OPEN ACCESS JOURNALS
Indiana Journal of Multidisciplinary Research
Abbreviated Tittle- Ind J Multi Res
ISSN *(Online)*- 2583-3820
Journal Homepage Link- https://indianapublications.com/Journals/IJMR
DOI: https://doi.org/10.5281/zenodo.21610211

INDIANA PUBLICATIONS
PRODUCTIVE AND QUALITY RESEARCH

Review Article

Volume-06|Issue-04|2026

Embedded Systems for Real-Time Facial Recognition: A Survey of Hardware-Accelerated Architectures

Bhargav M[1], Renuka S U[2], K Abhishek[3], Tanushree M M[4], Dr. Hareesh Kumar[5]

[1,2,3,4]Department of Electronics and Communication Engineering, RV Institute of Technology and Management, Bangalore, India,
[5]Associate Professor, Department of Electronics and Communication Engineering, SJC Institute of technology, Chikballapur, Karnataka, India,

Article History
Received: 05.05.2026
Accepted: 22.06.2026
Published: 25.07.2026

Citation
Bhargav, M., Renuka, S. U., Abhishek, K., Tanushree, M. M., Kumar, H. (2026) Embedded Systems for Real-Time Facial Recognition: A Survey of Hardware-Accelerated Architectures. *Indiana Journal of Multidisciplinary Research, 6*(4), 375-382.

Abstract: Real-time facial recognition is widely applied in surveillance, biometrics, and authentication. The demand for fast, low-power, and portable systems has led to the development of specialized hardware for high-speed inference. While traditional processors often face latency limitations, GPUs, FPGAs, ASICs, and multi-core CPUs provide effective acceleration. Unlike other surveys that primarily focus on algorithms, this article reviews several studies on hardware design and facial recognition methods. It also compares neural and non-neural approaches in terms of accuracy, speed, and suitability for embedded applications, aiming to guide the development of future low-latency AI systems.

Keywords: Facial Recognition, FPGA, Hardware Acceleration, Real-Time Processing, Non-Neural Network Algorithms, CNN, Embedded Processing, Parallel Processing, Low-Power Systems, Hardware Design

INTRODUCTION

Facial recognition [1] has become central to smartphones, security systems, personalization, and surveillance. With the increasing adoption of embedded devices such as smartphones, wearables, IoT devices, and edge systems [2], there is growing demand for fast, accurate, and energy-efficient solutions [3]. Convolutional Neural Networks (CNNs) enable high recognition accuracy but place significant computational demands on low-power devices [4]. FPGAs address this challenge effectively [5]. They offer programmability, parallel processing, and low power consumption, enabling customized CNN implementations [6]. This reduces latency, increases processing speed, and improves energy efficiency while supporting rapid prototyping and research into optimized hardware architectures [7], [8]. Techniques such as fixed-point arithmetic, model compression, and pipelined processing allow FPGAs to process high-resolution images in real time [9]. Their inherent parallelism [10] further accelerates inference. This paper reviews several FPGA-based facial recognition accelerators, highlighting their strengths, recent advances, and future potential.

FACE RECOGNITION SYSTEM

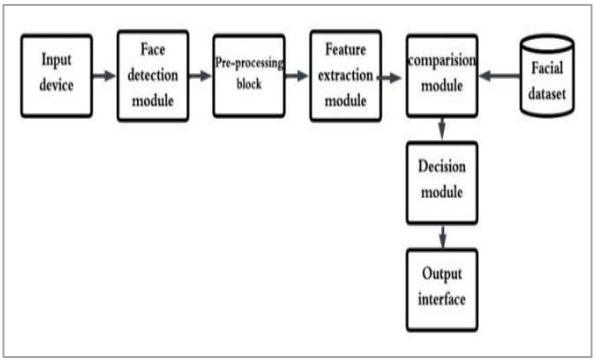

Fig. 1: Face Recognition System

Face recognition follows a pipeline consisting of detection, preprocessing, feature extraction, and recognition, as shown in Fig. 1. Each stage plays an important role in achieving high accuracy and real-time performance. The process begins with input images, which are resized or normalized before processing. During the detection stage [11], faces are located and enclosed within bounding boxes. Preprocessing [12] then reduces noise and compensates for variations in lighting, pose, and image size using techniques such as scaling and alignment. Feature extraction [13] converts facial images into compact feature representations, often using models such as MobileNet to capture distinctive facial characteristics. Finally, the recognition stage [14] compares these features with a database and determines identity based on similarity scores. Hardware platforms

***Corresponding Author:** Dr. Hareesh Kumar Email: drhareeshkumar.ece@sjcit.ac.in

such as FPGAs accelerate this process through low-latency parallel execution. Together, these stages transform raw facial images into reliable identity recognition, forming the foundation of modern biometric systems.

