

Design and Implementation of a Lightweight CNN-Based Face Recognition Attendance System on FPGA

Jing Wen Tan¹, Nabihah Binti Ahmad^{1,2}, Chessda Uttraphan Eh Kan^{1,2}, and Suhaila Isaak³

¹Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, Batu Pahat, Johor, Malaysia.

²VLSI and Embedded Technology (VEST) Focus Group, Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, Batu Pahat, Johor, Malaysia.

³Division of Electronic and Computer Engineering, Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Johor Bahru, Johor, Malaysia.

KEYWORDS

Face Recognition
Lightweight CNN
FPGA
Attendance System
Fixed-Point Quantization

ARTICLE HISTORY

Received 25 July 2026
Accepted 6 October 2026
Available online 20 October 2026

ABSTRACT

Low-cost Field-Programmable Gate Array (FPGA) implementation of convolutional neural network (CNN)-based face recognition remains challenging because of the limited computational resources and memory available on embedded hardware platforms. This study proposes a lightweight CNN-based face recognition attendance system implemented on a Cyclone IV EP4CE115 FPGA to achieve efficient real-time operation. The developed embedded platform combines live image acquisition using an OV7670 camera, fixed-point CNN inference, Video Graphics Array (VGA) visualization, and Universal Asynchronous Transmitter (UART)-based attendance logging within a single hardware system. The recognition model was trained using 590 facial images representing 42 identity classes, while data augmentation increased the number of training samples to 1,652. Software evaluation produced a validation accuracy of 94.32% and a testing accuracy of 98.88% before the trained network was converted into a fixed-point representation for hardware deployment using SystemVerilog. Register Transfer Level (RTL) verification demonstrated a classification accuracy of 97.62%, with prediction results remaining fully consistent with those generated by the Python fixed-point reference model. The implemented FPGA system operated at 1.86 frames per second with an estimated power consumption of 171.45 mW. The experimental results confirm that the proposed architecture provides a practical balance between recognition accuracy, hardware efficiency, and energy consumption, demonstrating the feasibility of deploying lightweight CNN-based attendance monitoring systems on resource-constrained FPGA devices.

© 2026 The Authors. Published by Penteract Technology.

This is an open access article under the CC BY-NC 4.0 license (<https://creativecommons.org/licenses/by-nc/4.0/>).

1. INTRODUCTION

Face recognition emerged as one of the most widely adopted biometric technologies for automatic identity verification because it provides a contactless, convenient, and reliable authentication mechanism [1]. In recent years, face recognition has been increasingly applied to attendance monitoring systems, replacing conventional methods such as manual sign-in sheets, Quick Response (QR) codes, and Radio Frequency Identification (RFID) cards. Although these

methods are simple to implement, they remain vulnerable to proxy attendance and identity misuse because they rely heavily on user participation rather than biometric verification.

The rapid development of convolutional neural networks (CNNs) has significantly improved the performance of face recognition by enabling automatic extraction of discriminative facial features from training data [1]. Compared with traditional handcrafted feature extraction methods, Sutabri et al. [2] reported that CNN-based approaches provide higher

*Corresponding author:

E-mail address: Nabihah Binti Ahmad <nabihah@uthm.edu.my>.

<https://doi.org/10.56532/mjsat.v6iS2.814>

2785-8901/ © 2026 The Authors. Published by Penteract Technology.

This is an open access article under the CC BY-NC 4.0 license (<https://creativecommons.org/licenses/by-nc/4.0/>).

recognition accuracy under variations in illumination, facial expression, and viewing angle, which makes them suitable for practical attendance applications.

However, implementing CNN-based face recognition models on embedded hardware remains challenging because convolution operations require substantial computational resources and memory [2]. Software implementations executed on central processing units (CPUs) often suffer from limited parallelism, while graphics processing units (GPUs) provide higher computational performance at the expense of increased power consumption and hardware cost [3]. Application-specific integrated circuits (ASICs) offer excellent performance and energy efficiency but lack the flexibility required for rapid design modifications and academic prototyping [4]. Among these platforms, Field-Programmable Gate Arrays (FPGAs) provide an effective balance between computational performance, power efficiency, hardware reconfigurability, and implementation cost, which makes them suitable for real-time embedded vision applications [5].

Although numerous FPGA-based face recognition systems have been reported, several limitations remain. Many reported implementations focus primarily on accelerating CNN inference using preprocessed image datasets and do not incorporate a complete embedded recognition system with live image acquisition and attendance recording [6]. In addition, high recognition performance is frequently demonstrated using high-end FPGA platforms with substantially greater hardware resources than those available on low-cost devices. These limitations reduce the practicality of deploying existing solutions in educational environments that require an affordable, self-contained attendance monitoring system.

Motivated by these challenges, this study develops a lightweight CNN-based face recognition attendance system implemented on a Cyclone IV FPGA. The proposed architecture integrates an OV7670 camera for live image acquisition, a fixed-point CNN accelerator implemented in SystemVerilog, a VGA display interface, and a UART-based attendance logging module into a complete embedded platform. The lightweight CNN architecture and fixed-point quantization strategy enable the system to operate within the hardware constraints of a Cyclone IV FPGA while maintaining high recognition accuracy and low power consumption. The effectiveness of the proposed system is evaluated through CNN performance analysis, RTL verification, and FPGA hardware implementation.

The main contributions of this work are summarized as follows:

- A lightweight CNN architecture specifically designed for deployment on a low-cost Cyclone IV FPGA platform.
- An integrated FPGA-based attendance system that combines live image acquisition, fixed-point CNN inference, VGA visualization, and UART-based attendance logging.
- A hardware implementation validated through software testing, RTL verification, and FPGA deployment, showing efficient resource utilization and low power consumption.

2. RELATED WORK

This section reviews prior studies on face recognition algorithms and FPGA-based face recognition systems. The discussion focuses on the evolution of face recognition techniques, recent FPGA implementations, and the research gaps that motivate the proposed work.

2.1 Face Recognition Algorithms

Over the past few decades, face recognition technology has undergone remarkable development, evolving from handcrafted feature extraction methods to data-driven deep learning techniques. Earlier face recognition systems mainly relied on handcrafted feature descriptors, including Principal Component Analysis (PCA) and Local Binary Patterns (LBP), were widely adopted because of their relatively low computational complexity and simple implementation [1], [7]. However, these methods are highly sensitive to variations in illumination, facial expression, head pose, and occlusion, which often limit their recognition performance in practical environments.

