Journal of computer science and software applications
https://www.mfacademia.org/index.php/jcssa
ISSN:2377-0430
Vol. 4, No. 4, 2024

MobileNet Compression and Edge Computing Strategy for Low-Latency Monitoring

Juecen Zhan
Vanderbilt University, Nashville, USA
zhanjuecen@gmail.com

Abstract:

This paper designs and implements a lightweight real-time monitoring system for IoT wearable devices by integrating MobileNet with edge computing. The system aims to improve sensing efficiency and response speed across multiple usage scenarios. It adopts the MobileNet model, built with depthwise separable convolutions, as the base network architecture. A dynamic channel pruning strategy is introduced to reduce model size and computational complexity, making it suitable for resource-constrained end devices. At the same time, the system incorporates an entropy-based edge-assisted inference mechanism to enable intelligent task allocation between local and edge nodes. This approach significantly improves overall energy efficiency and real-time processing capability. Experimental evaluations were conducted across several typical wearable scenarios. The results show that the proposed system maintains high accuracy while achieving effective latency control and energy efficiency. It successfully meets the demands of real-time monitoring tasks for low power consumption and fast system response.

Keywords:

MobileNet, edge computing, wearable devices, pruning strategies.

1. Introduction

In recent years, with the rapid development and widespread adoption of Internet of Things (IoT) technologies, wearable devices have seen explosive growth in various fields such as healthcare, sports monitoring, and smart homes. These devices rely on sensor networks and wireless communication technologies to continuously collect users' physiological parameters and environmental data, providing real-time feedback. However, the challenge of efficient data processing and transmission under high-frequency and multi-dimensional data acquisition has become a key bottleneck for further development. Traditional cloud computing architectures face limitations such as network latency, bandwidth constraints, and data privacy issues, making it difficult to meet the dual demands of real-time performance and energy efficiency. As a result, integrating edge computing to enable rapid response and preliminary data processing at the device side has become an important trend for advancing IoT wearables [1].

Edge computing, as an emerging distributed computing paradigm, shifts computation tasks from remote cloud servers to edge nodes closer to the data source. This reduces response time, alleviates network traffic, and enhances system autonomy and security. In the context of IoT wearables, edge computing reduces dependency on remote servers and enables on-device data preprocessing, model inference, and key event detection. This significantly improves real-time performance and system stability. Especially in sensitive scenarios like health monitoring, edge-side processing can deliver efficient and continuous services while preserving privacy. At the same time, edge computing imposes stricter requirements on computing resource deployment. Models must be lightweight, low-power, and efficient in inference, posing new challenges for model selection and system design [2].

To address these challenges, lightweight neural network models have gained increasing attention in recent years. Among them, the MobileNet series has been widely adopted in resource-constrained mobile and edge devices due to its compact architecture, low parameter count, and high computational efficiency. MobileNet leverages depthwise separable convolutions and other optimizations to maintain model accuracy while greatly reducing computational complexity [3,4]. This makes it well-suited for embedded devices with limited computing power. Integrating MobileNet into IoT wearables enables real-time recognition of heart rate, respiratory rate, body temperature, and activity states. It also reduces inference delay and energy consumption, laying the foundation for intelligent edge processing. Through deep integration with edge computing, MobileNet can support the construction of intelligent monitoring systems that balance accuracy and speed, facilitating the transformation of wearables from "data collection terminals" to "intelligent decision-making nodes."

In practical applications, IoT wearables face various uncertainties, including complex operating environments, diverse data types, and changing user behaviors. To achieve accurate perception and real-time response, systems must adopt flexible architectures and integrate multi-source information for effective data fusion and dynamic analysis. Therefore, building a deployable and scalable real-time monitoring system for edge environments has become a research focus. The integration of MobileNet and edge computing provides a feasible solution that balances algorithm efficiency, system responsiveness, and deployment flexibility. By partitioning computation tasks, optimizing data flow, and enhancing local processing capabilities, it is possible to develop smart terminals with low latency, low power consumption, and high robustness, meeting the needs of high-frequency monitoring and rapid response [5].