HARDWARE PLATFORMS FOR FACIAL RECOGNITION

Fixed-Architecture Devices – Multi-Core Processing Units and Graphics Processing Units

CPUs remain essential in embedded systems, handling control flow and preprocessing tasks in face recognition. However, their sequential architecture limits deep learning performance, even when multiple cores are used [15]. GPUs, designed for highly parallel computation, efficiently execute convolution-intensive operations and perform exceptionally well during model training and server-based recognition [16]. However, their high power consumption and cooling requirements make them less suitable for edge devices.

Performance comparisons clearly demonstrate this difference. The ARM Mali-T168 GPU was reported to be 15× slower than the Stratix V FPGA [17], while the Arria 10 FPGA achieved 763.5 FPS with superior energy efficiency [18]. Similarly, the Zynq AXZU5EV outperformed a conventional CPU by 22.3× [19]. These results highlight why FPGAs provide an excellent balance between performance, energy efficiency, and flexibility for real-time edge-based facial recognition, whereas CPUs and GPUs remain better suited for general-purpose computing and algorithm development.

Field-Programmable Gate Arrays (FPGAs)

FPGAs utilize configurable logic blocks (CLBs), DSP slices, and programmable routing resources to execute tasks in parallel, providing high performance with lower power consumption than CPUs and GPUs [20]. They are well suited for implementing deep learning models such as VGG, ResNet, YOLO, and MobileNet for face recognition [19] and object detection [21].

High-end FPGA platforms such as the Arria 10 (763.5 FPS) [18] and Stratix V (15× faster than GPUs) [17] demonstrate outstanding performance, although they require greater power and more complex hardware design. For practical deployments, mid-range platforms such as the Zynq-7000, Zynq UltraScale+, and Ultra96v2 (47.97 images/s at 2.967 W) [22], [23], [15] offer an effective balance between cost and performance. Lower-cost platforms including the PYNQ-Z1/Z2, ZC706, and Spartan-3 support lightweight AI applications [24]–[27]. Even older devices such as the Cyclone V [28], [29] and DE2-115 (60 FPS using only 8% memory) [30] remain valuable for research and educational applications. More recently, ReRAM/SRAM-based FPGA architectures have further improved efficiency, achieving 53.55 TOPS/W for neuromorphic AI applications [31].

FACE RECOGNITION ALGORITHMS

Non-Neural Network-Based Algorithms

Principal Component Analysis (PCA) reduces image dimensionality by projecting facial images into an eigenface space, where recognition is performed using distance measures such as Euclidean distance. On FPGAs, parallel matrix operations, pipelining, and fixed-point arithmetic enable real-time implementation, although conventional PCA remains sensitive to changes in illumination and facial expression. Kernel PCA overcomes many of these limitations by introducing nonlinear feature mapping [46].

Yadav et al. [47] proposed an FPGA implementation using 15 eigenfaces to preserve approximately 95% of the data variance, achieving efficient recognition suitable for low-power applications such as criminal identification.

Local Binary Pattern (LBP) is a lightweight texture-based feature extraction technique that is highly suitable for FPGA acceleration. It encodes local pixel neighborhoods into histograms, which are compared using Chi-square or Euclidean distance measures. FPGA implementations employ pipelined architectures, on-chip BRAM, and fixed-point arithmetic to achieve **15–30 FPS** with low power consumption. Further optimizations include histogram reduction, smaller image resolutions, and early termination strategies. Studies by Khoi et al. [48], Napoleon and Alfalou [49], and Chen et al. [30] demonstrate accurate real-time FPGA-based LBP recognition.

Other feature extraction techniques also improve recognition performance. Scale-Invariant Feature Transform (SIFT) identifies scale- and rotation-invariant keypoints [50], Gabor filters extract detailed spatial frequency information [51], and Eigenfaces simplify recognition through dimensionality reduction [52]. Together, these approaches illustrate the diversity of classical facial recognition techniques.

Neural Network Models

Convolutional Neural Networks (CNNs) form the foundation of modern facial recognition systems by learning hierarchical features ranging from simple edges to complex facial characteristics [53]. CNN architectures typically consist of convolutional, pooling, and fully connected layers [6], with different architectures designed for specific applications. MobileNet [54], [55] is optimized for embedded systems, ResNet [56] supports deeper feature learning, AlexNet [57] remains a widely used baseline architecture, and YOLO [58] excels in real-time object detection.

