Al-Rafidain Journal of Engineering Sciences

Journal homepage https://rjes.iq/index.php/rjes

ISSN 3005-3153 (Online)

Architectural Innovations in CNNs for Robust Face Recognition Across Varied Lighting and Poses: A Comparative Performance Analysis

Ali H. H. Al-Amili[1], Fatin E. M. Al-Obaidi[1], Ali A. D. Al-Zuky[1]

[1]Mustansiriyah University, College of Science, Physics Department, Baghdad, Iraq

ARTICLE INFO

Article history:
Received 8 October 2025
Revised 8 October 2025,
Accepted ,5 November 2025
Available online 15 December 2025

Keywords:
Face recognition
convolutional neural networks
deep learning
computer vision
artificial intelligence

ABSTRACT

Face recognition (FR) is a fundamental task in computer vision with applications in security, healthcare, and human–computer interaction. Although convolutional neural networks (CNNs) have significantly advanced FR performance, existing systems remain highly sensitive to variations in illumination, pose, and image quality. Moreover, reliance on benchmark datasets alone often limits generalizability to real-world conditions. In this work, a customized lightweight CNN architecture was designed to enhance recognition accuracy under diverse lighting and pose variations. The approach integrates both the Labeled Faces in the Wild (LFW) dataset and a locally collected dataset, ensuring evaluation under benchmark and real-world conditions. A robust preprocessing pipeline—including cropping, normalization, and augmentation—further strengthens the model's generalization. To avoid undertraining, model optimization was guided by validation loss and early stopping rather than fixed epoch counts. Experimental results show that the proposed model achieves 99.67% accuracy on the local dataset and 93.33% accuracy on LFW, with a compact model size of only 117 MB. In addition, the proposed CNN requires 33.63M parameters and 0.73 GFLOPs, which is substantially lower than ResNet101 (42M, 3.27 GFLOPs) and VGG-16 (134M, 3.09 GFLOPs), highlighting its efficiency in terms of both model size and computational complexity. Compared with state-of-the-art (SOTA) architectures such as ResNet101, GoogLeNet and VGG-16, the customized CNN delivers a favorable trade-off between accuracy, efficiency, and computational complexity. These results demonstrate that carefully designed lightweight CNNs, when combined with local and public datasets, can achieve robust face recognition in unconstrained environments, making them suitable for deployment in resource-limited real-world applications.

1. Introduction

Face recognition (FR) technology continues to evolve dynamically with ongoing research and development. Reliable personal identity authentication is critical to protecting sensitive information. This interdisciplinary technology has numerous applications and is utilized in various sectors, including security, human-robot interaction, genetic disorder diagnosis from facial features and appearance deformities [1], [2]. The protection of personal identity authentication methods has declined, and incidents such as forged IDs have occurred frequently. However, FR systems surpass traditional recognition methods in speed and efficiency, which is nearly instantaneous identification. It also offers a strong element by examining a whole feature in the face, making it a very secure method in many applications. Moreover, these systems' automation lowers the need for human intervention, reducing

Corresponding author E-mail address: alihh@uomustansiriyah.edu.iq
https://doi.org/10.61268/6tfgm814

operating expenses and human error, despite progress, FR still degrades under variable and making them a proper choice for numerous applications. Moreover, the automation of FR systems reduces human intervention, operating cost, and human error. Despite such progress, FR performance still degrades under variable illumination and pose, which motivates the present work [3]. FR exhibits prominent advantages in particular scenarios, such as criminal identification. An FR system does not require expensive equipment, in which only camera devices can collect facial images to train the model using a computer and then the system is ready to serve in real-world application [4]. However, FR still has shortcomings, such as the lighting conditions either low or strong light environments, data collection issues from privacy and errors associated with labelling the samples, and the complexity of creating the right data for a specific project [5]. Since the 1960, numerous FR techniques have been developed, and these techniques are often categorized as shallow learning methods since they require artificial experience to extract sample features and can only utilize a few fundamental features of images. On the other hand, Neural network-based techniques can extract more complex characteristics, like corner, edge, and texture information, and are regarded as deep learning techniques [6]. CNNs play an important role in the development of FR technology because they can efficiently extract facial features from images. A large amount of data is required to train this network. The size and quality of the training data have a critical impact on the performance of CNNs. Most datasets are limited in quantity and quality; the data may not be sufficient to achieve the convergence required to train an accurate model. The data may contain errors such as mislabeling or image quality deviations. Despite these issues, CNN-based FR methods have clear advantages. These networks independently extract complex features from labeled data, and eliminating the need for manual feature engineering [7]. CNNs provide translation equivariance and, with pooling, limited translation invariance (not full scale invariance), allowing faces to be recognized at different scales despite differences in location and size. This feature makes CNNs ideal for dealing with variations in facial orientation and expression. Recent innovations in CNN architectures (e.g., AlexNet, GoogLeNet, and ResNet), have improved the accuracy of FR systems, which demonstrate exceptional performance in diverse and challenging environments such as complex illumination, changing facial expressions, and partial occlusion. As a result, CNN-based FR has become an important research area, and significant progress has been made in both theoretical models and practical applications [8].

