LIGHTWEIGHT CNN MODELS FOR REAL-TIME IMAGE PROCESSING ON EDGE DEVICES: PERFORMANCE AWARENESS

Habi Patrick[1], Mary Mathews[2], Dr. Shailja Shukla[3], Dr. Nizar Banu[4]

[1] Asst. Prof, Department of Computer Science and Application, The Bhopal School of Social Sciences /College, Bhopal, India.
[2] Asst. Prof, Department of Computer Science and Application, The Bhopal School of Social Sciences /College, Bhopal, India.
[3] Professor, Advance Computing, Sanjeev Agrawal Global Educational/University, Bhopal, India.
[4]Professor, Department of Computer Science, Christ Deemed to be University, Bangalore, India
Email:[1]habi_patrick@yahoo.co.in,[2]marymathews207@gmail.com,[3]shailja.sharma2300@gmail,[4]nizar.banu@christ university.in.
Orchid Id number: [1] 0009-0009-0103-1443

Abstract
The hasty and swift growth in the world of Internet of Things (IoT) and edge computing devices has shaped a new sense of robust demand for the real-time image processing and directly dealing on resources inhibited to the edge devices such as smartphones, drones, embedded systems and smart cameras. The conventional deep study on Convolutional Neural Networks (CNNs) shows highly accurate and precise output and which are computationally very expensive and inappropriate for deployment or positioning on the edge platforms due to limited memory, power of control and processing capabilities. This paper investigates lightweight CNN models designed for real-time image processing on the edge devices. We can analyze architectural optimizations, model compression methods, efficiency and accuracy. The most popular lightweight architectures such as MobileNet, ShuffleNet and EfficientNet are also studied, reviewed and compared. This study highlights things to see i.e. how lightweight is the CNNs enable low dormancy, energy competent as well as resourceful and privacy preserving edge devices intelligence despite the fact that it is maintaining viable and competitive performance.

Keywords: *Lightweight CNN, Edge Computing, Real-Time Image Processing, Model Compression, Mobile Deep Learning.*

1. Introduction

The increasing deployment of edge computing devices in applications such as surveillance & investigation systems, smart cities & metropolises, healthcare monitoring and autonomous vehicles has shifted computation from the cloud servers to the network edge devices. The Real-time image dealing out at the edge which diminishes dormancy, enhancing data discretion & privacy and minimizing network bandwidth convention. Nevertheless, edge devices stereotypically have efficient computational power, memory and battery life which is making the disposition of large-scale CNN models unfeasible and unreasonable.

The traditional CNN architectures such as VGGNet, ResNet and Inception have been achieved for high accurateness and precision but it involves masses of constraints and also billions of floating-point operations also known as FLOPs. These constraints have stirred and inspired the expansion of lightweight CNN models that maintain the equilibrium between performance and efficiency. The lightweight CNNs are explicitly premeditated to reduce the computational intricacy while still retaining adequate depictive capacity for real-time image handing out different tasks.

This paper also reconnoiters the strategies principles, techniques and applications and claim of lightweight CNN models for edge-based real-time image processing.

2. Background and Motivation

Edge Device Constraints

Edge devices are basically having restricted processing and dealing out power, memory and energy convenience and obtainability while working. The large CNN architectures such as VGG and ResNet which are involve in high constraint and parameter counts and the floating-point maneuvers helping in forecasting about augmented inference potential, latency and power consumption when organized or deployed on the edge hardware.

Prerequisite for Lightweight Models

The lightweight CNN models aim to diminish the model scope and also making computation multifaceted or complicated while preserving tolerable accurateness and precision. This also enables and make it possible for the real-time computer visualized applications such as blemish or flaw detection, wearable health nursing devices and portable vision assistance and backing within the edge device restrictions.

3. Lightweight CNN Design Principles

This model of CNN design focuses on edifice an efficient and competent convolutional neural network that can be constructed to achieve top level performance with the negligible computational charge which is making them suitable and apt for real-time and resource-controlled environments. The following are the examples of mobile, IoT, edge devices.