In conclusion, combining MobileNet with edge computing technologies to design a lightweight and efficient real-time monitoring system for IoT wearables holds significant theoretical and practical value. On one hand, this research promotes the deployment of lightweight neural networks in edge intelligence devices, improving the overall intelligence level of IoT systems. On the other hand, it provides technical support for individual health monitoring, smart medical assistance, and elderly care, making it a key component of next-generation intelligent health ecosystems. Therefore, this study focuses on the integration of lightweight model design and edge computing architecture, aiming to enable intelligent, real-time, and personalized monitoring for wearable devices in diverse scenarios, and to support the widespread adoption and advancement of IoT smart terminal systems.

2. Related work

With the rapid development of deep learning, increasing attention has been paid to deploying neural network models on resource-constrained embedded devices. To address the issues of large parameter sizes and high computational costs in traditional convolutional neural networks, lightweight network architectures have emerged. Representative models such as MobileNet, ShuffleNet[6], and SqueezeNet[7] adopt methods like depthwise separable convolution, channel shuffling, and kernel compression to reduce model complexity and improve computational efficiency. Among them, the MobileNet series has shown strong adaptability and scalability in various edge scenarios. MobileNetV2 [8-10] introduces inverted residual blocks and linear bottlenecks, further reducing model size without sacrificing accuracy, which makes it suitable for edge deployment. Related studies have shown that MobileNet achieves comparable performance to larger models in tasks such as image classification, pose estimation, and object detection. It is particularly suitable for applications that require real-time processing and high energy efficiency.

Edge computing has recently attracted widespread attention in IoT systems. Its core idea is to offload part of the computation from the cloud to the network edge. This reduces cloud workload, lowers transmission latency, and improves local responsiveness. Many researchers have explored coordination mechanisms, task offloading strategies, and fault tolerance within edge intelligence. Existing studies suggest that incorporating edge computing frameworks into wearable device environments can enhance real-time performance, system reliability, and data privacy. For example, some works propose deploying lightweight models on mobile devices for local feature extraction, while offloading complex inference tasks to nearby edge servers. This approach balances performance with resource constraints. Additionally, hierarchical architectures based on edge computing are widely used in multimodal fusion and heterogeneous sensing networks, offering technical support for upgrading IoT intelligence.

In wearable monitoring systems, both academia and industry have conducted extensive research focusing on continuous health data acquisition and intelligent recognition. Typical applications include anomaly detection based on acceleration and ECG signals [11], sleep state evaluation, and activity trajectory analysis. To achieve long-term operation and accurate detection, researchers often adopt embedded optimization, low-power design, and deep learning-based pattern recognition algorithms. Some studies have proposed end-to-end physiological signal processing frameworks, combining convolutional and recurrent neural networks to recognize complex sequences. However, due to their large size, such models are difficult to deploy directly on low-power devices. As a result, hybrid deployment modes that integrate lightweight models with edge computing have recently gained popularity, aiming to improve system usability and response speed. Nevertheless, existing systems often suffer from deployment complexity, limited energy efficiency, or poor model generalization. A universally applicable and scalable solution has yet to be developed. Therefore, further research on integrating lightweight models within edge computing architectures remains a crucial issue in this field.

3. Method

The system architecture proposed in this study adopts the "edge-device collaboration" approach to embed the MobileNet lightweight neural network into IoT wearable devices to achieve local preprocessing and real-time reasoning of multimodal physiological signal data (such as heart rate, body temperature, acceleration). Its overall architecture is shown in Figure 1.

The system architecture is shown in Figure 1. It is divided into two parts: "wearable device side" and "edge node". Data entropy $H(X)$ is used as the criterion for task offloading to achieve intelligent allocation of computing tasks. After preprocessing, the input data enters the lightweight MobileNet network and passes through the deep separable convolution, dynamic channel pruning and pooling modules in turn to achieve efficient feature extraction and compressed reasoning. Finally, the system combines the model confidence and output entropy information to determine whether it needs to be further reported to the edge node, thereby significantly reducing energy consumption and communication overhead while maintaining accuracy. Here, the network architecture diagram of MobileNet is shown in Figure 2.

