Proceedings of CONF-MLA 2025 Symposium: Applied Artificial Intelligence Research
DOI: 10.54254/2755-2721/2025.BJ26597

Smart Camera Video Processing and Recognition System Based on Edge Computing

Xiaoxu Chen

Manchester Metropolitan Joint Institute, Hubei University, Wuhan, China
2077403020@qq.com

Abstract. The widespread adoption of smart cameras in areas such as security monitoring, intelligent transportation, and industrial quality inspection is fueled by the growing number of Internet of Things (IoT) devices and continuous progress in artificial intelligence (AI) algorithms. However, traditional video surveillance systems rely on cloud computing and face challenges like limited bandwidth, high latency, and privacy risks. Edge computing, a distributed architecture that brings computation and storage closer to the data source, offers an effective way to improve smart camera video processing and recognition. Therefore, the paper investigates smart camera video processing and recognition systems based on edge computing reviewing key technologies and implementation methods via recent research and typical applications. In addition, it examines the progress and features of edge platform architecture, video analysis and object detection algorithms, resource scheduling and energy management, model compression, and data security and privacy protection. The results show that edge-intelligent video systems are effective in reducing network load, lowering response latency, and improving the security of local data processing. However, they still face technical challenges in heterogeneous resource management, real-time scheduling, and multi-task collaboration. As such, this paper further reviews the main existing issues and offers a practical outlook on system optimization and future applications.

Keywords: Edge computing, Smart camera, Video processing, Target recognition

1. Introduction

The growth of smart security, intelligent transportation, and the urban Internet of Things (IoT) has sharply increased video processing demands, driving smart cameras to evolve into edge artificial intelligence (AI) terminals integrating sensing, analysis, and feedback. As such, video now accounts for over 80% of IoT data traffic [1]. Traditional cloud-based processing, which uploads raw video to servers, faces issues such as bandwidth strain, latency, and privacy risks, making it unsuitable for scenarios demanding low latency and high privacy. For this reason, edge computing has emerged as a solution. By placing computing resources near the data source, it enables cameras to perform tasks like object detection and behavior recognition locally, reducing latency, improving efficiency, and enhancing user privacy protection [2]. Combined with embedded AI chips such as Jetson and Coral Tensor Processing Units (TPUs), smart cameras can operate independently or collaborate with edge servers to handle complex scenarios like cross-view tracking and multi-camera coordination. This

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2025.BJ26597

shift of computation to the edge significantly improves system response speed and adaptability [3]. However, most current research still focuses on model compression and single-point deployment, with limited attention to co-design, integration, and privacy protection, which are key to reliable video recognition on constrained devices. Accordingly, this paper examines video processing and recognition in smart cameras under edge computing, reviewing hardware architectures, deployment strategies, algorithm optimization, and security design, while summarizing current progress and key challenges. Meanwhile, it highlights the practicality of edge platforms, algorithm adaptability, and deployment challenges, while outlining future directions in multimodal fusion, adaptive reasoning, and trusted AI to inform subsequent research and applications.

2. Fundamentals of edge computing and architecture of intelligent camera systems

2.1. The composition and framework of intelligent camera systems

As the key terminal of edge intelligence, smart cameras need to efficiently complete the closed-loop processing from video capture to preliminary analysis on the end side. The system typically consists of four modules, including video acquisition, computation processing, communication transmission, and data management, and forms a complete perception-to-decision pipeline via the coordination of data and control flows. The video acquisition module handles image sensing and preprocessing, ensuring input quality under varying lighting and motion conditions. The computational processing module performs image analysis, feature extraction, and local inference, to balance performance and energy efficiency. The communication module manages data exchange with the edge or cloud, adjusting bandwidth usage based on task demands. The data management module supports local caching, task scheduling, and model updates, boosting scalability and overall system efficiency. In addition, smart cameras must address power limits, harsh environments, and remote maintenance. Due to limited on-device capacity, edge platforms are necessary to handle complex tasks locally.