Recent advances in deep learning have resulted in considerable improvement in the performance of face recognition systems. CNNs learn hierarchical facial representations automatically from training data, without the need of manually designed feature descriptors. The CNN-based approaches have shown greater durability and recognition accuracy in different environmental conditions than the traditional methods [1]. Hence, many deep CNN architectures like Residual Network (ResNet) and Additive Angular Margin Loss (ArcFace) have obtained recognition accuracy higher than 99% on benchmark face recognition datasets [8], [9]. While such models perform excellently, they generally demand a considerable amount of computational resources, memory capacity and processing power, thus being inadequate for deployment on resource-constrained embedded platforms.

To overcome these limitations, recent studies have focused on lightweight CNN architectures that reduce model complexity while maintaining competitive recognition performance. Lightweight models employ compact network structures and optimized convolution operations to decrease the number of trainable parameters and computational workload [10]. These characteristics make lightweight CNNs particularly attractive for FPGA-based embedded systems, where hardware resources, memory capacity, and power consumption are limited. Thus, lightweight CNN architectures offer a feasible compromise between recognition accuracy and hardware efficiency for real-time face recognition applications.

2.2 FPGA-Based Face Recognition Systems

The increasing computational complexity of convolutional neural networks has encouraged researchers to investigate dedicated hardware platforms for accelerating face recognition systems. Among the available hardware solutions, FPGAs have attracted considerable attention because they provide parallel processing capability, low latency, hardware reconfigurability, and relatively low power consumption compared with conventional CPU- and GPU-based implementations [11]. These characteristics make FPGAs well-suited for real-time embedded vision applications where computational resources and energy efficiency are critical.

Representative FPGA-based face recognition systems reported in recent literature are summarized in Table 1. Santiwivat et al. [12] implemented a Multi-task Cascaded Convolutional Neural Network (MTCNN)-based face recognition system on a Xilinx Kria KV260 FPGA, achieving a recognition accuracy of 93.77% at 180.59 frames per second (FPS). Similarly, Zhuge et al. [13] proposed a hybrid efficient CNN architecture implemented on a Xilinx VU9P FPGA, reporting an accuracy of 99.63% with a processing speed of 42 FPS. Selvi et al. [14] developed a face recognition system based on Haar Cascade and Local Binary Pattern Histogram (LBPH) algorithms on a PYNQ-Z2 FPGA, achieving approximately 92% recognition accuracy while operating at 632.91 FPS. In another study, Albdairei et al. [15] implemented a deep CNN on a DE1-SoC FPGA platform, achieving 96.9% recognition accuracy at 0.36 FPS.

Although these studies indicate the feasibility of deploying face recognition algorithms on FPGA platforms, several limitations remain. Many existing implementations focus on accelerating CNN inference or improving recognition accuracy, without integrating a complete attendance-monitoring system. Moreover, high recognition performance is often achieved with high-end FPGAs that offer substantially greater computational resources than low-cost FPGAs. These limitations increase implementation cost and reduce the practicality of deploying FPGA-based face recognition systems in educational environments where affordability, hardware efficiency, and system integration are important.

Table 1. Comparison of Previous FPGA-Based Face Recognition Systems

Ref.	Algorithm	FPGA	Accuracy (%)	FPS	Limitation
[12]	MTCNN	Xilinx Kria KV260	93.77	180.59	High-end FPGA
[13]	Hybrid Efficient CNN	Xilinx VU9P	99.63	42	High hardware complexity
[14]	Haar Cascade + LBPH	PYNQ-Z2	92.00	632.91	Traditional algorithm
[15]	Deep CNN	DE1-SoC	96.90	0.36	Low processing speed
Proposed	Lightweight CNN	Cyclone IV EP4CE115 (DE2-115)	97.62 (RTL)	1.86	Low-cost FPGA

2.3 Research Gap

The literature review indicates that considerable progress has been made in FPGA-based face recognition systems. Nevertheless, several research gaps remain, limiting their practical deployment in real-world attendance applications.

First, many existing studies evaluate face recognition algorithms using pre-processed image datasets rather than live camera input [16]. Although these implementations demonstrate promising recognition performance, they do not fully address the challenges of real-time image acquisition,

preprocessing, and embedded system integration. Hence, their applicability to practical attendance monitoring systems remains limited.

Second, previous FPGA-based implementations focus on accelerating CNN inference or improving recognition accuracy, while relatively few studies integrate comprehensive attendance-monitoring functionality, including live image acquisition, user feedback, and attendance logging [17]. As a result, most reported systems evaluate only the recognition algorithm instead of demonstrating a fully integrated embedded attendance solution.

Furthermore, many existing FPGA implementations prioritize recognition accuracy and processing speed by employing high-performance FPGA devices, which makes their deployment less practical for cost-sensitive educational environments [18]. Although these platforms achieve excellent recognition accuracy and processing speed, their hardware cost and resource availability may limit practical deployment in educational institutions and other cost-sensitive applications.

To address the limitations identified in previous studies, this work proposes an FPGA-based real-time face recognition attendance system using a lightweight convolutional neural network implemented on a low-cost DE2-115 FPGA. The proposed system integrates live image acquisition, fixed-point CNN inference, VGA visualization, and UART-based attendance logging into a single embedded platform while maintaining high recognition accuracy, efficient hardware resource utilization, and low power consumption.

3. PROPOSED METHODOLOGY

This section describes the methodology used to develop the proposed FPGA-based face recognition attendance system. It presents the overall system architecture, dataset preparation, lightweight CNN architecture, fixed-point quantization process, FPGA hardware implementation, and verification procedure. These components are integrated to achieve accurate real-time face recognition while operating within the hardware resource constraints of a low-cost DE2-115 FPGA.

3.1 Overall System Architecture

Fig 1 illustrates the workflow of the developed embedded attendance platform. The developed platform combines image acquisition, image preprocessing, CNN inference, recognition visualization, and attendance logging into a unified embedded system implemented on a low-cost DE2-115 FPGA. The overall design emphasizes efficient hardware utilization while providing reliable real-time face recognition for attendance monitoring.

During system operation, facial images are first captured by the OV7670 camera and transferred to the image preprocessing module. The acquired images are converted to grayscale and resized to 48×48 pixels before being forwarded to the CNN accelerator. Standardizing the input image size reduces computational complexity, ensures compatibility with the trained CNN model, and minimizes memory usage during hardware implementation.

After preprocessing, the resized image is processed by the lightweight CNN accelerator implemented in System Verilog. The accelerator performs feature extraction and classification using fixed-point arithmetic, producing prediction scores for all 42 identity classes. The network consists of two convolutional layers, two max-pooling layers, one fully connected layer, and an output classification layer. To satisfy the hardware limitations of the Cyclone IV FPGA, the trained floating-point model is converted into a fixed-point representation before deployment, thereby reducing hardware resource consumption without significantly affecting recognition performance.