Depthwise Separable Convolutions: In this it replaces the standard of convolutions with depthwise and pointwise convolutions to radically change and reduce parameters and FLOPs which is used in MobileNet architecture model.

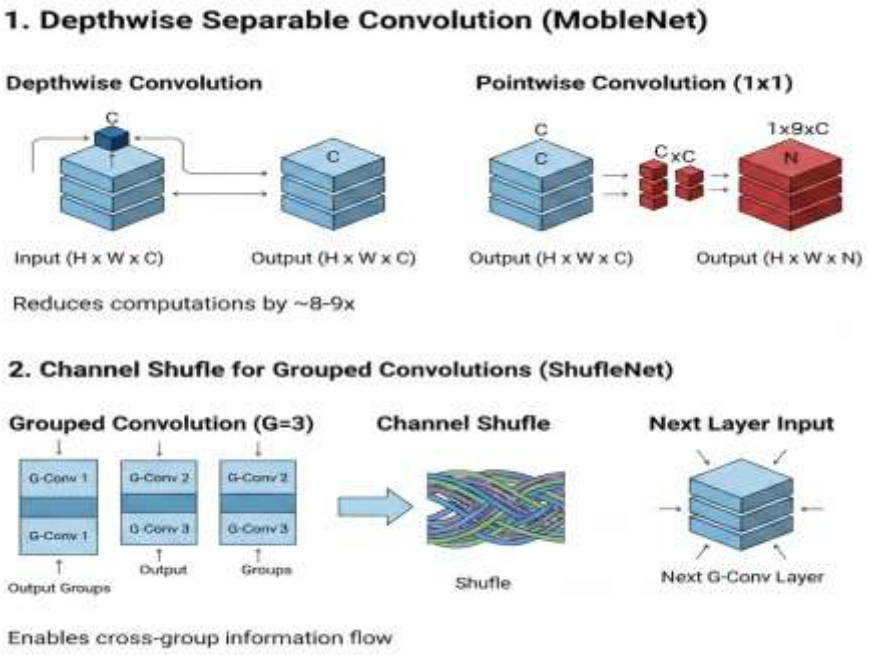
Reduced Model Parameters: Here it uses scarcer filters and even much slighter kernel sizes i.e. 3×3 model instead of 5×5 model to lower retention of memory and computation.

Bottleneck and Inverted Residual Blocks: In this model the compression features of dimensions are to reduced computation despite the fact that it preserves representational power and control.

Efficient Channel Utilization: Here in this channel pruning is applied and also network attention to eradicate the superfluous features which are not important.

Low-Precision Computation: In this it uses quantization techniques which is 8-bit or lesser to diminish memory convention and progress in the area of implication of speed.

Shallow but Effective Architectures: It is preferred scarcer layers with optimized feature extraction over very deep networks.



4. Popular Lightweight CNN Architectures

They are called neural network models which are unambiguously planned and designed to deliver good with accurateness and also with low complexity during the computation, making it idyllic for real-time and edge-based applications.

MobileNet (V1/V2/V3): The use of depth wise distinguishable intricacies and overturned residual blocks to reduce parameters meaningfully and making computation despite the fact which help in sustaining accurateness.

ShuffleNet: Provide service and employs the pointwise which helps in the assemblage of complication and shuffling of channels to permit sharing of information flow proficiently and with very low charge in the computational.

SqueezeNet: Introduction of "Fire segments" that can use 1×1 intricacies to reduce and diminish parameters even though the accomplishing and attaining the AlexNet-level accurateness with a very significantly smaller model proportion.

EfficientNet: It use multifarious clambering to balance the network penetration, girth, tenacity and firmness with accomplishment of high precision and also with scarcer restrictions.

GhostNet: It helps in generating "ghost" feature plans using via cheap linear operations and also dropping the redundancy of facts in feature extraction.

TinyCNN / Custom CNNs: It is low and narrow which is task precise architectures augmented and enhanced for a precise dataset which are usually used in the area of medicinal and EEG-based applications.