However, FR, as a field of computer vision and pattern recognition, has been extensively studied in the past few years as one of the most active research areas in artificial intelligence, where the system designed starts by detecting a face and then recognizing it . Researchers have developed several recognitions approaches to capture discriminative features. Traditional techniques typically include two processes: high-dimensional feature extraction and classifier design. On the other hand, CNN models automatically combine the classifier and feature extractor in an end-to-end manner [9], which significantly advanced FR technology. CNN can be adapted to various challenges to the real-world applications. In comparison to traditional FR techniques, CNN models consistently outperform them [10]. Researchers have adopted various strategies to optimize CNNs. For example, an efficient hybrid multi-layer CNN combined with Support Vector Machines (SVM) enhances FR by handling diverse datasets and reaches an accuracy of 99.87% [11]. Moreover, a proposed CNN model called RobFaceNet achieved balance by incorporating multiple features and attention mechanisms, achieving 95.95% and 92.23% on the CA-LFW and CP-LFW datasets, respectively, compared to 95.45% and 92.08% for the very deep ArcFace model [12]. Regarding low-resolution image issues, Mishra et al. [13] have utilized multiscale parallel deep CNN architectures to tackle the difficulties associated with low-resolution images of faces, offering a solution

that enhances accuracy in surveillance applications. Furthermore, the authors in [14] have designed an algorithm for low-resolution images, a crucial aspect of surveillance applications, in which it focuses on optimizing CNN classifiers to handle the challenges of pose and low-resolution imaging. Recent literature on DCNNs highlights considerable success in static conditions, yet struggles in dynamic environments remain prevalent. Furthermore, there is no real consideration for the computational challenges. This paper critically examines these shortcomings, particularly in adaptability to lighting and pose variations, and introduces an innovative approach that effectively addresses these challenges. Moreover, to increase the enhanced reliability, a local dataset has been created and compared with the LFW dataset in various scenarios. The design proficiently achieves FR in unconstrained settings, addressing challenges reported by prior methods. Beyond ResNet-101, GoogLeNet, and VGG-16, state-of-the-art (SOTA) face recognition models have been reported in literature to achieve superior verification accuracy on large-scale datasets. While these were not re-implemented in this study, their results are discussed to contextualize the performance of the customized CNN.

Despite remarkable progress, existing CNN-based face recognition systems still face challenges in uncontrolled environments, particularly under extreme variations in lighting and pose. Moreover, most studies rely heavily on benchmark datasets while neglecting the importance of integrating local, real-world data, which limits generalizability. Addressing these gaps, this paper proposes a lightweight, customized CNN framework specifically optimized for robust performance under diverse illumination and pose conditions. The proposed approach combines public (LFW) and locally collected datasets, advanced preprocessing, and systematic architectural refinements to balance accuracy, efficiency, and robustness.

The main contributions of this work are summarized as follows:

Customized CNN Design: Development of a lightweight deep CNN architecture that integrates convolutional, normalization, pooling, and dropout layers in a systematic manner to improve robustness against lighting and pose variations.

Integration of Local and Public Datasets: Novel use of both the Labeled Faces in the Wild (LFW) dataset and a locally collected dataset, ensuring evaluation and training under real-world conditions.

Advanced Preprocessing Pipeline: Implementation of preprocessing techniques (cropping, normalization, augmentation, noise addition) to enhance model generalization while reducing sensitivity to illumination and pose variability.

Comprehensive Experimental Analysis: Comparative evaluation of the proposed model against SOTA architectures (ResNet101, GoogLeNet, VGG-16), supported by detailed metrics (accuracy, TPR, FLOPs, parameters, and inference/training times).

Efficiency-Oriented Training Strategy: Adoption of validation-guided training (early stopping) instead of fixed epochs, ensuring convergence while reducing overfitting and computational overhead.

The arrangement of this research is as follows: Section 2 describes the tools and methodologies of CNN design and optimization. Section 3 presents the experimental results which demonstrates the effectiveness of the proposed approach. Finally, Section 4 presents the conclusions and outlines directions for future work.