Once CNN inference is completed, the predicted identity is simultaneously forwarded to the VGA display controller and the UART communication interface. The VGA module provides real-time visualization of the recognition result, whereas the UART module transmits the recognized identity to a host computer for automatic attendance recording, together with the corresponding timestamp. By integrating image acquisition, hardware-accelerated CNN inference, user visualization, and attendance management into a single FPGA platform, the proposed system provides a practical solution for embedded real-time attendance monitoring.

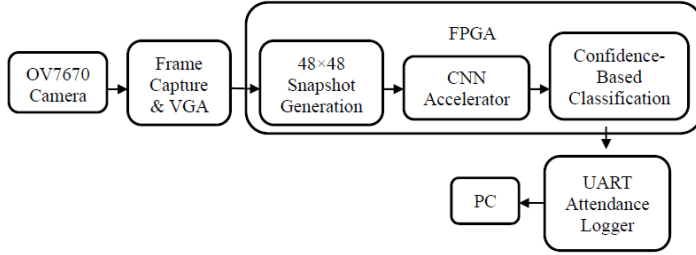


Fig. 1. Overall System Architecture of the Proposed FPGA-Based Face Recognition System.

3.2 Dataset Preparation

The dataset used in this study comprises 590 facial images across 42 identity classes. The first 40 classes were obtained from the Olivetti Research Laboratory (ORL) face dataset, while two additional classes were created to represent a registered user (ID40) and unknown faces (ID41). The inclusion of an unknown-face class enables the proposed system to reject unregistered users, making it more suitable for practical attendance monitoring applications.

The dataset was divided into training, validation, and testing subsets in a 70:15:15 ratio. To improve model generalization and reduce the risk of overfitting, data augmentation techniques were applied to the training images, increasing the total number of training samples from 590 to 1,652. The validation and testing datasets remained unchanged to ensure an unbiased evaluation of the trained model.

Before training, all facial images were converted to grayscale and resized to 48×48 pixels. Grayscale images were selected to reduce memory consumption and computational complexity while preserving sufficient facial features for recognition. The input image size was chosen as a compromise between recognition performance and hardware resource requirements, making the trained CNN more suitable for implementation on the resource-constrained Cyclone IV FPGA.

A summary of the final dataset distribution used for model development is shown in Table 2.

Table 2. Dataset Distribution for 42 Classes

Class	Original	Training	Validation	Testing
ID00-ID39	400	1120	60	60
ID40	110	308	17	16
ID41	80	224	11	13
Total	590	1652	88	89

3.3 Lightweight CNN Architecture

The proposed attendance system employs a lightweight convolutional neural network specifically designed for implementation on resource-constrained FPGA hardware. Instead of adopting a deep CNN with numerous convolutional layers, the network architecture is intentionally simplified to reduce computational complexity, memory usage, and hardware resource requirements while maintaining satisfactory face recognition performance.

As illustrated in Fig. 2, the network accepts a 48×48 grayscale facial image as input and processes it through two convolutional layers, two max-pooling layers, one fully connected layer, and a final classification layer. The convolutional layers automatically learn representative facial features, whereas the max-pooling operations reduce the spatial dimensions of intermediate feature maps, thereby decreasing computational workload and memory access during hardware execution. The extracted features are subsequently flattened and classified by the fully connected layer to generate prediction logits for all 42 identity classes.

The compact network architecture contains approximately 211,000 trainable parameters, making it substantially smaller than conventional deep CNN models designed for large-scale face recognition tasks. Reducing the number of trainable parameters not only simplifies hardware implementation but also decreases memory storage requirements and arithmetic operations on the FPGA. Consequently, the lightweight architecture provides an effective compromise between recognition accuracy, computational efficiency, and hardware resource utilization, making it suitable for real-time embedded attendance monitoring applications.

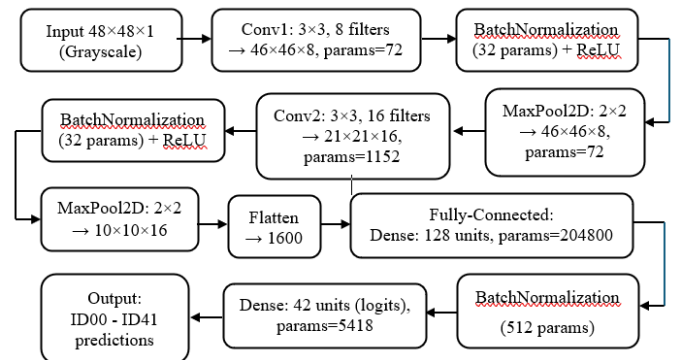


Fig. 2. Lightweight CNN Architecture for 42 Class Face Recognition.

3.4 Fixed-Point Quantization

Although floating-point arithmetic provides high numerical precision during CNN training, it is generally unsuitable for implementation on low-cost FPGA platforms because of its considerable hardware complexity and resource consumption. Therefore, the trained CNN model was converted into a fixed-point representation prior to hardware deployment to improve computational efficiency and reduce memory requirements.

SafeScale quantization was employed to generate hardware-compatible network parameters while preserving the numerical behavior of the original floating-point model. In the proposed implementation, network weights were represented as 8-bit signed integers, whereas intermediate activations were represented as 16-bit fixed-point values. Individual scaling factors were determined for each network layer to minimize quantization errors and maintain classification performance after conversion.

Table 3 outlines the adopted quantization configuration. The selected numerical precision represents a practical trade-off between arithmetic accuracy and hardware efficiency. Using 8-bit weights substantially decreases parameter storage requirements, while 16-bit activations provide sufficient numerical range for feature extraction and classification. Following quantization, all network parameters were exported and directly integrated into the SystemVerilog CNN accelerator, ensuring numerical consistency between the Python reference model and the FPGA implementation.

Table 3. Quantization Scheme

Parameter Type	Original Format	Quantized Format	Per-Layer Scaling
Weights	32-bit float	8-bit signed integer	Multiplier + shift
Biases (Accumulator-scale)	32-bit float	16-bit signed integer	Included in multiplier/shift
Activations	Floating-point	16-bit fixed-point	Dynamic scaling

3.5 FPGA Hardware Implementation

The quantized lightweight CNN was implemented on a DE2-115 FPGA using System Verilog. The FPGA implementation integrates image acquisition, CNN inference, result visualization, and attendance logging into a single embedded platform. Fig 3 illustrates the overall hardware architecture of the proposed implementation.

The hardware implementation consists of four major functional modules: the image acquisition module, the CNN accelerator, the VGA display controller, and the UART communication interface. The image acquisition module captures live facial images from the OV7670 camera and forwards the pre-processed grayscale image to the CNN accelerator. The accelerator performs feature extraction and classification using fixed-point arithmetic before transmitting

the predicted identity to both the VGA controller and the UART module. The VGA interface displays the captured image together with the recognition result in real time, while the UART interface sends the recognition result to a host computer for automatic attendance recording.