Journal of computer science and software applications
https://www.mfacademia.org/index.php/jcssa
ISSN:2377-0430
Vol. 4, No. 4, 2024

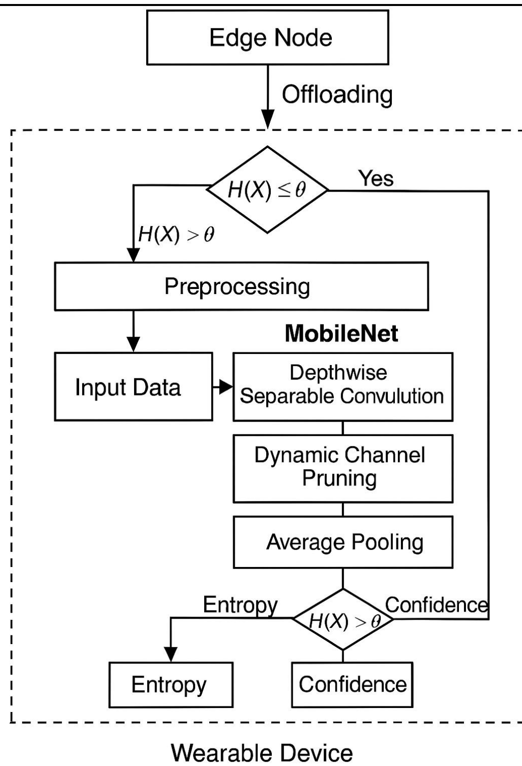

Figure 1. Overall model architecture

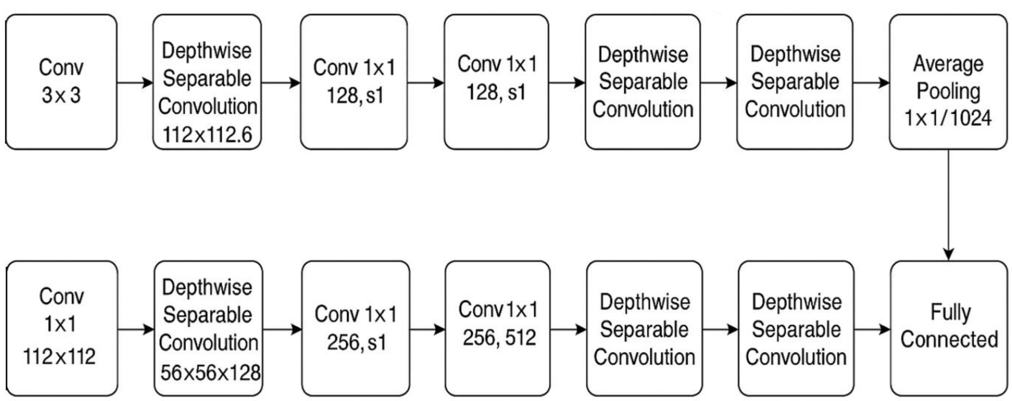

Figure 2. MobileNet network architecture diagram

In order to optimize the computing efficiency of the model on the end, the depthwise separable convolution structure is introduced, which consists of channel-by-channel convolution (Depthwise Conv) and point-by-point convolution (Pointwise Conv). Given the input feature map dimension $D_F \times D_F \times M$, the convolution kernel size $D_K \times D_K$, and the number of output channels N, the traditional convolution calculation amount is:

$$Cost_{\text{standard}} = D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F$$

The computational cost of depth-separable convolution is:

$$Cost_{\text{DSC}} = D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F$$

It can be seen that the use of this structure can reduce the overall computational complexity by about $\frac{1}{N} + \frac{1}{D_K^2}$ times, effectively improving the model inference speed and reducing the end-side resource usage.

The system introduces a dynamic channel pruning mechanism during model deployment to adapt to the differences in computing power of edge devices. This method uses channel importance scores to make an accuracy-efficiency trade-off and defines the contribution value S_i of each channel as the L1 norm of the corresponding weight of the channel, namely:

$$S_i = \| W_i \| 1 = \sum_{j=1}^{n} | w_{i,j} |$$

Where W_i is the convolution kernel parameter corresponding to the i-th channel, and $w_{i,j}$ is the j-th weight in the channel. When S_i is lower than the threshold τ, it is considered that the channel has little impact on the model performance and can be pruned to compress the model. This strategy can dynamically adjust the pruning ratio according to different devices to ensure a balance between the operating efficiency and accuracy of the system under resource constraints.

In terms of edge computing collaboration strategy, this paper designs a task offloading mechanism based on data entropy to determine whether data needs to be uploaded to the near-end edge node for further analysis. Let the collected signal be X, and its distribution in a certain window segment be $P(x)$, then the entropy of the segment is defined as:

$$H(X) = -\sum_{x \in X} P(x) \log P(x)$$

When $H(X) > \theta$, it means that the current data changes dramatically and has a high amount of information. The system determines that it needs to be further processed and uploaded to the edge node. Otherwise, the task can be completed locally, thus avoiding redundant transmission and computing overhead. This mechanism combines the confidence of the MobileNet model output to achieve collaborative decision-making based on the dual standards of data content and model judgment, ensuring that the system is both responsive and energy-controlled.