2. Tools and Methodology

The methodology of the FR system is explained in this part. The approach integrates advanced machine learning techniques with a customized DCNN design. In addition, an effective combination of preprocessing methods (resizing, normalization, cropping, augmentation) reduces sensitivity to lighting and pose variations and enhancing its application breadth. The description of the hardware and software configurations is as follows:

2.1 Hardware and Software Utilized for Recognition System

A laptop with the following hardware was used for the experiments: Processor: 13th Gen Intel(R) Core (TM) i7-13620H, 2.40 GHz; RAM: 16.0 GB; GPU: NVIDIA GeForce RTX 3060, 6GB. The software tool used for implementing and testing the FR system was MATLAB R2023b. This version of MATLAB has an environment for development and implementation, including the Deep Learning Toolbox for building and training the CNN model and other add-ons, such as the Image Processing Toolbox, were utilized for data preprocessing and visualization. Also, the Vision Toolbox's \vision.CascadeObjectDetector was used. The study used neural network functions to train and validate CNN models. MATLAB's video processing tools extracted frames from the smartphone's captured videos. Face data were captured using a Realme smartphone with a camera resolution of 12.5 MP (4:3 aspect ratio), and the video recording was HD 1080p, 30 fps.

2.2 Data Preparation

2.2.1 The Local Dataset

The local dataset for training and validation was built by capturing videos of five individuals, each no longer than one minute, under different lighting and facial pose conditions [15]. The camera was positioned to replicate real-world scenarios, capturing frontal and slightly angled facial views. Next, the frames were extracted from videos using MATLAB's VideoReader function and images were saved per subject for training and testing. Figure 1(a) shows a sample of the locally created dataset.

Data Augmentation (DA) is a prevalent and crucial preprocessing technique for CNN to reach significant performance. A MATLAB code has been designed to perform DA. It applies various transformations to create new versions of the original images, thereby increasing the size and diversity of the dataset up to four times. Figure 1(b) shows a sample of a dataset with various augmentation methods. Augmentation was applied to the training set only; test sets remained unchanged.

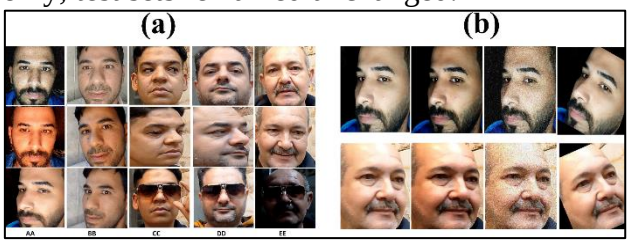

Figure 1. (a) Sample of Locally Created Dataset Before the Preprocessing, and (b) Sample of a Pre-processed Local Dataset, Including Changing in Lighting Condition, Adding Noise and Rotation [15]

2.2.2 Public dataset

The LFW dataset is among the most widely used benchmarks for unconstrained face recognition [16]. It contains over 13,000 labeled images collected from the web and features a diverse group of people photographed under a variety of conditions. Given its inherent diversity in poses, lighting, and facial expressions, this dataset is ideal for evaluating labeled face systems in the wild. For this research, a subset of the LFW dataset was selected and customized to make it suitable for the proposed system in which the identities with ≥75 images have been retained, yielding 5 identities and 375 total images. Selection criteria and final counts are reported to ensure reproducibility. The selection of the images was random for each person, ensuring variability in lighting conditions, facial poses, and expressions. Preprocessing steps were applied to ensure consistency with the CNN input and compatibility with the model. The dataset was split into two subsets: a training set (75%) and a validation set (25%) for evaluating the model's performance with a confusion matrix. To further enhance the dataset's variability and robustness, DA was applied to the training set only; the test set remained untouched to ensure an unbiased evaluation. Using a public and recognized dataset enhances the credibility of the research and provides a benchmark for comparing it with the local one that created for the proposed system with existing methods. Figure 2 (a) shows a sample of the public dataset LFW, and Figure 2 (b) shows a sample of LFW dataset with various augmentation methods.

Figure 2. (a) Sample of Public Dataset (LFW) Before the Preprocessing, and (b) Sample of a Pre-processed Public Dataset, Including Changing in Lighting Condition, Adding Noise and Rotation [16]

2.3 FR system and Network Design

2.3.1 Steps of FR System

The proposed FR system is designed to reliably identify individuals under diverse lighting and pose conditions. Figure 3 illustrates the pipeline of the system. A video stream is first captured by a standard camera. From this stream, facial regions are detected using a cascade object detector. The detected face images undergo preprocessing steps, including cropping, normalization, resizing, and data augmentation, to reduce sensitivity to environmental variations.