To improve hardware efficiency, the CNN accelerator was designed using a resource-sharing architecture. Instead of implementing all convolution filters simultaneously, two physical convolution engines are reused sequentially to generate multiple feature maps. This approach significantly reduces the number of required DSP blocks and logic resources while maintaining the functionality of the trained CNN model. Furthermore, fixed-point arithmetic eliminates the need for floating-point computation, reducing memory usage and improving hardware efficiency on the DE2-115 FPGA.

The modular hardware architecture simplifies system integration and enables independent verification of each functional block before complete FPGA deployment. This design approach improves implementation flexibility while ensuring reliable real-time operation for the proposed attendance monitoring system.

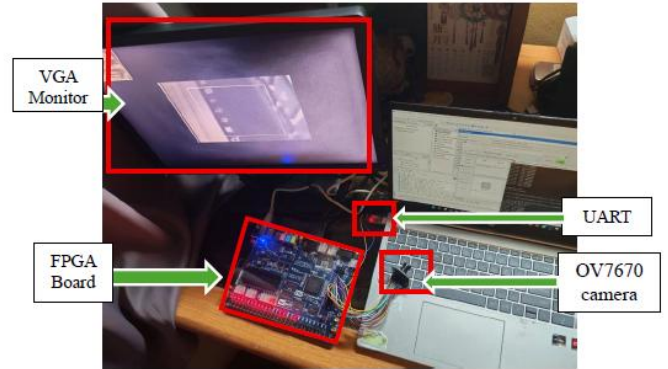


Fig. 3. Overall FPGA hardware architecture of the proposed attendance system.

3.5.1 CNN Accelerator

The CNN accelerator serves as the core computational module of the proposed FPGA-based face recognition system. Its primary function is to execute CNN inference using the quantized lightweight network and generate the final identity prediction. To address the hardware limitations of the Cyclone IV EP4CE115 FPGA, the accelerator was developed in System Verilog using fixed-point arithmetic and a resource-sharing architecture that minimizes hardware resource utilization without significantly compromising recognition performance.

Fig 4 illustrates the hardware organization of the CNN accelerator, while the functional description of each processing module is summarized in Table 4. Instead of executing all convolution operations simultaneously, the proposed design reuses two physical convolution engines to compute multiple convolution filters sequentially. This resource-sharing strategy substantially reduces the number of required DSP blocks, logic elements, and on-chip memory resources, enabling the lightweight CNN to be efficiently deployed on a low-cost FPGA platform.

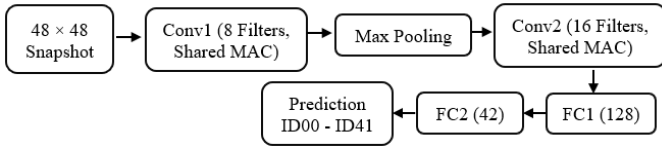


Fig. 4. FPGA CNN Accelerator Architecture with Resource Sharing.

Table 4. CNN Accelerator

Module	Function
Convolution	Feature extraction
Max-Pooling	Feature map down-sampling
Flatten	Feature vector generation
Fully Connected	Classification
Output	Identity prediction

The inference process begins with the quantized input image entering the convolution stage, where representative facial features are extracted using fixed-point multiply-accumulate (MAC) operations. The generated feature maps are subsequently processed by the max-pooling modules to reduce their spatial dimensions and computational complexity. After feature extraction, the feature maps are flattened into a one-dimensional feature vector and passed to the fully connected layer to compute the output logits for the 42 registered identity classes. The class associated with the highest output logit is selected as the predicted identity before being forwarded to the subsequent decision logic.

The CNN parameters generated through the SafeScale quantization process are stored in the FPGA on-chip memory and accessed during inference. Since all arithmetic operations are performed using fixed-point representations, the accelerator avoids the overhead associated with floating-point computation while maintaining numerical consistency with the Python fixed-point reference model. The modular hardware architecture also facilitates independent functional verification of each processing stage before complete system integration, thereby improving implementation reliability and simplifying hardware debugging.

3.5.2 Image Acquisition and Display Interface

The image acquisition module captures real-time facial images using an OV7670 camera connected directly to the Cyclone IV FPGA. The camera continuously transmits image data to the FPGA, where the captured frames are buffered and processed before CNN inference. To reduce computational complexity and memory consumption, each captured image is converted to grayscale and resized to 48×48 pixels, matching the input dimensions required by the lightweight CNN model.

After image preprocessing, the processed image is forwarded to the CNN accelerator for feature extraction and face classification. The preprocessing stage ensures that all input images have a consistent format, thereby improving recognition reliability and simplifying the subsequent hardware implementation. The reduced image resolution also minimizes memory bandwidth requirements while preserving sufficient facial features for accurate recognition.

The recognition results are displayed through a VGA interface to provide immediate visual feedback during system operation. The VGA controller generates the required synchronization signals and continuously outputs the processed image together with the predicted identity on an external monitor. This visual feedback enables users to verify the recognition result in real time and facilitates hardware debugging during FPGA development. The integration of image acquisition, preprocessing, CNN inference, and VGA visualization enables the implemented system to operate as a complete real-time embedded face recognition platform. The real-time visualization provided by the VGA interface allows the recognition outcome to be verified immediately after image capture, improving system usability and supporting hardware validation during FPGA implementation.

3.5.3 Snapshot-Based Recognition

In contrast to conventional real-time face recognition systems that continuously process incoming video frames, the FPGA adopts a snapshot-based recognition mechanism as shown in Fig 5 to improve hardware efficiency and avoid redundant attendance records. The recognition process is initiated only when the user presses the snapshot button, allowing the FPGA to capture a single facial image for CNN inference. This event-driven approach significantly reduces unnecessary computations because the CNN accelerator is activated only when face recognition is required.

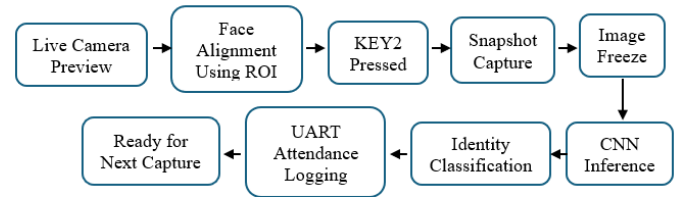


Fig. 5. Snapshot-based Recognition Sequence Implemented.