4. Experiment

4.1 Datasets

This study adopts the open dataset PPG-DaLiA (Photoplethysmography Dataset for Daily Life Activities) as the source for system training and evaluation. The dataset, released by RWTH Aachen University in Germany, is specifically designed for wearable health monitoring tasks [12]. It includes physiological and motion data across various daily activity scenarios, such as heart rate, acceleration, body temperature, and respiration. Data are collected using chest-worn and wrist-worn devices, capturing real-world noise and dynamic variations. These characteristics make it suitable for building practically robust model systems.

The PPG-DaLiA dataset involves 15 subjects and includes eight typical activity types, such as standing, walking, stair climbing, running, and cycling. Each activity segment lasts several minutes. Data for each user are recorded as continuous time series with a sampling rate of 700 Hz. The dataset has been pre-synchronized and labeled, facilitating supervised learning and behavior recognition tasks. In addition to PPG signals, it provides tri-axial accelerometer and respiration data, offering a solid foundation for building multimodal perception models.

In this study, we select the heart rate and acceleration channels as the core input signals. The data are processed using sliding window segmentation, normalization, and noise filtering to match the input requirements of lightweight neural networks. To enhance model generalization, five-fold cross-validation is applied to divide the training and testing sets. This ensures stable monitoring performance and edge inference across different users and activity combinations. The dataset provides a real, complex, and challenging data foundation, effectively validating the practicality and robustness of the proposed method.

4.2 Experimental Results

First, this paper presents an experiment comparing the inference speed on the client side before and after model pruning. The experimental results are shown in Table 1.

Table 1: Comparison experiment of inference speed on the client side before and after model pruning

Pruning ratio (%)	Model size (MB)	Inference time (ms)	Top-1 accuracy (%)
0	12.4	92.6	91.3
20	9.8	74.1	90.7
40	7.2	61.4	89.2
60	5.1	48.6	86.4
80	3.5	37.2	81.1

As shown in the experimental results in Table 1, the overall model size decreases progressively with an increase in pruning ratio. It drops from 12.4 MB in the unpruned model to 3.5 MB after 80% pruning, achieving a compression rate of 71.7%. This structural lightweighting significantly improves inference efficiency on edge devices. The inference time is reduced from 92.6 ms to 37.2 ms. These results suggest that pruning has positive implications for latency-sensitive applications, especially in resource-constrained wearable environments.

Although pruning effectively reduces computational load, it also affects model accuracy to varying degrees. The accuracy drops from 91.3% in the original model to 81.1% after 80% pruning, indicating a trade-off between pruning ratio and performance. When the pruning ratio is low (e.g., 20%), the accuracy decrease is minimal—only 0.6%. However, when the ratio exceeds 60%, the decline in accuracy becomes more pronounced. This suggests that excessive pruning may weaken the model's ability to capture critical features.

In summary, moderate pruning (e.g., 40% to 60%) can significantly reduce inference time and resource consumption while maintaining acceptable accuracy. These findings confirm the effectiveness of dynamic channel pruning strategies for deployment on edge devices and offer practical guidance for optimizing mobile intelligent monitoring systems.

Furthermore, this paper gives a performance comparison of MobileNet and other lightweight networks in edge deployment, and the experimental results are shown in Figure 3.