The preprocessed facial images are then passed into the customized CNN model. The CNN maps each face to a high-dimensional feature embedding, which is subsequently compared either through Softmax classification (for closed-set recognition) or embedding similarity (cosine distance) for verification tasks. This ensures that recognition is not solely dependent on classification but can generalize across identities. The system is designed for deployment in secured authentication scenarios where only authorized individuals are recognized; the term "secured area" refers to application domains such as restricted office entry, laboratory access, or device unlocking.

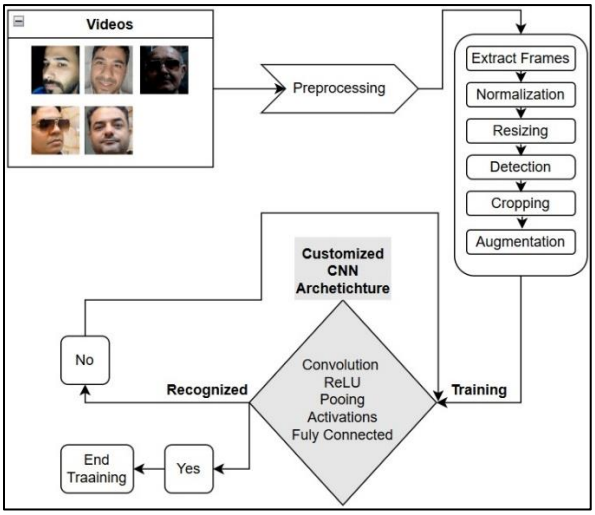

Figure 3. The Flowchart of the Proposed Face Recognition System Showing the Most Important Layers in the Customized Model

2.3.2 Designing the Network

The proposed DCNN was developed using a stepwise design methodology. This means that layers were incrementally added, and their impact on performance was carefully evaluated. The initial architecture began with a single convolutional layer, followed by max pooling. Successive modifications introduced multiple convolutional blocks, batch normalization, and dropout layers to improve stability and prevent overfitting. This systematic, step-by-step design ensured that each architectural addition was performance-driven, striking a balance between accuracy, computational cost, and robustness. The architecture targets complex facial patterns, aiming to improve accuracy and robustness under unconstrained conditions. The network design is inspired by traditional DCNN architectures like AlexNet, and was modified to meet task-specific requirements while remaining lightweight.

The final model consists of 14 layers:

Convolutional blocks (Conv3 with 64, 128, and 256 filters) for hierarchical feature extraction.

Batch normalization and ReLU activations to stabilize and accelerate training.

Max-pooling layers to reduce spatial complexity while preserving discriminative features.

A fully connected layer (256 units) to integrate extracted features.

This layer joins the features that have been collected to a high-dimensional vector. This is the last representation that is used before classification. The mathematical functions of this layer are as follows:

$$Z = W^T X + b \quad (1)$$

where X is the input vector, W is the weight matrix and b is the bias term.

A dropout layer (p = 0.5) to improve generalization.

This layer was added to mitigate overfitting, which randomly disables 50% of the neurons during training, improving the generalization ability of the model. Finally, classification is performed using a Softmax layer, which computes the probability distribution over all classes for each input. The Softmax function is defined as follows:

$$P(y=k|x) = \frac{e^{k^z}}{\sum_j^K e^{j^z}} \quad (2)$$

where P(y=k|x) is the probability of class k and kz is the output for the k-th class. This robust architecture provides efficient and accurate FR by leveraging SOTA CNN components and techniques to deliver high performance in a variety of settings. A Softmax classifier for identity recognition across multiple classes [17], [18].

2.4 Evaluation Metrics

To assess the performance of proposed model, several key metrics were used. These metrics are essential for understanding different aspects of model accuracy and responsiveness:

Recall (R): Is the ratio of true positive results to the total number of cases that are actually positive. It measures the model's ability to detect all relevant instances.

$$R = TP / (TP + FN) \quad (3)$$

Precision (p): This metric highlights the accuracy of the positive predictions made by the model, crucial for applications where false positives carry a significant cost.

$$p = TP / (TP + FP) \quad (4)$$

F1-score balances precision and recall of the model, and provide a single metric summarizes model performance when both false positives and false negatives are in concern.

$$F_1 = \frac{2PR}{PR} \quad (5)$$

Where: TP (true positives), FP (false positives), FN (false negatives), TN (true negatives).

3. Results and Discussion

In the following section, a set of experiments is conducted to analyze the CNN configurations. The experimental analysis was conducted in three successive phases. First, different CNN architectures were explored by varying the number and composition of layers to determine the configuration that yielded the best performance. Second, once the architectural design was fixed, a systematic investigation of key hyperparameters (e.g., batch size, learning rate, number of epochs, and input resolution) was carried out to further optimize the model. Finally, the best-performing customized CNN obtained through these two phases was benchmarked against SOTA architectures such as ResNet101, GoogLeNet, and VGG16 in order to assess its relative accuracy, efficiency, and computational complexity. This stepwise procedure ensured a fair and transparent evaluation of both the internal design choices and the external competitiveness of the proposed model.