After a snapshot is captured, the input image is pre-processed and forwarded to the lightweight CNN accelerator for feature extraction and classification. The network generates an output logit for each of the 42 identity classes, where the class with the highest logit is selected as the predicted identity. To improve recognition reliability, a confidence-gap criterion is employed to distinguish registered users from unknown faces. The confidence gap is calculated as the difference between the highest and second-highest output logits, as expressed in (1).

$$CG = L_{max} - L_{second} \quad (1)$$

where CG denotes the confidence gap, L_{max} is the highest output logit, and L_{second} is the second-highest output logit. A larger confidence gap indicates greater confidence in the predicted identity, whereas a smaller confidence gap suggests that multiple classes have similar prediction scores, increasing the possibility of incorrect classification.

The calculated confidence gap is compared with a predefined rejection threshold to determine whether the recognition result should be accepted. If the confidence gap exceeds the predefined threshold, the predicted identity is accepted as a registered user and forwarded to the attendance

logging module. Otherwise, the input image is classified as an unknown face, and no attendance record is generated. Based on experimental evaluation, the rejection threshold was set to 300, providing an effective balance between false acceptance and false rejection while maintaining stable recognition performance.

The proposed snapshot-based recognition mechanism offers several advantages over continuous recognition systems. Since CNN inference is executed only when a snapshot is captured, unnecessary computations are eliminated, reducing FPGA workload and power consumption. Furthermore, the event-driven recognition process prevents repeated attendance records for the same individual and improves the overall reliability of the attendance monitoring system.

3.5.4 UART-Based Attendance Logging

The attendance logging module provides communication between the FPGA-based face recognition system and a host computer through a Universal Asynchronous Receiver-Transmitter (UART) interface. After the CNN accelerator completes the classification process, the predicted identity is transmitted via the UART module to a Python-based application running on the host computer for attendance recording.

The UART module operates as the communication bridge between the hardware and software components of the system. Each recognized identity is transmitted as a serial data packet, where the Python application receives, decodes, and associates the identity with the corresponding attendance record. If the recognized face belongs to a registered user, the attendance information together with the corresponding timestamp is automatically recorded in the attendance database. As a result, if an unknown face is detected, the system rejects the recognition result and no attendance record is generated.

The UART-based communication mechanism enables reliable data transfer while maintaining a simple hardware implementation. Separating the CNN inference process from attendance management also improves system flexibility, allowing the FPGA to focus on real-time face recognition while the host computer performs data storage and user management. This architecture simplifies future system expansion, such as integrating database servers or cloud-based attendance management platforms.

3.5.5 Hardware Verification

A comprehensive verification procedure was conducted to validate the correctness of the proposed FPGA implementation prior to hardware deployment. The verification process consisted of three sequential stages: software evaluation, register-transfer level (RTL) simulation, and FPGA hardware validation. This multi-stage approach ensured that the implemented hardware accurately preserved the inference behaviour of the trained lightweight CNN model throughout the deployment process.

The first verification stage employed the Python fixed-point reference model to establish the expected classification results after SafeScale quantization. The quantized network parameters generated by the software model were then exported

and integrated directly into the System Verilog implementation. During RTL simulation, identical input images were applied to both the Python model and the hardware design to evaluate numerical consistency between software and hardware inference.

The RTL verification focused on validating the functionality of the convolution, max-pooling, flattening, and fully connected modules, as well as the final classification logic. For each test image, the predicted identity, output logits, and confidence-gap calculation generated by the hardware implementation were compared with the corresponding outputs from the Python fixed-point model. This verification procedure confirmed that the proposed CNN accelerator correctly preserved the computational behaviour of the software reference model without introducing significant numerical deviations.

Following successful RTL verification, the complete system was synthesized and deployed on the DE2-115 FPGA development board for hardware validation. Experimental evaluation using the testing dataset demonstrated that the implemented accelerator achieved an RTL verification accuracy of 97.62%, while maintaining complete prediction consistency with the Python fixed-point reference model. These results verify that the proposed hardware implementation performs reliable fixed-point CNN inference and provides a dependable foundation for real-time FPGA-based face recognition attendance monitoring.

4. EXPERIMENTAL RESULTS AND DISCUSSION

This section presents the experimental results obtained from the proposed face recognition attendance system. The CNN model is first evaluated using the validation and testing datasets, followed by RTL verification and FPGA hardware implementation. Finally, the performance of the proposed system is compared with previous studies, and the obtained results are discussed.

4.1 Experimental Setup

The proposed face recognition attendance system was evaluated through software validation, RTL verification, and FPGA hardware implementation. The CNN model was first trained and validated in Python with TensorFlow, while the hardware modules were designed in SystemVerilog and synthesized with Intel Quartus Prime for deployment on the Cyclone IV EP4CE115 FPGA development board.

The experimental platform consisted of an OV7670 camera for real-time image acquisition, a DE2-115 FPGA development board for CNN inference, a VGA monitor for visualizing recognition results, and a host computer for attendance management via UART communication. During system operation, facial images were captured by the camera, pre-processed into 48×48 grayscale images, and forwarded to the FPGA-based CNN accelerator for real-time face recognition.

The trained CNN model was developed using a dataset comprising 42 identity classes, including 40 registered identities from the ORL face dataset, one registered user class, and one unknown-face class. The dataset was divided into training, validation, and test subsets in a 70:15:15 ratio, and data

augmentation was applied to increase the number of training samples and improve model generalization.

To verify the correctness of the hardware implementation, the FPGA outputs were compared with the Python fixed-point reference model using the same testing dataset. The system performance was evaluated based on recognition accuracy, RTL verification accuracy, FPGA resource utilization, processing speed, and power consumption. These performance metrics were used to assess the effectiveness of the proposed lightweight CNN architecture and its suitability for real-time FPGA-based attendance monitoring. Table 5 summarized the experimental setup settings for this work.

Table 5. Experimental Setup

Item	Specification
CNN Framework	TensorFlow
Programming Language	Python
HDL	System Verilog
FPGA Device	Cyclone IV EP4CE115 (DE2-115)
Camera	OV7670
Display	VGA Monitor
Communication	UART
Input Image Size	48 × 48 Grayscale
Classes	42
Dataset	ORL + Registered User + Unknown Face

4.2 CNN Performance

The performance of the proposed lightweight CNN was first evaluated through software training before FPGA implementation. The CNN model was trained using the prepared face image dataset with data augmentation to improve generalization and reduce overfitting. Model performance was evaluated on both validation and test datasets to assess recognition capability before hardware deployment.

Table 6 demonstrates the classification performance of the trained CNN model. The proposed network achieved a validation accuracy of 94.32% and a test accuracy of 98.88%, indicating that the lightweight architecture learned discriminative facial features despite its compact structure. The high testing accuracy demonstrates that the proposed CNN provides reliable face recognition performance while maintaining a relatively small number of trainable parameters.