On FPGAs, these models are commonly optimized through quantization, operator fusion, and parallel execution to achieve an effective balance between speed, accuracy, and energy efficiency. Studies have

***Corresponding Author:** Dr. Hareesh Kumar Email: drhareeshkumar.ece@sjcit.ac.in

demonstrated MobileNet implementations on Zynq and Arria FPGA platforms achieving up to **763.5 FPS** [19], [18], while MobileNetV2 achieves hundreds of frames per second with significantly lower power consumption [45], [23].

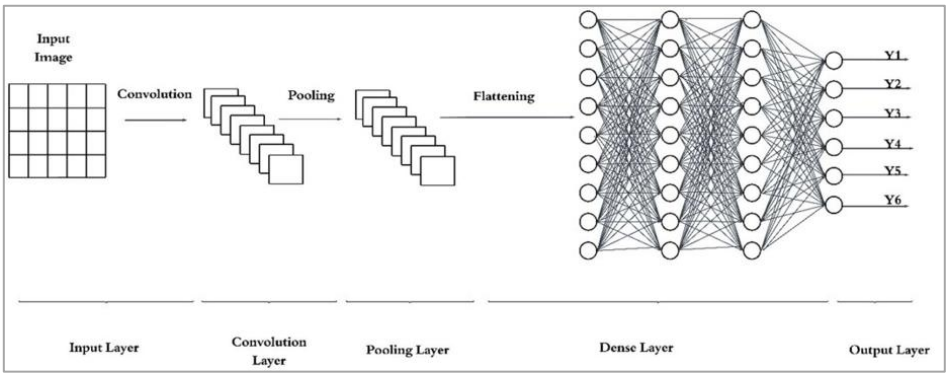

Fig. 2. Convolutional Neural Network Layers

ResNet implementations achieve high recognition accuracy while consuming substantially less energy than GPUs [15], [26]. Other architectures demonstrate different trade-offs. AlexNet has been accelerated to deliver multi-TOPS performance with significant energy savings on FPGA platforms [40], [17], whereas YOLO continues to enable real-time object detection on Ultra96v2 and ZCU102 platforms with over **90% precision** at low power [15], [43], [21].

Beyond these mainstream architectures, researchers have investigated FPGA implementations for emotion recognition [26], sparse CNNs [22], and resource-aware object detection [30]. Additional studies have explored DSP-based accelerators [27], compact CNN architectures [38], integer-only inference [25], RTL implementations [28], in-memory computing using ReRAM/SRAM [31], and WebAssembly-based FPGA acceleration [29]. Broader surveys and system-level studies [16], [32]–[37], [42], [44] further highlight these developments, demonstrating that FPGA-based CNN accelerators provide an effective balance between computational performance, energy efficiency, and scalability for practical facial recognition applications.

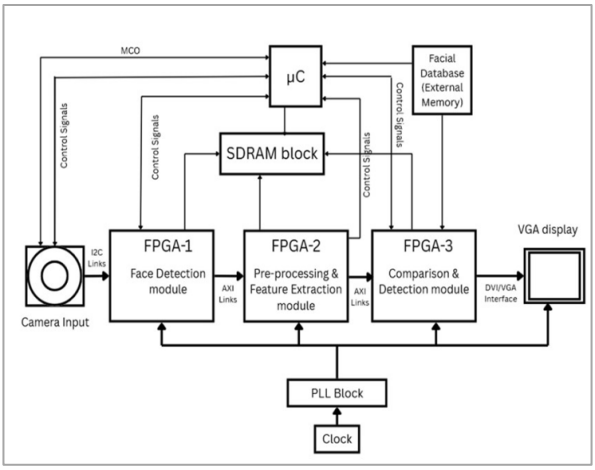

Fig. 3. Multi-Chip Face Recognition Accelerator

Table: Comparison of Pipelining and Parallelism-Based CNN/DNN Acceleration Methods

Sl. No.	Author(s)	Year	Ref.	Method	Dataset	Platform	Accuracy	Key Findings
1	Angelos Kyriakos et al.	2024	[49]	Pipelined CNN accelerator	MNIST	Virtex VC707	98.66%	High accuracy, efficient power
2	Rashed Al Amin et al.	2024	[52]	Review of pipelined CNNs	Custom	FPGA	N/A	High throughput, resource/latency tradeoff
3	Kasem Khalil et al.	2023	[51]	CNN with pipelining for energy efficiency	CIFAR-10	Arria 10 GX	98.81%	Power efficient, moderate speed
4	Rajesh Kedia et al.	2021	[53]	DNN tuning via DSE (pipeline exploration)	Standard DNNs	ZCU102 FPGA	Varies	28× fewer simulations
5	Jaemyung Kim et al.	2021	[25]	Lightweight CNN with LLTQ	FERPlus-A	ZC706 SoC	86.58%	10 FPS, 2.3 W
6	Monika Hemnani et al.	2016	[57]	Parallel imaging methods	Various	MPSoC, GPU	N/A	General review, parallel efficiency