To ensure robustness, the local dataset was also incorporated into the training phase alongside LFW dataset. This arrangement exposed the network to both public and local dataset and study the effect of each dataset. Furthermore, model training was guided by validation loss curves with early stopping, rather than a fixed number of epochs, to guarantee proper convergence and to prevent undertraining.

3.1 The Impacts of CNN Layers

The performance of different CNN architectures was compared by varying the number of layers and evaluate their impact on FR. The models were trained using the LFW dataset, which includes preprocessing techniques. However, various metrics were

analyzed, such as validation accuracy, precision, and elapsed time, for the training performance with a fixed hyperparameter (Mini Batch Size = 32, initial Learn Rate =0.0001, maxEpochs = 5, and Image Size 64×64, as shown in table 1, where True Positive Rate (TPR) = (TP / (TP + FN)).

Table 1. Performance of customized CNN architectures with varying numbers of layers on the LFW dataset, with a Fixed hyperparameter for all models

Model	Customized-A	Customized-B	Customized-C
Number of Layers	7	14	14
Validation Accuracy (%)	87.67	83.00	92.00
Elapsed time	10 sec	14 sec	18 sec
Precision	0.7936	0.6772	0.8000
TPR	0.696	0.68	0.794

The customized-A model displays a basic CNN confirmation with only 7 layers as following: [input + convolution (Conv3, 16) + Batch Normalization (BN) + Rectified Linear Unit (ReLU) + Fully connected (FC) + Softmax + output Classification] resulted a validation accuracy of 87%, demonstrating rapid learning and good generalization, and the stable losses indicate minimal overfitting. The confusion matrix shows that the model does not always generalize well across categories. This allows relatively accurate predictions to be made for certain classes, i.e., predictions are correct, but predictions for other classes turn out to be more or less accurate.

In the particular customized-B model, an additional Conv(3, 32) is added along with an extra layer to capture more complex features. This increases the number of layers to 14 and improves accuracy and robustness. Results showed that efficient and fast training was achieved with a validation accuracy of 83%. Improvements to feature detection or class-specific changes may be needed, as evidenced by the confusion matrix, which shows only moderate accuracy and significant differences between many classes.

Finally, the Customized-C model (Figure 4) retains the same depth as Customized-B but increases the number of convolutional filters, yielding improved validation accuracy. This model achieved a validation accuracy of 92%, showing a stable and improving performance trend over the training epochs. The model's precision has improved across multiple classes when compared to the earlier versions. Nearly all individuals have identified within an improved accuracy in identifying all classes, according to confusion matrix. However, the Models A–C varied multiple components jointly; a one-factor-at-a-time ablation is left for future work.

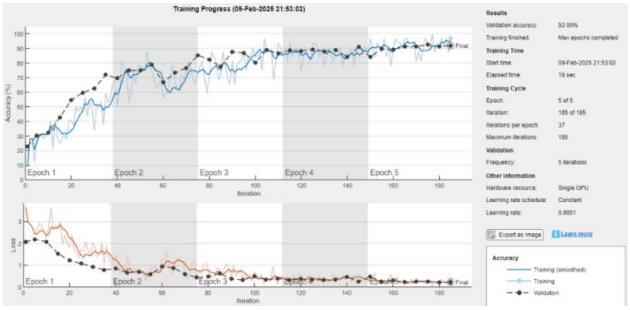

Figure 4. The Training Progress of the Proposed Model the Customized-C

The customized models (A, B, and C) were designed to explore the combined effect of architecture depth and regularization components. While this introduces some inconsistency, the goal was to reflect realistic CNN design choices where multiple components are tuned together. To address the undertraining concern, the training of the best-performing customized-C model was extended to 20 epochs, and the updated loss curves confirmed convergence, strengthening the reliability of the reported accuracy as discussed in the next sections.

3.2 The Impact of Hyperparameter Tuning on the Model Performance

To advance FR capabilities, it is crucial to show how adjusting training hyperparameters affects the performance of Customized-C model that obtained from table 1 with fixed its layer. It is important to note that the hyperparameter tuning experiments presented in this section were not designed to isolate the individual impact of each hyperparameter.

Instead, the adjustments were made jointly in an exploratory manner with the primary objective of empirically identifying the best-performing configuration of the Customized-C model. This optimal configuration was then selected as the reference design for subsequent comparisons with SOTA architectures. However, the tuning of model hyperparameter is essential not only for improving accuracy but also for adapting the model to diverse environments, ensuring robust performance under varying conditions.