Table 6. CNN Performance

Performance Metric	Result
Validation Accuracy	94.32%
Testing Accuracy	98.88%
Number of Classes	42
Training Images	1652

To further evaluate the learning behaviour of the proposed CNN, the training and validation accuracy and loss curves are presented in Fig 6.

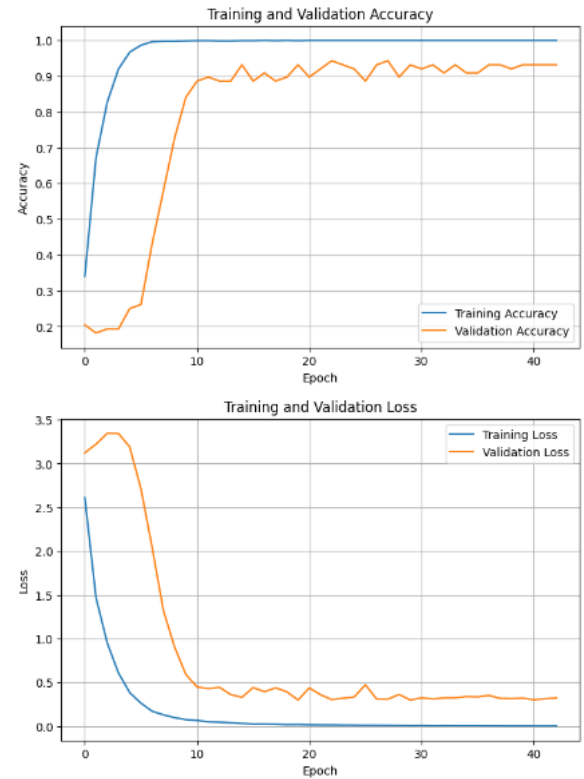


Fig. 6. Training and validation accuracy and loss curves of the proposed lightweight CNN.

As shown in Fig 6, both the training and validation accuracy increased steadily throughout the training process before gradually converging. Similarly, the training and validation losses decreased consistently, with no significant fluctuations. The small gap between the training and validation curves indicates that the proposed lightweight CNN achieved good generalization capability and did not exhibit severe overfitting. These results demonstrate that the selected network architecture effectively learned discriminative facial features while maintaining stable convergence during training. The trained model achieved a validation accuracy of 94.32% and a testing accuracy of 98.88%, demonstrating that the lightweight CNN provides reliable recognition performance despite its compact architecture. These software evaluation results served as the baseline for the subsequent fixed-point quantization and FPGA implementation.

The obtained results indicate that reducing the network complexity does not significantly compromise recognition accuracy. By employing only two convolutional layers and a lightweight network architecture, the proposed model successfully balances classification performance and computational complexity. This characteristic is particularly important for FPGA implementation, where hardware resources and on-chip memory are limited.

The trained CNN model was subsequently quantized using the SafeScale fixed-point quantization approach before being deployed on the Cyclone IV FPGA. The software evaluation results provide the baseline performance for subsequent RTL verification and hardware implementation discussed in the following sections.

4.3 RTL Verification

Before deploying the proposed CNN accelerator on the Cyclone IV FPGA, register-transfer level (RTL) verification was performed to validate the functional correctness of the hardware implementation. The objective of this verification stage was to ensure that the System Verilog implementation preserved the numerical behaviour of the quantized CNN model and produced prediction results consistent with the Python fixed-point reference model.

The verification process was conducted using the same test dataset for both the Python fixed-point model and the RTL simulation. For each input image, the predicted identity generated by the hardware implementation was compared with the corresponding software output. The comparison confirmed that the convolution, max-pooling, and fully connected operations were correctly implemented using fixed-point arithmetic without introducing significant numerical deviations.

Table 7 indicates the RTL verification results. The proposed hardware implementation achieved an RTL verification accuracy of 97.62% while maintaining complete consistency in predictions with the Python fixed-point reference model. The small reduction in classification accuracy compared with the floating-point software model is primarily due to the loss of numerical precision introduced during fixed-point quantization. Nevertheless, the obtained accuracy indicates that the proposed quantization strategy successfully preserves the recognition capability of the original CNN model after hardware implementation.

Table 7. RTL Verification Results

Metric	Result
Verification Accuracy	97.62%
Reference Model	Python Fixed-Point
Prediction Consistency	100%
Verification Status	Passed

Fig 7 shows the RTL verification output, including the predicted identity, confidence gap, unknown decision, and comparison with the Python fixed-point reference model. The hardware correctly classified the input image as ID40, with a confidence gap of 925, which exceeded the predefined rejection threshold of 300.

```

# =====
# CNN DONE
# =====
# Predicted ID      = 40
# Max logit        = 1207
# Second logit     = 282
# Confidence gap    = 925
# Unknown flag     = 0
# Final unknown    = 0
# LEDR            = 111010001101000011
# =====
# LOGIT COMPARE SUMMARY
# =====
# Max difference    = 10
# Mismatch > 5     = 8
# & PASS: RTL predicted ID40, same as Python fixed-point.
# =====

```

Fig. 7. Console Log.

The RTL verification results demonstrate that the proposed CNN accelerator accurately performs fixed-point inference and provides a reliable foundation for FPGA deployment. Therefore, the verified hardware design was subsequently

synthesized and implemented on the Cyclone IV FPGA for real-time face recognition evaluation.

4.4 FPGA Hardware Performance

The verified CNN accelerator was synthesized and implemented on the Cyclone IV EP4CE115 FPGA to evaluate its hardware performance. The implementation results were assessed in terms of hardware resource utilization, processing speed, and power consumption. These metrics were selected to determine whether the proposed lightweight CNN architecture is suitable for deployment on a resource-constrained FPGA.

Table 8 summarizes the hardware resource utilization of the proposed architecture. The implementation utilized 36% of the available logic elements, 1% of the registers, and 7% of the embedded DSP blocks, demonstrating that the lightweight CNN architecture requires relatively low computational resources. The on-chip M9K memory utilization reached 94%, primarily due to storing quantized CNN parameters and intermediate feature maps during inference. Despite the relatively high memory utilization, the implementation was successfully completed without exceeding the available hardware resources of the Cyclone IV FPGA.

Table 8. FPGA Resource Utilization Summary

Resource	Usage	Utilisation
Logic Elements	41,530 / 114,480	36%
Registers	28,585 / 117,053	24%
I/O Pins	84 / 529	16%
M9K Memory Blocks	404 / 432	94%
Block Memory Bits	3,213,184 / 3,981,312	81%
Embedded 9-bit Multipliers	37 / 532	7%

The FPGA implementation achieved a processing speed of 1.86 frames per second (FPS) while consuming approximately 171.45 mW of power, as summarized in Table 9. Although the proposed system operates at a lower frame rate than some high-end FPGA implementations, the achieved performance is sufficient for snapshot-based attendance monitoring, where continuous video processing is not required. Furthermore, the low power consumption demonstrates the effectiveness of the lightweight CNN architecture, fixed-point quantization, and resource-sharing hardware design in reducing computational.