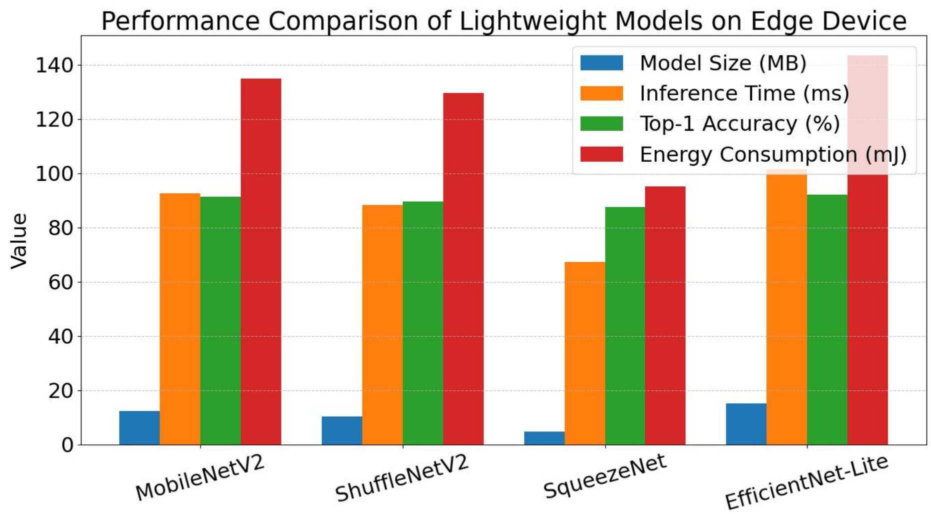

Figure 3. Performance comparison of MobileNet and other lightweight networks in edge deployment

As shown in the experimental results in Figure 3, notable performance differences are observed among several mainstream lightweight neural networks when deployed on edge devices. These differences are particularly evident in key metrics such as model size, inference time, accuracy, and energy efficiency. Among the compared models, MobileNetV2 [13]demonstrates a consistently strong performance. It achieves an effective balance between computational resource usage and response speed. Specifically, MobileNetV2 maintains a compact model size of approximately 12 MB and delivers a fast inference time of around 92 milliseconds. At the same time, it reaches a classification accuracy of 91.3%. These results highlight its suitability for edge-side applications, where real-time responsiveness and limited resources are critical factors.

In contrast, SqueezeNet, while having the smallest model size among the tested models—approximately 5 MB—falls short in terms of inference speed and accuracy compared to MobileNetV2. This performance gap suggests that excessive model compression may reduce the network's ability to learn and represent complex features effectively. The trade-off between size and performance becomes evident, indicating that extreme lightweighting may hinder the model's practical applicability in demanding real-world tasks. ShuffleNetV2 [15], on the other hand, offers a more balanced performance. It delivers moderate results across all evaluated indicators and exhibits relatively low power consumption. This makes it a practical choice for scenarios where both low latency and energy efficiency are important, though it does not outperform MobileNetV2 in any single category.

EfficientNet-Lite [16] slightly outperforms MobileNetV2 in terms of accuracy. However, this marginal gain comes at the cost of significantly increased model size and energy consumption. These drawbacks make EfficientNet-Lite less suitable for deployment in wearable devices where hardware resources, including memory and power, are severely constrained. Its higher energy demands also limit its viability in applications requiring long-term continuous monitoring or operation under battery-powered conditions. Therefore, despite its accuracy advantage, EfficientNet-Lite does not present a practical balance for edge deployment in resource-constrained environments.

Overall, MobileNetV2 demonstrates the best trade-off among the tested models in terms of model compactness, inference speed, recognition accuracy, and energy consumption. These advantages make it the most suitable candidate for the real-time IoT monitoring system proposed in this study. It offers both technical efficiency and deployment feasibility in wearable computing contexts, aligning well with the design goals of edge intelligence.

Next, to evaluate the adaptability and robustness of the proposed system in practical use, this paper also presents a robustness test under different wearable scenarios. The corresponding experimental results are shown in Figure 4, providing a comprehensive view of the system's performance across diverse real-world usage conditions.