On one hand, table 2 illustrates the impact of hyperparameter Adjustments on LFW dataset and it reports an exploratory hyperparameter–tuning sequence performed on the fixed Customized-C architecture. The intent was not to isolate the effect of each hyperparameter but to empirically identify a high-performing configuration to be used in subsequent comparisons with SOTA models. Starting with C1-public, the model exhibited clear underfitting (41.67% accuracy) due to very few epochs and a small batch size. Progressive increases in epochs, batch size, and input resolution (C2-public –C3-public) substantially improved learning stability and accuracy, while further refinements through lower learning rates and extended training cycles (C4-public –C5-public) enhanced generalization. The final configuration, Customized-C5-public, achieved the highest validation accuracy of 93.33%, demonstrating that incremental and combined tuning of key hyperparameters can yield a more robust and reliable model. This optimized version was therefore selected as the reference design for subsequent benchmarking against state-of-the-art models. The results presented in the table above show the impact of incremental changes in model parameters on the efficiency of the FR system. Starting from modified model C1-public, the validation accuracy increases significantly as the epoch increases and other parameters are adjusted. With an improved strategy combining higher image quality and lower learning rates, the highest validation accuracy was achieved 93.33% by modifying customized-C5-public model. Extended training and improved FR capabilities are strongly correlated, and each subsequent change in training parameters (such as larger epochs and mini-batch size) produces more precise and accurate model results.

Table 2. Impact of hyperparameter adjustments (epochs, batch size, learning rate, and image resolution) on the Customized-C model using the LFW dataset. Note: hyperparameters were adjusted jointly in an exploratory manner to identify a best-performing configuration (Customized-C5); this table is not intended as a one-factor ablation.

Parameter	Customized-C1-public	Customized-C2-public	Customized-C3-public	Customized-C4-public	Customized-C5-public	No. of Layers
MaxEpochs	2	4	20	16	20	Input + Conv (3, 64)+ BN + ReLU+ MaxPool + Conv (3, 128)+ BN + ReLU+ MaxPool + FC (256) + Dropout(0.5)+ FC + Softmax + Output Classification layer = 14
Mini batch Size	16	32	64	128	64	
Initial Learn Rate	0.01	0.001	0.0001	0.00001	0.0001	
Image Size	16 x 16	32 x 32	64 x 64	256 x 256	128 x 128	
Validation Accuracy %	41.67	90.33	92.00	81.33	93.33	
Elapsed time	7 sec	13 sec	18 sec	1 min 29 sec	1 min 40 sec	
Precision	0.35418	0.83818	0.76956	0.67543	0.83484	

On the other hand, table 3 shows the impact of hyperparameter adjustments in the same arrangement as the previous table 2, but using the local dataset to compare the performance of the resultant model in two different datasets. Analyzing two result models reveals a significant performance discrepancy. For validation accuracy and precision, the local dataset consistently exhibits a good performs comparing to the LFW dataset under comparable experimental conditions, and to avoid subject leakage, all frames from a given identity were kept within the same partition. However, the higher classification accuracy of the local dataset compared to the LFW dataset

is likely due to the homogeneity in the characteristics of the images within the local dataset. This high accuracy is advantageous for applications targeting similar image sets, in which models trained on homogeneous local datasets perform well under comparable conditions but require further validation to avoid overfitting before deployment in diverse settings. In addition, the specific adaptation of the model to the characteristics inherent in the local dataset can be another because of its superior accuracy. Since the images were captured from a controlled set of subjects, the model is more effectively learning distinctive features specific to demographic groups, and it can enhance model performance due to reduced intra-class variability and focused learning on relevant features. However, there is a risk of overfitting, where models may not generalize well to new or diverse data. especially for the local dataset. On the other hand, a strict choice of images was adopted by considering the wide variety of lighting and angles. In addition, a preprocessing was used such as cropping faces, which reduced the radiant and unnecessary background, and augmentation, which offered more generalities in the training process.

Table 3. Performance of the Customized-C model on the local dataset under identical hyperparameter variations as Table 2. Results show consistently higher accuracy due to dataset homogeneity

Model	Customized-C1-local	Customized-C2-local	Customized-C3-local	Customized-C4-local	Customized-C5-local	No. of Layers
Max Epochs	2	4	8	16	20	Input + Conv (3, 64)+ BN + ReLU+ MaxPool + Conv (3, 128)+ BN + ReLU+ MaxPool + FC (256) + Dropout(0.5)+ FC + Softmax + Output Classification layer = 14
Mini batch Size	16	32	64	128	64	
Initial Learn Rate	0.01	0.001	0.0001	0.00001	0.0001	
Image Size	16 x 16	32 x 32	64 x 64	256 x 256	128 x 128	
Validation Accuracy %	92.00	100	99.00	97.33	99.67	
Elapsed time	20 sec	20 sec	26 sec	1 min 40 sec	1 min 47 sec	
Precision	0.920	1.000	0.990	0.9733	0.9966	

Overall, tables 2 and 3 report exploratory experiments where multiple hyperparameters (epochs, batch size, learning rate, image resolution) were varied jointly to simulate practical tuning scenarios. This approach highlights the sensitivity of the model to compound adjustments. A systematic ablation study, in which one hyperparameter is varied at a time, can be done for future work to provide isolated insights.