5. Prototypical Compression Techniques system

They are the methods which help in reducing the size, retention track and handling complex computations for the deep learning representations despite of the fact that it is important to preserve for their prognostic performance. These techniques and performance are specifically very significant for deploying the models on resources which are connected to inhibited devices such as mobile phones and edge devices and systems. These are some of the following techniques:

- Pruning or Clipping: In this method it removes the redundant or less significant weights, strainers or neurons from the links to diminish the model magnitude as well as computation.
- Quantization: This method converts the model weights and initiations from the high meticulousness e.g., 64-bit floating point to low meticulousness formats e.g., 8-bit integers and improving inference rapidity also making it reduce retention convention.
- Knowledge Distillation or Refinement: This technique trains a smaller "apprentice" model to mimic the productivities or the core depictions of a larger as well as proficient "coach" model.
- Mass Allocation: The method forces numerous network connections that can be shared with the equivalent or matching constraints making it reduce the entire number of inimitable masses or weights.
- Low Rank Factoring: It decomposes the bulky mass or weight matrices into less important matrices to diminish the computation and storage capacity.

6. Applications of Lightweight CNNs on Edge Devices

Some of the real time applications which are using this technology:

- Face Recognition: Mobile authentication systems
- Medical Imagery: Portable diagnostic devices
- Real-time Entity Recognition: Smart surveillance cameras and autonomous drones
- Industrial Scrutiny: Fault uncovering during manufacturing
- Smart Transportation: Traffic monitoring and pedestrian detection

7. Performance Valuation Metrics

They are quantitative measures which is used to assess and liken the efficiency, accurateness, dependability and competence of a system with the model type, its algorithm or even the process. They also offer a detached way to valuate and detect how well a system can executes its envisioned task by gauging conclusions against the predefined norms, criteria or grounded actuality.

Referring this machine learning and classification systems actually does the performance valuation by metrics which are used to regulate how precisely a model can envisages the outcomes, how good it is to handles errors and in what way it is vigorous when assigned pragmatic to hidden or unseen data. Commonly all metrics include accurateness, precision, recall, F1-score, sensitivity, specific and ROC-AUC each time capturing diverse facets of the model performance.

Lightweight CNN models are weighed through a amalgamation of efficiency and performance pointers to determine and come to point of their effectiveness and usefulness on edge devices. Inference latency reflects or show the time taken by the network to generate outputs which is basically vital for applications which requires fast comebacks. The model size is measured in megabytes which epitomizes the storage and memory requirements making it influence on deployment of the devices with limited power and resources. Computational intricacy is gauged using FLOPs which estimate the number of operations which is needed during inference. Accuracy is most important factor which is used to verify that the model maintenance, reliability, predictiveness of performance in spite of being compact. Energy consumption is also considered as lower-level power usage which is crucial for sustainable operation on battery driven or embedded systems used. Mutually these are some of the evaluation metrics which helps to determine whether a lightweight CNN is suitable for edge-based implementations of devices.

8. Challenges of Image Processing on Edge Devices

Deploying the CNNs techniques on the edge platforms presenting quite a few tests and challenges:

Limited Computational Resources: Edge devices lacks in making powerful GPUs or TPUs making it limited or restricted for the generating output and finishing of deep and complex CNN architectures.

Memory Restrictions: Here it requires large models with extensive memory for restrictions and intermediate feature maps which is exceeding edge device capabilities.

Power Usage: High level of computation which leads to increased energy usage making it reduce the battery lifespan in mobile and embedded systems.

Latent Necessities: Real-time applications make demand of low inference latency which is also problematic to accomplish with heavy models.

9. Imminent Research Guidelines

Imminent research will be focusing on building of the extra efficient and intelligent learning systems used in combination for advanced techniques:

• Neural Architecture Search (NAS) used to make robotically design compact interconnected structures that can attain high performance and reduced computational price.

• Edge cloud combined inference will also improve the scalable and real-time receptiveness by allocating the tasks based on latency sensitivity and dealing out complication.