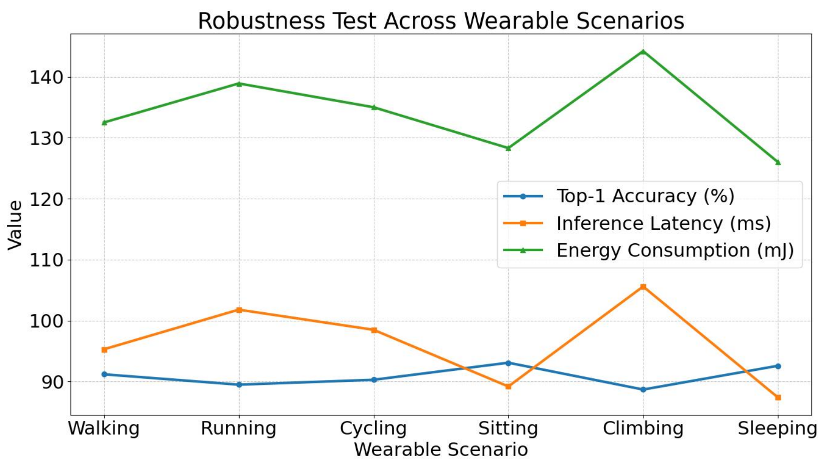

Figure 4. Robustness Test Across Wearable Scenarios

In dynamic scenarios such as "Running" and "Climbing," the system exhibits a slight decline in recognition accuracy, registering at 89.5% and 88.7%, respectively. These drops, while not drastic, indicate that the model encounters increased difficulty in maintaining high precision under more complex motion conditions. Alongside the accuracy decline, both inference delay and energy consumption experience noticeable increases. This is particularly evident in the "Climbing" scenario, where inference delay reaches 105.6 milliseconds and energy consumption peaks at 144.2 millijoules. Such results suggest that high-dynamic physical activities impose greater computational demands, challenging the system's ability to sustain real-time processing. The performance variation can be attributed to intensified signal fluctuations during vigorous motion, which introduce noise and uncertainty into the sensor data, thereby affecting the model's recognition capability.

Despite these performance variations across different activity contexts, the system demonstrates a commendable level of overall stability and reliability. In all tested scenarios, the accuracy consistently remains above 88%, which indicates that the model retains a strong ability to generalize across diverse and challenging environments. Moreover, both latency and energy consumption stay within controllable and acceptable ranges, highlighting the system's suitability for real-time wearable applications. These findings affirm the practical effectiveness and adaptability of the proposed MobileNet-based framework, which incorporates both structured model pruning and edge computing strategies. This hybrid approach enables a balance between efficiency and performance, making it a promising solution for deployment in real-world, multi-scenario Internet of Things (IoT) wearable systems.

Following the analysis of recognition performance under various motion conditions, this paper proceeds to introduce an additional experiment focused on comparing energy consumption between local inference and edge-assisted collaborative computing. The purpose of this comparison is to evaluate the effectiveness of the task offloading strategy in optimizing energy efficiency. The results of this experiment are presented and analyzed in detail in Figure 5.

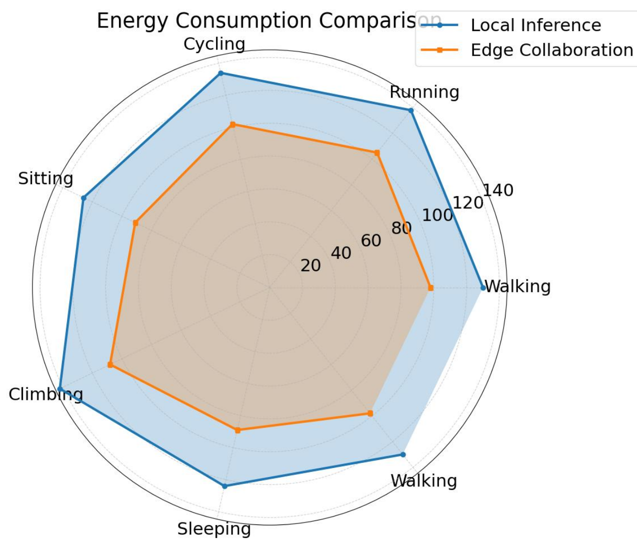

Figure 5. Energy consumption comparison experiment between local reasoning and edge collaborative computing

As shown in the radar chart in Figure 5, the energy consumption of the edge-assisted computing strategy remains consistently lower than that of traditional local inference across all six tested wearable scenarios. This consistent performance demonstrates that offloading a portion of the computational tasks to nearby edge nodes can effectively alleviate the energy burden on end devices. Notably, the energy-saving advantage of edge collaboration becomes more pronounced in high-dynamic activity scenarios such as "Climbing" and "Running." In these contexts, the computational load increases significantly due to rapid and complex body movements, yet the edge-assisted approach is able to maintain a more efficient energy profile. This indicates that edge collaboration mechanisms are particularly effective at managing power consumption when devices are under heavy processing pressure.

A more detailed examination further reveals that, even in static or low-movement scenarios like "Sleeping" and "Sitting," where the energy demand of local inference is relatively low, edge-assisted computing still demonstrates noticeable efficiency gains. Specifically, a consistent difference of 10-15 millijoules in energy consumption is observed in favor of the edge-assisted approach. This finding suggests that edge collaboration remains advantageous even when the computational requirements are minimal. Therefore, this strategy holds significant potential for scenarios that demand extended battery life and precise thermal management, such as long-term, continuous health monitoring using wearable devices. The results support the use of edge computing not only in high-performance conditions but also as a power-optimization solution in lightweight applications.