3.3 Comparison of Customized Model with SOTA Architectures

Table 4 presents the comparative evaluation of the customized CNN model customized-C and the pre-trained models, including ResNet101, GoogLeNet, and VGG-16. To standardize complexity reporting, model size (MB), number of trainable parameters, and FLOPs were reported. The customized model with just 14 layers and a size of 29.9 MB, reached an accuracy of 93.33% on the LFW dataset, showing that its simpler design and specific preprocessing methods work well together.

The deeper pre-trained models, including ResNet101 (with 101 layers), shows a significantly low accuracy 65.67% where increasing the model depth does not guarantee better results, especially in limited-class situation. Also, GoogLeNet underperformed the customized model with an accuracy of (82.00%). On the other hand, VGG-16 's accuracy reached up to (96%),with a very large model size (953 MB), it suffered notable computational expense.

When the customized-C model retrained with a larger images size 224×224, the accuracy went down to 89.67%, which gave an indication of the importance to change the model's design when changing the input data size accordingly.

This comparison shows that a well-made, task-focused CNN model, when paired with good preprocessing and careful adjustment of settings, can match or even do better than much bigger pretrained models in accuracy and efficiency.

Table 4. Comparative evaluation of the customized-C CNN against SOTA models (ResNet101, GoogLeNet, VGG-16). Model size is reported in MB. The training parameters and FLOPs for each model was calculated

No	Model	Parameters (Millions)	FLOPs (Billions)	Number of layers	Model Size on the Desk (MB)	Accuracy (%)	Hyperparameters
1	Customized-C	33.630	0.73	14	117	93.33	MaxEpochs=20 Mini batch Size=64 Initial LearnRate=0.0001 Image Size=[128 128]
		102.83	2.23	14	363	89.67	MaxEpochs=20 Mini batch Size=64 Initial Learn Rate=0.0001 Image Size=[224 224];
2	ResNet101	42.450	3.27	101	303	65.67	
3	GoogLeNet	5.9786	3.00	22	043	82.00	
4	VGG-16	134.28	3.09	16	528	96.00	

4. Conclusion and future work

This study presented a customized CNN architecture for face recognition under challenging conditions of varying illumination and pose. By systematically refining the network design and integrating advanced preprocessing, the proposed model achieved strong performance while maintaining a compact size of only 117 MB. When trained jointly on both the LFW and the locally collected dataset, the model achieved up to 99.67% accuracy on local data and 93.33% accuracy on LFW, demonstrating its ability to generalize across different environments. Compared with SOTA architectures such as ResNet101, GoogLeNet, and VGG-16, the customized CNN achieved a favorable balance between accuracy, efficiency, and computational cost. The findings highlight three key insights. First, integrating locally collected data alongside public benchmarks enhances robustness and provides realistic evaluation conditions. Second, carefully designed lightweight CNNs can achieve accuracy levels comparable to deeper models while being more efficient in terms of FLOPs, parameters, and training time. Third, guiding training with validation-based early stopping ensures proper convergence and prevents undertraining, as confirmed by loss curve analysis. Future work will explore attention mechanisms, transformer-based blocks, and cross-modal learning to further enhance robustness in unconstrained environment. These enhancements could further strengthen

adaptability to unconstrained environments, making the system more resilient for real-world deployment in security, healthcare, and other AI-driven applications.

References

[1] J. Qiang, D. Wu, H. Du, H. Zhu, S. Chen, and H. Pan, "Review on Facial-Recognition-Based Applications in Disease Diagnosis," Jul. 01, 2022, MDPI. doi: 10.3390/bioengineering9070273.

[2] H. L. Gururaj, B. C. Soundarya, S. Priya, J. Shreyas, and F. Flammini, "A Comprehensive Review of Face Recognition Techniques, Trends, and Challenges," IEEE Access, vol. 12, pp. 107903–107926, 2024, doi: 10.1109/ACCESS.2024.3424933.