Table 9. FPGA Performance

Performance Metric	Result
FPGA Device	Cyclone IV EP4CE115 (DE2-115)
Processing Speed	1.86 FPS
Power Consumption	171.45 mW

Overall, the hardware evaluation confirms that this work achieves a practical balance between recognition accuracy, hardware resource utilization, processing speed, and energy efficiency. These results demonstrate that the proposed architecture is suitable for real-time FPGA-based attendance monitoring using a low-cost Cyclone IV FPGA platform. These results indicate that the proposed lightweight CNN architecture is suitable for practical real-time attendance monitoring on resource-constrained FPGA platforms without requiring high-end hardware accelerators.

4.4.1 Real-Time Recognition Demonstration

The real-time operation of the proposed FPGA-based attendance system was validated through hardware experiments using the DE2-115 FPGA development board. During system operation, facial images were captured by the OV7670 camera, processed by the lightweight CNN accelerator, and classified in real time. The recognized identity was displayed through the VGA interface while the corresponding attendance information was simultaneously transmitted to the host computer through the UART communication module.

Fig 8 presents an example of successful face recognition performed by the presented system. The recognized identity was correctly displayed on the VGA monitor, while the attendance record was automatically stored by the host computer. The obtained results demonstrate the successful integration of image acquisition, CNN inference, recognition visualization, and attendance logging within a single embedded platform.

To improve system reliability, the proposed snapshot-based recognition mechanism performs CNN inference only after a snapshot is captured. In addition, the confidence-gap decision logic effectively rejects unknown users whose confidence values do not satisfy the predefined threshold. Consequently, duplicate attendance records and false recognition results can be significantly reduced during practical operation. The hardware demonstration confirms that the proposed implementation operates reliably under real-time conditions and successfully performs automatic attendance monitoring using a low-cost Cyclone IV FPGA platform.

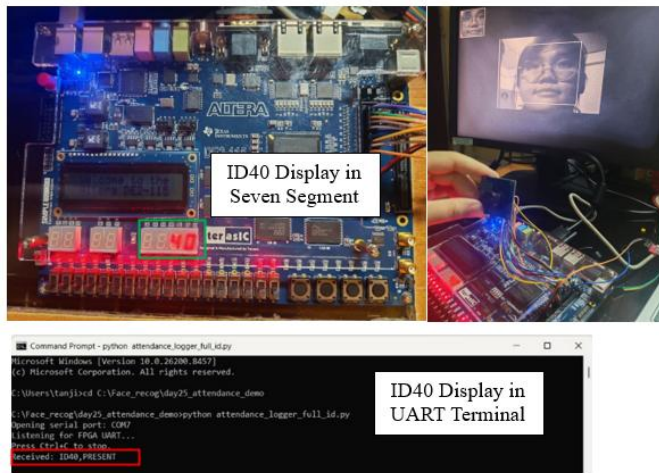


Fig. 8. Real-time demonstration of the proposed FPGA-based attendance monitoring system.

4.5 Comparison with Previous Studies

Table 10 compares the proposed FPGA-based face recognition attendance system with several representative FPGA-based recognition systems reported in recent studies. Since the reported systems were developed for different applications, datasets, CNN architectures, and FPGA platforms, direct comparison using a single performance metric may not fully reflect the overall system capability. Therefore, multiple evaluation criteria, including recognition accuracy, processing speed, power consumption, hardware platform, and application

scope, are considered to provide a more comprehensive performance comparison.

The MTCNN-based implementation reported in [12] achieved a substantially higher processing speed by utilizing a high-performance Xilinx Kria KV260 SoC FPGA. However, the implemented system achieved a higher software recognition accuracy (98.88% compared with 93.77%) while consuming significantly less power (0.171 W compared with 5.41 W). These results indicate that the lightweight CNN architecture provides an efficient solution for resource-constrained FPGA platforms without compromising recognition performance.

Compared with the deep CNN implementation presented in [15], the proposed system achieved both higher recognition accuracy and a higher processing speed while requiring considerably lower power consumption. Although both systems perform face recognition for attendance applications, the proposed implementation demonstrates that a lightweight CNN combined with fixed-point quantization and resource-sharing hardware architecture can effectively reduce computational overhead while maintaining reliable recognition accuracy.

The lightweight CNN accelerator reported in [10] achieved an accuracy of 98.56% with an exceptionally high processing speed of 3065 FPS. However, the reported system was developed for handwritten digit recognition using only 10 output classes and 28×28 grayscale images, which represents a substantially less complex classification task than the proposed face recognition system. In contrast, the proposed architecture performs face recognition over 42 identity classes using 48×48 facial images while incorporating additional functionalities, including unknown-face rejection, snapshot-based recognition, VGA visualization, and UART-based attendance logging. Consequently, the proposed implementation addresses a more practical and comprehensive embedded attendance monitoring application.

Table 10. Comparison of the Proposed System with Existing FPGA-Based FPGA Recognition Systems

Author	[12]	[15]	[10]	Proposed
Year	2024	2022	2024	2026
Algorithm	MTCNN	Deep CNN	Lightweight CNN	Lightweight CNN
FPGA Device	Xilinx Kria KV260 SoC FPGA	DE1-SoC FPGA	PYNQ-Z2 FPGA	Cyclone IV EP4CE115 (DE2-115)
Accuracy (%)	93.77	96.90	98.56	98.88 (software) 97.62 (RTL)
Processing Speed (FPS)	180.59	0.36	3065	1.86
Operating Frequency (MHz)	–	–	100	5
Power Consumption (W)	5.41	10.70	1.947	0.171
Real-Time Camera	Yes	Yes	Yes	Yes

Overall, the comparison demonstrates that the proposed lightweight CNN architecture provides an effective balance between recognition accuracy, hardware efficiency, and implementation cost. Although the achieved processing speed (1.86 FPS) is lower than that of several specialized FPGA accelerators, the lightweight CNN accelerator employs a snapshot-based recognition mechanism rather than continuous video processing. Therefore, high frame rates are not required for the intended attendance monitoring application. Furthermore, the combination of fixed-point quantization, resource-sharing CNN acceleration, and low-power FPGA implementation enables the implemented system to operate efficiently on a low-cost Cyclone IV FPGA while maintaining high recognition accuracy and complete attendance monitoring functionality.