***Corresponding Author:** Dr. Hareesh Kumar Email: drhareeshkumar.ece@sjcit.ac.in

Table: Comparison of Novel Architectures

Sl. No.	Author(s)	Year	Ref.	Architecture / Method	Dataset / Application	Platform	Accuracy	Key Findings
1	Jin Wu et al.	2022	[31]	In-memory compute (ReRAM/SRAM)	CIFAR, MNIST, ImageNet	Custom hardware	87.34%	53.55 TOPS/W
2	Jinyeol Kim et al.	2024	[27]	WebAssembly accelerator	Custom tasks	Cyclone-V FPGA	N/A	70× faster
3	Gitae Park et al.	2023	[50]	Multi-FPGA interface (Aurora)	N/A	ZCU102, VU19P	N/A	2000× UART
4	Linu M. Jiji & Ragimol	2016	[29]	Booth Multiplier + FCUs	DSP/Image tasks	Spartan-3 FPGA	N/A	38.6 ns delay, low area-delay

Table: Comparison of Quantization Approaches

Sl. No.	Author(s)	Year	Ref.	Quantization Approach / Method	Dataset	Platform	Accuracy	Key Findings
1	Jaemyung Kim et al.	2022	[25]	Integer-only quantization with CNN	GTSRB	ZC706 FPGA	96.95%	40 FPS, 960 MOPS
2	Peng Lv et al.	2020	[21]	YOLOv2 + Winograd + quantization	COCO	FPGA (unspecified)	N/A	Faster convolutions, lower resource utilization
3	Yongmei Zhou & Jingfei Jiang	2015	[56]	5-layer CNN using fixed-point HLS	MNIST	Zedboard FPGA	98%	16.58 GMACS, 16.42× faster

Table: Comparison of Pruning and Compression Approaches

Sl. No.	Author(s)	Year	Ref.	Pruning / Compression Method	Dataset	Platform	Accuracy	Key Findings
1	Xiaodi Yin et al.	2024	[22]	Block pruning with custom storage	Not specified	Zynq UltraScale+	65.30%	Balanced speed & power
2	Xiaofei Xie et al.	2022	[23]	MobileNet V2 with pruning	CIFAR-10	Zynq-7020 FPGA	86.03%	Low power, resource efficient

Table: Comparison of Memory and Dataflow Optimization Approaches

Sl. No.	Author(s)	Year	Ref.	Memory / Dataflow Technique	Dataset	Platform	Accuracy	Key Findings
1	Ru Ding et al.	2019	[28]	RTL-based CNN with DSP optimization	Face dataset	Cyclone V FPGA + ARM	<1% error	60× MACs/cycle, 80% DSP utilization
2	Qiao Yin et al.	2020	[46]	Efficient CNN scheduling	AlexNet	VCU118 FPGA	N/A	2.8 TOPS, 300 MHz
3	Lester Kalms et al.	2016	[58]	KAZE feature extraction	Iguazu, Boat	Zynq-7020 SoC	N/A	236.91× faster, 40% less memory

***Corresponding Author:** Dr. Hareesh Kumar Email: drhareeshkumar.ece@sjcit.ac.in

Table: Comparison of Other FPGA/SoC Techniques

Sl. No.	Author(s)	Year	Ref.	Technique / Method	Dataset	Platform	Accuracy	Key Findings
1	Akarsh Anil Kumar et al.	2024	[48]	Enhanced Canny edge detection	Various images	FPGA (Verilog)	N/A	33.4% processing time reduction
2	Ghattas Akkad	2024	[55]	Review of RISC-V/FPGA SoC trends	Not specific	RISC-V, FPGA, SoC	N/A	Highlights and future trends; no experiments
3	Dingkun Yang et al.	2023	[19]	Optimized MobileNet	Not specified	ZYNQ AXZU5EV	68.20%	22.3× CPU performance, 6.51 W power
4	Sang-Soo Park et al.	2023	[16]	GPU-optimized LeNet-5	MNIST	AMD APU	N/A	Fast, efficient implementation with low overhead
5	Victoria H. Kim et al.	2023	[24]	Custom CNN accelerator	CIFAR-10	PYNQ-Z1/Z2	N/A	16% better power efficiency, 15% faster
6	Huynh Phuc Nghi et al.	2022	[15]	ResNet-50, YOLOv3-tiny	VN Plate Dataset	Zedboard, Ultra96v2	93–94%	47.97 images/s, 2.967 W
7	Xing Liu et al.	2022	[18]	MobileNet with QL-HEFT	Not specified	Arria 10, Stratix 10	N/A	763.5 FPS
8	Kuan-Hung Chen et al.	2022	[47]	YOLOv4 edge detection	Supermarket dataset	ZCU102 FPGA	90%	Faster than GPU
9	P. D. Rao et al.	2022	[54]	Comparative study	Various	Multiple platforms	N/A	Review only
10	Lin Bai et al.	2018	[45]	Depthwise convolution	ImageNet	Arria 10 SoC	CPU-equivalent	266.6 FPS, 20× faster
11	Marco Bettoni et al.	2017	[17]	HW-CNN (AlexNet-based)	Not specified	Stratix V FPGA	84.70%	15× GPU and 3× CPU efficiency