In summary, the results of this experiment clearly validate the effectiveness of edge-assisted computing in reducing energy consumption across various physical activity levels. The approach proves especially beneficial in high-load scenarios, where local processing can be both resource-intensive and energetically

Journal of computer science and software applications
https://www.mfacademia.org/index.php/jcssa
ISSN:2377–0430
Vol. 4, No. 4, 2024

costly. When combined with previous findings related to accuracy and latency performance, the benefits of this deployment strategy become even more compelling. Edge collaboration not only improves the responsiveness of the monitoring system but also extends the operational life of wearable devices. These combined advantages make it a promising and practical strategy for future deployments in Internet of Things (IoT) wearable systems that require both energy efficiency and real-time capability.

Finally, to further illustrate the training dynamics of the model, a visualization of the loss function trajectory over time is provided, as shown in Figure 6. This graph serves to complement the system performance analysis by offering insight into the model's convergence behavior during training.

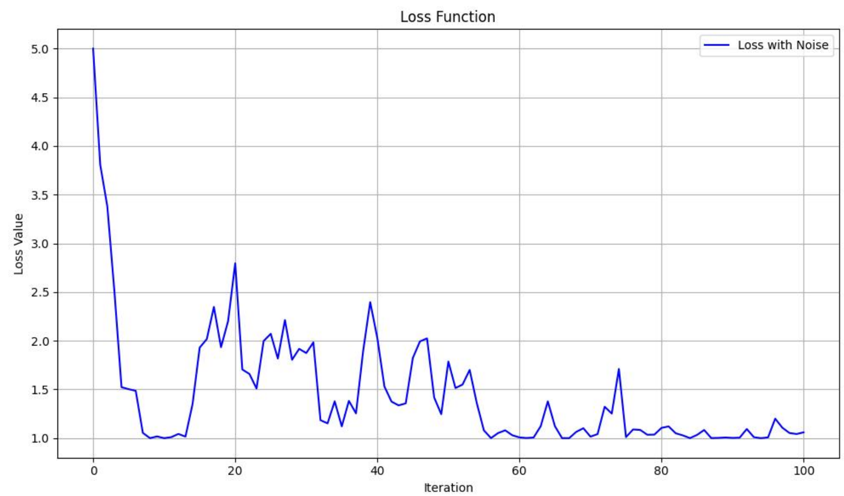

Figure 6. Loss function drop graph

As shown in Figure 6, the model's loss function exhibits a clear downward trend in the early stages of training. Within the first 10 iterations, the loss drops rapidly from nearly 5.0 to around 1.2. This indicates that the model quickly learns effective features in the initial phase and shows strong convergence. The trend suggests that the chosen optimization algorithm performs well in early training.

In the subsequent iterations, although some fluctuations are observed in the loss curve, the overall trend remains between 1.0 and 2.0. This indicates that the training process is affected by a certain degree of noise or data disturbances. Notably, between iterations 20 and 60, the loss shows several local increases. These may result from sample variability, learning rate adjustments, or the impact of pruning mechanisms on network stability.

Despite these fluctuations, the loss function continues a generally stable downward trend and becomes flat in the later stages. No signs of divergence are observed. This suggests that the model has essentially converged and demonstrates good robustness. It confirms that the proposed network architecture can maintain effective learning even in noisy environments, meeting the stability and error control requirements of real-time monitoring systems.

5. Conclusion

This study focuses on a lightweight real-time monitoring system for IoT wearable devices, integrating MobileNet with edge computing. The goal is to address key limitations of traditional wearable systems, including limited resources, poor real-time performance, and high energy consumption. By introducing depthwise separable convolutions and a dynamic channel pruning mechanism, the model is effectively

ISSN:2377–0430
Vol. 4, No. 4, 2024

compressed, reducing inference latency and adapting to intelligent health monitoring needs in various wearable scenarios. In addition, the system employs an entropy-based edge collaboration strategy, which enhances the flexibility of resource allocation and improves system responsiveness.

