventional CNN was reported with a fixed 30-epoch training schedule. A stricter future ablation should train the baseline and optimized architectures under identical callbacks, epoch limits and stopping criteria so that the contribution of the ABC-selected configuration can be separated from the final training protocol.

Finally, the study reports only four paired executions. Although the resulting standard deviations were small and the reported paired tests were significant, a larger number of repeated runs would provide a stronger basis for inference and would permit more reliable confidence intervals, distributional checks and robustness analysis.

VII. CONCLUSION

This study evaluated the effect of Artificial Bee Colony optimization on a CNN classifier for multimodal face-palm-iris biometric authentication. The comparison used the same MULB dataset, common preprocessing and fused PCA-LDA input representation for the conventional CNN and ABC-CNN. The optimized model achieved higher mean accuracy, precision, recall and F1-score and lower FPR and FNR than the baseline. Accuracy increased by 3.89 percentage points and F1-score by 3.93 percentage points, while FNR decreased by 3.897 percentage points. The paired-samples analysis reported significant differences for all seven evaluated measures at $p < 0.001$.

The improvement was obtained with a 0.062 ms increase in classifier-level recognition time. Thus, the principal contribution of the optimization was a better recognition-error trade-off rather than faster inference. The remaining training-validation gap, software-only evaluation and small number of repeated runs show that further validation is needed. Future work should test identical training protocols for both classifiers, increase the number of independent executions, evaluate the optimized model on real smart-home edge hardware, and measure full end-to-end authentication latency under varying acquisition and environmental conditions.

REFERENCES

[1] D. Karaboga and B. Basturk, “On the performance of artificial bee colony (ABC) algorithm,” *Applied Soft Computing*, vol. 8, no. 1, pp. 687–697, 2008. Crossref

[2] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, 2020. Crossref

[3] B. Bischl et al., "Hyperparameter optimization: Foundations, algorithms, best practices and open challenges," *WIREs Data Mining and Knowledge Discovery*, vol. 13, no. 2, e1484, 2023. Crossref

[4] Ö. Inik, "SwarmCNN: An efficient method for CNN hyperparameter optimization using PSO and ABC metaheuristic algorithms," *The Journal of Supercomputing*, vol. 81, Art. 874, 2025. Crossref

[5] N. Alay and H. H. Al-Baity, "Deep learning approach for multimodal biometric recognition system based on fusion of iris, face, and finger vein traits," *Sensors*, vol. 20, no. 19, Art. 5523, 2020. Crossref

[6] Y. Wang, D. Shi, and W. Zhou, "Convolutional neural network approach based on multimodal biometric system with fusion of face and finger vein features," *Sensors*, vol. 22, no. 16, Art. 6039, 2022. Crossref

[7] A. Alshardan et al., "Multimodal biometric identification: Leveraging convolutional neural network architectures and fusion techniques with fingerprint and finger vein data," *PeerJ Computer Science*, vol. 10, e2440, 2024. Crossref

[8] S. A. El_Rahman and A. S. Alluhaidan, "Enhanced multimodal biometric recognition systems based on deep learning and traditional methods in smart environments," *PLOS ONE*, vol. 19, no. 2, e0291084, 2024. Crossref

[9] S. Juluri and G. Madhavi, "SecureSense: Enhancing person verification through multimodal biometrics for robust authentication," *Scalable Computing: Practice and Experience*, vol. 25, no. 2, pp. 1040–1054, 2024.

[10] O. N. Kadhim and M. H. Abdulameer, "Biometric identification advances: Unimodal to multimodal fusion of face, palm, and iris features," *Advances in Electrical and Computer Engineering*, vol. 24, no. 1, pp. 91–98, 2024. Crossref

[11] S. Pahuja and N. Goel, "Multimodal biometric authentication: A review," *AI Communications*, vol. 37, no. 4, pp. 525–547, 2024. Crossref

[12] P. Chitrapu, M. K. Morampudi, and H. K. Kalluri, "A secure and explainable multimodal biometric system using trust adaptive fusion for face and fingerprint," *Scientific Reports*, vol. 16, Art. 14244, 2026. Crossref

[13] J. S. Yalli, M. H. Hasan, T. J. Low, and S. M. Al-Selwi, "Authentication schemes for Internet of Things (IoT) networks: A systematic review and security assessment," *Internet of Things*, vol. 30, Art. 101469, 2024.