DISCUSSION

In response to the need for embedded systems that support high-performance, low-latency facial recognition, we introduce a Multi-Chip Parallel Processing Accelerator Model. As shown in Fig. 3, the architecture separates the facial recognition pipeline across multiple interconnected FPGAs controlled by a central microcontroller. This design is intended to overcome the limitations of a single-chip implementation by exploiting task-level parallelism, thereby maximizing throughput while minimizing cost and power consumption.

Architectural Overview

The proposed design employs a pipelined three-stage FPGA architecture controlled by a master control unit, supported by a hybrid memory subsystem, and synchronized using a common clocking scheme. Three dedicated FPGAs are integrated into the accelerator, with each FPGA assigned to one of the major stages of the facial recognition pipeline.

The first FPGA processes the raw video input to detect and locate facial regions. The second FPGA performs normalization and feature extraction using a hardware-accelerated CNN. The third FPGA matches the extracted feature vectors with entries stored in the external database and generates the final identification decision.

A central microcontroller supervises the overall operation of the system. It configures the camera and FPGA modules, generates control signals, and manages database access, making hardware–software co-design an integral part of the architecture.

Data management is based on a hybrid memory model, where SDRAM is used as intermediate storage and a high-speed frame buffer, while the facial database is stored in external memory. This separation prevents bandwidth conflicts between high-rate video streams and database lookups. A master clock with PLL-based clock distribution ensures synchronized operation across all FPGAs, enabling reliable high-speed inter-chip communication over AXI links.

Operational Flow and Benefits

The accelerator operates in a highly pipelined manner. While the first FPGA detects a face in the current frame, the second FPGA simultaneously extracts features from the previously detected face, and the third FPGA performs feature matching and identification using

***Corresponding Author:** Dr. Hareesh Kumar Email: drhareeshkumar.ece@sjcit.ac.in

results from the preceding stage. This overlapping execution maximizes hardware utilization and significantly reduces overall inference latency.

The proposed multi-chip architecture provides an effective balance between performance, power efficiency, and cost. By utilizing several mid-range FPGAs, such as the Zynq-7020 or PYNQ-Z2, the design avoids the overhead associated with high-end devices while delivering greater efficiency than single-chip FPGA or CPU/GPU-based solutions. The architecture is inherently scalable, power-efficient, and optimized for lightweight CNN-based real-time facial recognition applications.

Limitations and Future Directions

Although the proposed model offers significant advantages, it also introduces challenges related to design complexity and the need for reliable hardware–software co-design. Future research should address several important areas.

Security and privacy remain critical concerns in real-time facial recognition systems. Future work should focus on hardware-level protection against adversarial attacks and the integration of privacy-preserving techniques, such as federated learning and homomorphic encryption, to safeguard biometric data during processing.

Another important direction is the trade-off between FPGA and ASIC platforms. While FPGAs provide flexibility and reconfigurability for evolving algorithms, ASICs can deliver higher performance and greater energy efficiency in large-scale deployments. Developing automated design flows that enable seamless migration from FPGA prototypes to ASIC implementations represents a promising research direction.

Finally, the integration of these accelerators into IoT ecosystems requires further investigation. Scalable communication protocols, efficient power management strategies, and flexible edge–cloud architectures must be developed to support secure, efficient, and distributed deployment in next-generation smart systems.