2.2. The empowerment of camera systems by edge computing

Based on multi-module collaboration, edge computing effectively supports the end-side inference of smart cameras by allocating computational resources nearby, boosting response time and processing performance. Compared with the traditional remote processing relying on the cloud, local inference can complete target detection and behavior analysis, significantly reduce latency and bandwidth consumption, and reduce the dependence on network stability. Currently, low-power AI chips paired with lightweight models are widely applied in real-time demanding scenarios, such as intelligent transportation and security surveillance. For example, the Jetson platform runs lightweight detection models to enable real-time local analysis with limited power consumption. For better performance, embedded platforms, fast inference engines, and model compression techniques are widely used to ensure stable device operation [5]. Edge computing can also complete the preliminary analysis on the end side and only upload key features or results, reducing network load and privacy leakage risk, which is suitable for crowded or private areas. However, limited computing power and storage make it challenging for a single edge node to handle large-scale analysis in complex scenarios, making cloud-edge collaboration a key development direction for intelligent video systems.

2.3. The evolution and structure of cloud-edge collaborative architectures

To address the limitations of edge devices in computing power and storage, smart camera systems increasingly adopt a three-layer “cloud-edge-end” cooperative architecture, as illustrated in Figure 1

DOI: 10.54254/2755-2721/2025.BJ26597

[6]. This architecture enables large-scale, high-precision video analysis via layered task allocation and the combined computing strengths of each layer. The end-side camera handles data collection and initial inference for rapid local response, while the edge node aggregates multiple video streams, performs regional analysis, and manages device coordination. Meanwhile, ,the cloud is responsible for model training, cross-scene optimization, and historical data mining. In this workflow, task flow moves upward from the end to the cloud. Conversely, control flow moves downward, with the cloud issuing models and commands. Thus, the edge and end collaborate in response, enabling intelligent sensing across time and space. Besides, edge nodes can share resources through device-to-device (D2D) communication to support collaborative inference among multiple cameras. This architecture combines the real-time responsiveness of the end, the near-field computing capability of the edge, and the powerful computational resources of the cloud, and has been widely applied in smart cities, traffic monitoring, and other scenarios requiring high real-time processing capability [7,8].



Figure 1. Cloud–edge–end collaborative architecture

3. Edge computing platform architecture and system implementation technologies

3.1. Embedded edge computing platforms and acceleration chips

In the cloud-edge-end collaborative architecture, embedded computing platforms and acceleration chips form the essential hardware basis for enabling edge computing. They provide the necessary computational power to process multiple video streams and perform deep learning inference in real time, all within constrained space and power budgets. At present, mainstream embedded platforms commonly use the ARM architecture, featuring multi-core CPUs and embedded GPUs to enable both general computing and visual processing. They are also well-suited for mainstream lightweight deep learning frameworks, supporting efficient deployment in edge scenarios. Widely used in areas like city surveillance and traffic monitoring, NVIDIA Jetson and Huawei's Ascend Atlas series are representative examples of current embedded platforms [9]. Paired with dedicated AI acceleration chips such as Edge TPU and Movidius, these platforms can greatly boost neural network inference performance while supporting low-power, high-concurrency deployment at the edge. In customized and highly flexible scenarios, such as in-vehicle or industrial inspection, FPGAs can be optimized for specific models by relying on programmable logic, while ASICs commonly used in large-scale deployments can achieve efficient inference through high integration and extreme power control. The actual choice should consider factors such as computational load, power constraints, cost, and ecosystem support. For example, road surveillance places greater emphasis on energy efficiency and

DOI: 10.54254/2755-2721/2025.BJ26597

cost, whereas tasks such as industrial quality inspection or high-precision analysis call for more powerful computing to ensure recognition accuracy [10].