5. CONCLUSION

In conclusion, this study investigated the implementation of a lightweight convolutional neural network (CNN) for real-time face recognition attendance monitoring on a low-cost FPGA platform. A complete embedded attendance system was successfully developed by integrating an OV7670 camera, image preprocessing, a fixed-point CNN accelerator, VGA-based recognition visualization, and UART-based attendance logging within a Cyclone IV EP4CE115 FPGA. The proposed architecture demonstrates that reliable face recognition can be achieved on resource-constrained hardware through the combination of lightweight network design, fixed-point quantization, and efficient hardware implementation.

The developed CNN model achieved a validation accuracy of 94.32% and a testing accuracy of 98.88% during software evaluation. Following SafeScale fixed-point quantization, the hardware implementation was verified via RTL simulation, achieving 97.62% verification accuracy while maintaining prediction consistency with the Python fixed-point reference model. The implemented system operated at 1.86 frames per second with an estimated power consumption of 171.45 mW, indicating that the proposed design successfully balances recognition performance, hardware resource utilization, and energy efficiency on a low-cost FPGA platform.

Compared with many existing FPGA-based face recognition implementations, this work goes beyond CNN acceleration by providing a complete embedded attendance-monitoring solution. The integration of live image acquisition, snapshot-based recognition, confidence-gap-based unknown-face rejection, VGA visualization, and UART communication enables the proposed system to perform automatic attendance recording without requiring continuous video processing. These features enhance the practicality of the developed system for educational attendance monitoring while maintaining efficient use of FPGA resources.

As a result, the experimental results confirm that the proposed lightweight CNN architecture is suitable for deployment on resource-constrained FPGAs, providing an effective compromise among computational complexity, recognition accuracy, hardware efficiency, and implementation cost. The developed system demonstrates the feasibility of implementing an embedded, real-time face-recognition

attendance platform using a low-cost Cyclone IV FPGA, without relying on high-end hardware accelerators.

Future research may focus on optimizing the CNN accelerator to improve processing throughput while maintaining low power consumption and efficient hardware utilization. The proposed system can also be extended to support a larger number of registered users, higher-resolution image inputs, and more advanced lightweight CNN architectures. Furthermore, integrating the attendance logging module with a centralized database or cloud-based management platform would enhance the scalability and practicality of the proposed system for real-world deployment.

REFERENCES

- [1] 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: <https://doi.org/10.1109/ACCESS.2024.3424933>.
- [2] T. Sutabri, Pamungkur, A. Kurniawan, and R. E. Saragih, "Automatic attendance system for university student using face recognition based on deep learning," *Int. J. Mach. Learn. Comput.*, vol. 9, no. 5, pp. 668–674, Sep. 2019, doi: <https://doi.org/10.18178/ijmlc.2019.9.5.856>.
- [3] F. Pacini, T. Pacini, G. Lai, A. M. Zocco, and L. Fanucci, "Design and Evaluation of CPU-, GPU-, and FPGA-Based Deployment of a CNN for Motor Imagery Classification in Brain-Computer Interfaces," *Electronics (Switzerland)*, vol. 13, no. 9, May 2024, doi: <https://doi.org/10.3390/electronics13091646>.
- [4] Y. Hu, Y. Liu, and Z. Liu, "A Survey on Convolutional Neural Network Accelerators: GPU, FPGA and ASIC," in *2022 IEEE 14th International Conference on Computer Research and Development, ICCRD 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 100–107, doi: <https://doi.org/10.1109/ICCRD54409.2022.9730377>.
- [5] S. I. Venieris and C.-S. Bouganis, "fpgaConvNet: A Framework for Mapping Convolutional Neural Networks on FPGAs," London, Aug. 2016, doi: <https://doi.org/10.1109/FCCM.2016.22>.
- [6] S. Zhang, "Real Time Image Processing on FPGAs," Liverpool, Jan. 2018.
- [7] Riddhi Patel and S. B. Yagnik, "A Literature Survey on Face Recognition Techniques," *International Journal of Computer Trends and Technology*, vol. 5, no. 4, 2013. Accessed: Jan. 23, 2026. [Online]. Available: <http://www.ijettjournal.org>
- [8] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, Jun. 2019, pp. 4685–4694, doi: <https://doi.org/10.1109/CVPR.2019.00482>.
- [9] D. S. Trigueros, L. Meng, and M. Hartnett, "Face Recognition: From Traditional to Deep Learning Methods," Oct. 2018. Accessed: Jan. 25, 2026, doi: <https://doi.org/10.48550/arXiv.1811.00116>.
- [10] J. Ju and C. Liu, "FPGA-Based Lightweight CNN Acceleration System for Real-Time Recognition," in *2024 9th International Conference on Integrated Circuits and Microsystems, ICICM 2024*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 581–585, doi: <https://doi.org/10.1109/ICICM63644.2024.10814533>.
- [11] PCBONLINE Team, "GPU vs FPGA vs ASIC vs CPU: Which is Used for AI Electronics?" Accessed: Dec. 25, 2025. [Online]. Available: <https://www.pcbonline.com/blog/gpu-vs-fpga-vs-asic-vs-cpu.html>
- [12] S. Santiwivat, "Multiple face detection and recognition for system-on-chip FPGAs," 2024. Accessed: Feb. 01, 2026, doi: <https://doi.org/10.58837/chula.the.2024.235>.
- [13] C. Zhuge, X. Liu, X. Zhang, S. Gummedi, J. Xiong, and D. Chen, "Face Recognition with Hybrid Efficient Convolution Algorithms on FPGAs," Mar. 2018, Accessed: Feb. 02, 2026, doi: <https://doi.org/10.48550/arXiv.1803.09004>.

- [14] S. S. Selvi, D. Bharanidharan, A. Qadir, and K. R. Pavan, "FPGA Implementation of a Face Recognition System," in *Proceedings of CONECCT 2021: 7th IEEE International Conference on Electronics, Computing and Communication Technologies*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 3–5. doi: <https://doi.org/10.1109/CONECCT52877.2021.9622348>.
- [15] A. J. A. Albdairi et al., "Face Recognition Based on Deep Learning and FPGA for Ethnicity Identification," *Applied Sciences (Switzerland)*, vol. 12, no. 5, Mar. 2022, doi: <https://doi.org/10.3390/app12052605>.
- [16] A. J. Shepley, "DEEP LEARNING FOR FACE RECOGNITION: A CRITICAL ANALYSIS," Casuarina, Darwin, Jul. 2019.
- [17] Z. Qi, W. Chen, R. A. Naqvi, and K. Siddique, "Designing Deep Learning Hardware Accelerator and Efficiency Evaluation," *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: <https://doi.org/10.1155/2022/1291103>.
- [18] X. Sui et al., "A Hardware-Friendly Low-Bit Power-of-Two Quantization Method for CNNs and Its FPGA Implementation," *Sensors*, vol. 22, no. 17, Sep. 2022, doi: <https://doi.org/10.3390/s22176618>.