[3] G. Gao, Y. Yu, J. Yang, G. J. Qi, and M. Yang, "Hierarchical Deep CNN Feature Set-Based Representation Learning for Robust Cross-Resolution Face Recognition," IEEE Transactions on Circuits and Systems for Video Technology, vol. 32, no. 5, pp. 2550–2560, May 2022, doi: 10.1109/TCSVT.2020.3042178.

[4] A. Nemavhola, C. Chibaya, and S. Viriri, "A Systematic Review of CNN Architectures, Databases, Performance Metrics, and Applications in Face Recognition," Feb. 01, 2025, Multidisciplinary Digital Publishing Institute (MDPI). doi: 10.3390/info16020107.

[5] Z. Chen, "A study on the CNN-based face recognition across various facial angles," Applied and Computational Engineering, vol. 35, no. 1, pp. 265–271, Feb. 2024, doi: 10.54254/2755-2721/35/20230404.

[6] Shanshan Guo, Shiyu Chen, and Yanjie Li, "Face recognition based on convolutional neural network and support vector machine," Proceedings of the IEEE International Conference on Information and Automation (ICIA), Ningbo, China, pp. 1787–1792, Aug. 2016. DOI: 10.1109/ICInfA.2016.7832086

[7] X. Zhou and T. C. Zhu, "Survey of Research on Face Recognition Methods Based on Depth Learning," in Journal of Physics: Conference Series, Institute of Physics, 2024. doi: 10.1088/1742-6596/2717/1/012027.

[8] V. K. N. Kamlesh Pai, Sachinkumar Mogaveera, Manoj Balraj, and Deepak Aeloor, "Face recognition using convolutional neural networks," Proceedings of the 2nd International Conference on Trends in Electronics and Informatics (ICOEI 2018), pp. 165–170, IEEE, 2018. DOI: 10.1109/ICOEI.2018.8553720

[9] Susanta Malakar, Werapon Chiracharit, and Kosin Chamnongthai, "Masked Face Recognition with Generated Occluded Part using Image Augmentation and CNN Maintaining Face Identity," IEEE Access, vol. XX, pp. 1–9, 2017, doi:

[10] K. H. Teoh, R. C. Ismail, S. Z. M. Naziri, R. Hussin, M. N. M. Isa, and M. S. S. M. Basir, "Face Recognition and Identification using Deep Learning Approach," in Journal of Physics: Conference Series, IOP Publishing Ltd, Mar. 2021. doi: 10.1088/1742-6596/1755/1/012006.

[11] A. S. Darma, F. S. Mohamad, O. A. Diekola, and I. M. Sulaiman, "Deep Learning Approach for Face Recognition Based on Multi-Layers CNN&SVM," International Journal of Engineering Trends and Technology, vol. 71, no. 8, pp. 388–409, Aug. 2023, doi: 10.14445/22315381/IJETT-V71I8P234.

[12] A. Khalifa, A. A. Abdelrahman, T. Hempel, and A. Al-Hamadi, "Towards efficient and robust face recognition through attention-integrated multi-level CNN," Multimed Tools Appl, vol. 84, no. 14, pp. 12715–12737, Apr. 2025, doi: 10.1007/s11042-024-19521-0.

[13] N. K. Mishra, M. Dutta, and S. K. Singh, "Multiscale parallel deep CNN (mpdCNN) architecture for the real low-resolution face recognition for surveillance," Image Vis Comput, vol. 115, Nov. 2021, doi: 10.1016/j.imavis.2021.104290.

[14] S. S. Rajput and K. V. Arya, "CNN Classifier based Low-resolution Face Recognition Algorithm," in 2020 International Conference on Emerging Frontiers in Electrical and Electronic Technologies, ICEFEET 2020, Institute of Electrical and Electronics Engineers Inc., Jul. 2020. doi: 10.1109/ICEFEET49149.2020.9187001.

[15] Ali H. H. Al-Amili, "Ali's Face Recognition Dataset for CNN Training," GitHub Repository GitHub Repository, Available at: https://github.com/AlAs3000/FaceRecognitionDataset.git. Accessed: Jul. 07, 2025. [Online]. Available: https://github.com/AlAs3000/FaceRecognitionDataset.git

[16] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments." [Online]. Available: http://vis-www.cs.umass.edu/lfw/.

[17] S. J. Shahbaz, A. A. D. Al-Zuky, and F. E. M. Al-Obaidi, "Real-Night-time Road Sign Detection by the Use of Cascade Object Detector," Iraqi Journal of Science, vol. 64, no. 6, pp. 4064–4075, 2023, doi: 10.24996/ijs.2023.64.6.43.

[18] S. J. Shahbaz, A. A. D. Al-Zuky, and F. E. M. Al-Obaidi, "(2022) The Evaluation of Cascade Object Detector in Recognizing Different Samples of Road Signs".