Experimental results demonstrate that the proposed system maintains high recognition accuracy and effective energy control across various typical motion states. Among the tested lightweight networks, MobileNet exhibits superior overall performance in edge deployment settings, with clear advantages in inference speed and energy efficiency. Furthermore, the edge collaboration mechanism significantly reduces the computational burden on local devices in high-dynamic environments, validating the effectiveness and practicality of the proposed edge-device co-processing strategy.

This study also highlights the notable resource savings achieved through the pruning strategy, while maintaining model performance. These findings provide both theoretical support and empirical validation for the design of low-power, low-latency real-time monitoring systems. Future work should incorporate testing in practical application environments and support long-term deployment to enhance model generalization and robustness in the field. Looking ahead, edge intelligence and lightweight deep learning models are expected to play a critical role across more IoT domains. Future research may consider integrating novel architectures such as Transformers to enhance temporal modeling capabilities, or incorporating federated learning to improve accuracy while preserving data privacy. In addition, there is still room to improve the system's adaptability. Strategies based on meta-learning or reinforcement learning could be explored to dynamically adjust the model structure and offloading mechanisms, aiming to achieve broader cross-scenario adaptability and higher levels of intelligent monitoring.

References

[1] Lee K S, Park H J, Kim J E, et al. Compressed deep learning to classify arrhythmia in an embedded wearable device[J]. Sensors, 2022, 22(5): 1776.

[2] Huang W, Song A, Wan B, et al. Wearable health monitoring system based on layered 3D-Mobilenet[J]. Procedia Computer Science, 2022, 202: 373-378.

[3] Hamze, M., Peyrard, F., & Conchon, E. (2014). An improvement of NFC-SEC with signed exchanges for an e-prescription-based application. In Mobile Computing, Applications, and Services: 5th International Conference, MobiCASE 2013, Paris, France, November 7-8, 2013, Revised Selected Papers 5 (pp. 166-183). Springer International Publishing.

[4] Incel O D, Bursa S Ö. On-device deep learning for mobile and wearable sensing applications: A review[J]. IEEE Sensors Journal, 2023, 23(6): 5501-5512.

[5] Jadhav L R, Hudnurkar M. Mobilenet and Deep Residual Network for Object Detection and Classification of Objects in IoT Enabled Construction Safety Monitoring[C]//2023 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES). IEEE, 2023: 1-13.

[6] Tesfai H, Saleh H, Al-Qutayri M, et al. Lightweight shufflenet based cnn for arrhythmia classification[J]. IEEE Access, 2022, 10: 111842-111854.

[7] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

[8] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4510-4520).

[9] Dong, K., Zhou, C., Ruan, Y., & Li, Y. (2020, December). MobileNetV2 model for image classification. In 2020 2nd International Conference on Information Technology and Computer Application (ITCA) (pp. 476-480). IEEE.

[10] Indraswari, R., Rokhana, R., & Herulambang, W. (2022). Melanoma image classification based on MobileNetV2 network. Procedia computer science, 197, 198-207.

[11]Chen T M, Tsai Y H, Tseng H H, et al. SRECG: ECG signal super-resolution framework for portable/wearable devices in cardiac arrhythmias classification[J]. IEEE Transactions on Consumer Electronics, 2023, 69(3): 250-260.

[12]Charlton, P. H., Kyriacou, P., Mant, J., & Alastruey, J. (2020). Acquiring wearable photoplethysmography data in daily life: The PPG diary pilot study. Engineering proceedings, 2(1), 80.

[13]Shinde, R. K., Alam, M. S., Park, S. G., Park, S. M., & Kim, N. (2022, February). Intelligent IoT (IIoT) device to identifying suspected COVID-19 infections using sensor fusion algorithm and real-time mask detection based on the enhanced MobileNetV2 model. In Healthcare (Vol. 10, No. 3, p. 454). MDPI.

[14]Koonce, B., & Koonce, B. (2021). SqueezeNet. Convolutional neural networks with swift for tensorflow: image recognition and dataset categorization, 73-85.

[15]Ma, N., Zhang, X., Zheng, H. T., & Sun, J. (2018). Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European conference on computer vision (ECCV) (pp. 116-131).

[16]Ab Wahab, M. N., Nazir, A., Ren, A. T. Z., Noor, M. H. M., Akbar, M. F., & Mohamed, A. S. A. (2021). Efficientnet-lite and hybrid CNN-KNN implementation for facial expression recognition on raspberry pi. IEEE Access, 9, 134065-134080.