REFERENCES

1. Barnouti, W. E. M. N. H., & Al-Dabbagh, S. S. M. (2016). Face recognition: A literature review. *International Journal of Applied Information Systems, 11*(4), 21–31.
2. Yang, J., Qian, T., Zhang, F., & Khan, S. U. (2021). Real-time facial expression recognition based on edge computing. *IEEE Access, 9*, 76178–76190.
3. Wong, K. C. K., & Hunter, A. (2021). Enhancing low-energy facial recognition devices and abilities to improve security: A practical study. In *2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)* (pp. 184–189). IEEE.
4. Ben Fredj, H., Sghaier, S., & Souani, C. (2021). An efficient face recognition method using CNN. In *2021 International Conference of Women in Data Science at Taif University (WiDSTaif)* (pp. 1–5). IEEE.
5. Matai, J., Irturk, A., & Kastner, R. (2011). Design and implementation of an FPGA-based real-time face recognition system. In *2011 IEEE 19th Annual International Symposium on Field-Programmable Custom Computing Machines* (pp. 97–100). IEEE.
6. Kortli, Y., Jridi, M., Al Falou, A., & Atri, M. (2020). Face recognition systems: A survey. *Sensors, 20*(2).
7. Vidya Dharan, S., Khalil-Hani, M., & Shaikh-Husin, N. (2015). Hardware acceleration of a face detection system on FPGA. In *2015 IEEE Student Conference on Research and Development (SCOReD)* (pp. 283–288). IEEE.
8. Xie, L., Fan, X., Cao, W., & Wang, L. (2018). High throughput CNN accelerator design based on FPGA. In *2018 International Conference on Field-Programmable Technology (FPT)* (pp. 274–277). IEEE.
9. Baobaid, A., Meribout, M., Tiwari, V. K., & Pena, J. P. (2022). Hardware accelerators for real-time face recognition: A survey. *IEEE Access, 10*, 83723–83739.
10. Oukaira, A., Karime, A., Roberge, V., & Tarbouchi, M. (2023). New architecture for real-time image computing using parallel processing based on DSP/FPGA. In *2023 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)* (pp. 1–4). IEEE.
11. Dang, K., & Sharma, S. (2017). Review and comparison of face detection algorithms. In *2017 7th International Conference on Cloud Computing, Data Science and Engineering (Confluence)* (pp. 629–633). IEEE.
12. Hazim, N. (2016). Improve face recognition rate using different image pre-processing techniques. *American Journal of Engineering Research, 5*, 46–53.
13. Adnan, S., Ali, F., & Abdulmunem, A. A. (2020). Facial feature extraction for face recognition. *Journal of Physics: Conference Series, 1664*, Article 012050.
14. Oloyede, M., Hancke, G., & Myburgh, H. (2020). A review on face recognition systems: Recent approaches and challenges. *Multimedia Tools and Applications, 79*.
15. Nghi, H. P., & Thinh, T. N. (2022). A CNN-based vehicle identification solution in parking system with hardware accelerator on FPGA. In *2022 RIVF International Conference on Computing and Communication Technologies (RIVF)* (pp. 518–523). IEEE.
16. Park, S.-S., Hong, J.-H., & Chung, K.-S. (2017). Modified convolution neural network for highly effective parallel processing. In *2017 IEEE*