• Event-driven and thwarting convolutional neural networks is discovered to enable low power usage and brain enthused computation that can process material only when important signal changes happen.

• The design of algorithms are specialized hardware which can support constricted optimization and also leading to quicker, energy resourceful and application explicit deep learning deployments.

10. Conclusion

The Lightweight CNN models play a very important and critical role in enabling of the real-time image which helps in processing on edge devices. It can be done by optimizing the architectural strategy and engaging the compression techniques which are applied in these models to accomplish an effective stability between effectiveness and exactness. As the edge computing devices continues to expand the lightweight CNNs techniques will endure to be scalable, low expectancy and energy resourceful intelligent systems.

References:

1. M. Abbas Abu Talib, S. Setumin, A. Izhar Che Ani, and S. J. Abu Bakar, "An analysis of lightweight convolutional neural network models for image classification task on edge device," in Proc. IEEE Int. Conf. Control Syst., Computing and Engineering, 2025, pp. 1–6, doi: 10.1109/ICCSCE65566.2025.11182642.
2. A. Surendar, "Lightweight CNN architecture for real-time image super-resolution in edge devices," Nat. J. Signal Image Process., vol. 1, no. 1, pp. 1–8, Mar. 2025, doi: 10.17051/NJSIP/01.01.01.
3. S. Naveen and M. R. Kounte, "Optimized convolutional neural network at the IoT edge for image detection using pruning and quantization," Multimedia Tools Appl., vol. 84, pp. 5435–5455, 2025, doi: 10.1007/s11042-024-20523-1.
4. F. Soria and C. C. Kingdon, "Lightweight CNN architectures for next-gen computing applications and edge device inference," Electron. Commun. Comput. Summit, vol. 2, no. 2, pp. 19–27, Jun. 2024.
5. Y. Kwon, W. Kim, and H. Kim, "HARD: Hardware-aware lightweight real-time semantic segmentation model deployable from edge to GPU," in Computer Vision – ACCV 2024, Lecture Notes in Computer Science, vol. 15478, Singapore: Springer, 2025, pp. 133–150, doi: 10.1007/978-981-96-0963-5_15.
6. Q. Meng and S. Lee, "Lightweight YOLOv5 with ShuffleNetV2 for rice disease detection in edge computing," Comput. Mater. Contin., vol. 86, no. 1, pp. 1–15, 2026.
7. L. Liu and Z. Xu, "Optimizing lightweight neural networks for efficient mobile edge computing," Sci. Rep., vol. 15, Article 22056, Jul. 2025, doi: 10.1038/s41598-025-04652-7.
8. A. Surendar, "Lightweight CNN Architecture for Real-Time Image Super-Resolution in Edge Devices," National Journal of Signal and Image Processing, 2025.
9. "Implementing Lightweight CNN Models for Real-Time Product Defect Detection with Edge Computing in Manufacturing Industries," IndiaAI, 2025.
10. M. Gao, "Toward Real-Time and Efficient Edge Intelligence: Advances and Challenges in Lightweight Machine Learning," ST Engineering Protection, 2025.
11. F. Soria and C. C. Kingdon, "Lightweight CNN Architectures for Next-Gen Computing Applications and Edge Device Inference," ECC Summit, 2024.
12. "Systematic Review of Lightweight Convolutional Neural Network Architectures on Edge Devices," International Journal of Research in Engineering and Science (IJRES).
13. "Optimizing Lightweight Neural Networks for Efficient Mobile Edge Computing," Nature Scientific Reports, 2025.
14. "MESNET: Integrating Lightweight CNNs and Real-Time Carbon Tracking for Sustainable Image Classification," Discover Sustainability, 2025.
15. "Optimized CNN at the IoT Edge Using Pruning and Quantization," Multimedia Tools and Applications, 2025.
16. "Lightweight Deep Learning Model Research Topics for Resource-Constrained Devices," S-Logix.
17. "EfficientNet B0 and Performance Analysis for Embedded Systems," SCITEPRESS, 2025.