***Corresponding Author:** Dr. Hareesh Kumar Email: drhareeshkumar.ece@sjcit.ac.in

International Conference on Information Reuse and Integration (IRI) (pp. 325–331). IEEE.
17. Bettoni, M., Urgese, G., Kobayashi, Y., Macii, E., & Acquaviva, A. (2017). A convolutional neural network fully implemented on FPGA for embedded platforms. In *2017 New Generation of CAS (NGCAS)* (pp. 49–52). IEEE.
18. Liu, X., Yang, J., Zou, C., Chen, Q., Yan, X., Chen, Y., & Cai, C. (2022). Collaborative edge computing with FPGA-based CNN accelerators for energy-efficient and time-aware face tracking system. *IEEE Transactions on Computational Social Systems, 9*(1), 252–266.
19. Yang, D., & Luo, Z. (2023). A parallel processing CNN accelerator on embedded devices based on optimized MobileNet. *IEEE Internet of Things Journal, 10*(21), 18844–18852.
20. Hajirassouliha, A., Taberner, A., Nash, M., & Nielsen, P. (2018). Suitability of recent hardware accelerators (DSPs, FPGAs, and GPUs) for computer vision and image processing algorithms. *Signal Processing: Image Communication, 68*.
21. Lv, P., Liu, W., & Li, J. (2020). An FPGA-based accelerator implementation for YOLOv2 object detection using the Winograd algorithm. In *2020 International Conference Proceedings* (pp. 1894–1898).
22. Yin, X., Wu, Z., Li, D., Shen, C., & Liu, Y. (2023). An efficient hardware accelerator for block sparse convolutional neural networks on FPGA. *IEEE Embedded Systems Letters*.
23. Xie, X., Zhao, G., Wei, W., & Huang, W. (2022). MobileNetV2 accelerator for power- and speed-balanced embedded applications. In *2022 IEEE 2nd International Conference on Data Science and Computer Application (ICDSCA)* (pp. 134–139). IEEE.
24. Kim, V. H., & Choi, K. K. (2023). A reconfigurable CNN-based accelerator design for fast and energy-efficient object detection systems on mobile FPGA. *IEEE Access, 11*, 59438–59445.
25. Kim, J., Kang, J.-K., & Kim, Y. (2022). A low-cost fully integer-based CNN accelerator on FPGA for real-time traffic sign recognition. *IEEE Access, 10*, 84626–84634.
26. Kim, J., Kang, J.-K., & Kim, Y. (2021). A resource-efficient integer-arithmetic-only FPGA-based CNN accelerator for real-time facial emotion recognition. *IEEE Access, 9*, 104367–104381.
27. Jiji, L. M., & Ragimol. (2016). Embedded implementation of efficient accelerator architecture. In *2016 International Conference on Emerging Technological Trends (ICETT)* (pp. 1–6). IEEE.
28. Kim, J., Kim, R., Oh, J., & Lee, S. E. (2024). Accelerating embedded WebAssembly based on FPGA. In *2024 21st International SoC Design Conference (ISOCC)* (pp. 187–188). IEEE.
29. Ding, R., Su, G., Bai, G., Xu, W., Su, N., & Wu, X. (2019). An FPGA-based accelerator of convolutional neural networks for face feature extraction. In *2019 IEEE International Conference on Electron Devices and Solid-State Circuits (EDSSC)* (pp. 1–3). IEEE.
30. Chen, Y.-P., Liu, C.-H., Chou, K.-Y., & Wang, S.-Y. (2016). Real-time and low-memory multi-face detection system design based on Naïve Bayes classifier using FPGA. In *2016 International Automatic Control Conference (CACS)* (pp. 7–12). IEEE.
31. Wu, J., Zhao, B., Wen, H., & Zhao, Q. (2022). Design of neural network accelerator based on in-memory computing theory. In *2022 4th International Conference on Natural Language Processing (ICNLP)* (pp. 547–551). IEEE.
32. Kyriakos, A., Kitsakis, V., Louropoulos, A., Papatheofanous, E.-A., Patronas, I., & Reisis, D. (2019). High-performance accelerator for CNN applications. In *2019 29th International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS)* (pp. 135–140). IEEE.
33. Al Amin, R., & Obermaisser, R. (2024). Towards resource-efficient and low-latency CNN accelerators for FPGAs: Review and evaluation. In *2024 3rd International Conference on Embedded Systems and Artificial Intelligence (ESAI)* (pp. 1–10). IEEE.
34. Khalil, K., Kumar, A., & Bayoumi, M. (2023). Low-power convolutional neural network accelerator on FPGA. In *2023 IEEE 5th International Conference on Artificial Intelligence Circuits and Systems (AICAS)* (pp. 1–5). IEEE.
35. Kedia, R., Goel, S., Balakrishnan, M., Paul, K., & Sen, R. (2021). Design space exploration of FPGA-based systems with multiple DNN accelerators. *IEEE Embedded Systems Letters, 13*(3), 114–117.
36. Hemnani, M. (2016). Parallel processing techniques for high-performance image processing applications. In *2016 IEEE Students' Conference on Electrical, Electronics and Computer Science (SCEECS)* (pp. 1–4). IEEE.
37. Park, G., Taing, T., & Kim, H. (2023). High-speed FPGA-to-FPGA interface for a multi-chip CNN accelerator. In *2023 20th International SoC Design Conference (ISOCC)* (pp. 333–334). IEEE.
38. Zhou, Y., & Jiang, J. (2015). An FPGA-based accelerator implementation for deep convolutional neural networks. In *2015 4th International Conference on Computer Science and Network Technology (ICCSNT)* (Vol. 1, pp. 829–832). IEEE.
39. Yin, Q., Li, Y., Huang, H., Li, H., Zhang, Q., Cao, B., & Zhang, J. (2020). FPGA-based high-performance CNN accelerator architecture with high DSP utilization and efficient scheduling mode. In *2020 International Conference on High Performance Big Data and Intelligent Systems (HPBDIS)* (pp. 1–7). IEEE.
40. Kalms, L., Elhossini, A., & Juurlink, B. (2016). FPGA-based hardware accelerator for KAZE feature extraction algorithm. In *2016 International*

***Corresponding Author:** Dr. Hareesh Kumar Email: drhareeshkumar.ece@sjcit.ac.in

Conference on Field-Programmable Technology (FPT) (pp. 281–284). IEEE.
41. Kumar, A., Gireddy, S., Shashank, S., & Rajeshwari, B. (2024). Hardware accelerator for edge detection. In *2024 Conference Proceedings* (pp. 1–7).
42. Akkad, G., Mansour, A., & Inaty, E. (2024). Embedded deep learning accelerators: A survey on recent advances. *IEEE Transactions on Artificial Intelligence, 5*(5), 1954–1972.
43. Chen, K.-H., Wu, C.-B., Hwang, Y.-T., Chen, H.-L., Lin, J.-M., & Fan, C.-P. (2022). Hardware acceleration of YOLO-based convolutional neural network detector by deep learning processor unit for intelligent autonomous mover. In *2022 IET International Conference on Engineering Technologies and Applications (IET-ICETA)* (pp. 1–2). IET.
44. Dhilleswararao, P., Boppu, S., Manikandan, M. S., & Cenkeramaddi, L. R. (2022). Efficient hardware architectures for accelerating deep neural networks: Survey. *IEEE Access, 10*, 131788–131828.
45. Bai, L., Zhao, Y., & Huang, X. (2018). A CNN accelerator on FPGA using depthwise separable convolution. *IEEE Transactions on Circuits and Systems II: Express Briefs, 65*(10), 1415–1419.
46. Shah, J., Sharif, M., Raza, M., & Azeem, A. (2013). A survey: Linear and nonlinear PCA-based face recognition techniques. *International Arab Journal of Information Technology, 10*, 536–545.
47. Yadav, M., Koul, R., & Suneja, K. (2020). FPGA-based hardware design of PCA for face recognition. In *2020 7th International Conference on Signal Processing and Integrated Networks (SPIN)* (pp. 642–646). IEEE.
48. Khoi, P., Thien, L. H., & Viet, V. H. (2016). Face retrieval based on local binary pattern and its variants: A comprehensive study. *International Journal of Advanced Computer Science and Applications, 7*(6).
49. Napoléon, T., & Alfalou, A. (2014). Local binary patterns preprocessing for face identification/verification using the VanderLugt correlator. In D. Casasent & T.-H. Chao (Eds.), *Optical Pattern Recognition XXV* (Vol. 9094, Article 909408). SPIE.
50. Geng, C., & Jiang, X. (2009). SIFT features for face recognition. In *2009 2nd IEEE International Conference on Computer Science and Information Technology* (pp. 598–602). IEEE.
51. Okajima, K. (1998). Two-dimensional Gabor-type receptive field as derived by mutual information maximization. *Neural Networks, 11*(3), 441–447.
52. Ho, H.-T., Nguyen, L. V., Le, T. H. T., & Lee, O.-J. (2024). Face detection using eigenfaces: A comprehensive review. *IEEE Access, 12*, 118406–118426.
53. Vyas, A. S., Prajapati, H. B., & Dabhi, V. K. (2019). Survey on face expression recognition using CNN. In *2019 5th International Conference on Advanced Computing and Communication Systems (ICACCS)* (pp. 102–106). IEEE.
54. Shaila, S. G., Shivamma, D., Monica, U. S., & Tejashree, K. (2022). Facial expression recognition for compound emotions using MobileNet architecture. In *2022 International Conference on Artificial Intelligence and Data Engineering (AIDE)* (pp. 187–190). IEEE.
55. Hsiao, S.-F., & Tsai, B.-C. (2021). Efficient computation of depthwise separable convolution in MobileNet deep neural network models. In *2021 IEEE International Conference on Consumer Electronics–Taiwan (ICCE-TW)* (pp. 1–2). IEEE.
56. Anming, H. (2023). Research on facial expression recognition model based on ResNet network. In *2023 3rd Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS)* (pp. 272–275). IEEE.
57. Al-Amshawi, M. Z., Najim, A. H., Hassan, M. Y., Shutnan, W. A., & Abdulsattar, N. F. (2023). Recognizing human emotions from facial images with the help of AlexNet neural network, genetic algorithm, and decision tree classification. In *2023 Al-Sadiq International Conference on Communication and Information Technology (AICCIT)* (pp. 272–275). IEEE.
58. Tejaswi, K., Mokshith, D., Sai Pradeep, E., Mahesh Kumar, C., & Kumar, M. (2023). Facial expression recognition using YOLO. In *2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE)* (pp. 1–8). IEEE.

***Corresponding Author:** Dr. Hareesh Kumar Email: drhareeshkumar.ece@sjcit.ac.in