3.2. Operating systems and lightweight deployment frameworks

In addition to embedded hardware, reliable operating systems and deployment frameworks play a vital role in enabling stable operation and efficient updates of AI on the end side of smart cameras. This is especially important, as edge intelligence scenarios place high demands on system real-time performance, stability, and resource scheduling. Embedded Linux distributions like Ubuntu Core and Buildroot, are widely used for video stream processing and general system management due to their customizability and strong hardware compatibility. In contrast, real-time operating systems (RTOS) like FreeRTOS and RT-Thread are better suited for latency-sensitive applications, offering millisecond- or even microsecond-level responsiveness via efficient multi-thread scheduling and interrupt handling. The efficient multi-thread scheduling and interrupt response mechanism supports millisecond or even microsecond fast processing. On top of the system layer, containerization and lightweight deployment solutions are key to enabling agile delivery of edge AI services. Tools like Docker and K3s encapsulate AI applications and dependencies into portable modules, supporting remote updates and rapid deployment. Besides, open-source frameworks such as EdgeX Foundry enhance software-hardware decoupling, lowering the adaptation cost across different edge hardware platforms. Limited edge computing power requires lightweight and efficient inference frameworks. In this context, operating systems and containers are crucial for stable, scalable deployment [11].

3.3. System scheduling and energy consumption optimization strategies

Supported by hardware and software platforms and containerized deployment, efficient system scheduling and energy efficiency management is a key part of ensuring sustainable operation of edge reasoning. In the face of multiple video streams, limited computing power and dynamic scene changes, traditional static task scheduling is difficult to adapt to the complex and changing load requirements at the edge. Dynamic-aware scheduling mechanism flexibly adjusts resource allocation and processing priority by real-time sensing task urgency, content complexity and node load status. For example, the VideoEdge system combines frame content analysis and multi-layer response strategies to switch task execution between local and cloud based on scene changes, reducing bandwidth pressure and node power consumption [12]. Heterogeneous arithmetic scheduling further improves the hardware utilization of edge devices, where CPU, GPU, NPU, and other arithmetic units can be dynamically allocated according to the task characteristics and model complexity, and the CoEdge framework dynamically switches and load balances the tasks by estimating the latency and power consumption of each processing unit, which is a good balance between real-time performance and energy efficiency [13]. To reduce the overhead of repetitive inference, feature caching and reuse mechanisms are also common optimizations. For low-dynamic video streams, the system can cache the key region features of the previous frame at the edge node and reuse them to reduce redundant computation; while in high dynamic environments, the cache can be dynamically refreshed in combination with the change detection algorithm to ensure the accuracy of the results. At the hardware level, energy-efficiency management techniques such as Dynamic Voltage Frequency Regulation (DVFS) automatically adjust the voltage and operating frequency of the computing units through real-time load prediction, balancing the arithmetic power and power consumption without affecting the core reasoning task [14].

DOI: 10.54254/2755-2721/2025.BJ26597

4. Edge-side video recognition and data security mechanisms

4.1. Video task classification and recognition process

In edge environments, video analytics involves tasks like object detection, classification, tracking, and behavior analysis. Through effective module coordination, edge-side video analytics forms a closed-loop framework that enables intelligent decision-making directly from real-time visual input. At its core, this approach leverages advanced object detection techniques to identify key targets, such as pedestrians and vehicles, in complex scenes with high efficiency. Thereafter, classification and attribute recognition methods extract detailed semantic information to distinguish between fine-grained object categories. To preserve temporal consistency and support deeper insight into dynamic behaviors, multi-object tracking algorithms generate continuous trajectories across video frames. These interlinked methods collectively enable the system to recognize patterns, infer intent, and detect anomalies, thus enhancing the responsiveness and intelligence of edge video systems. In public safety, intelligent transportation, and related domains, event analysis plays a critical role in identifying complex behaviors, such as retrograde motion, perimeter breaches, or falls, which can in turn trigger timely alerts to enhance system responsiveness and practical utility. Besides, specialized functions like facial recognition and crowd analysis are often integrated with detection and tracking modules to support a range of application scenarios, including access control and passenger flow monitoring. To address the constraints of edge computing, various input-stage pre-processing techniques like dynamic frame rate adjustment, region of interest (ROI) extraction, and resolution compression are widely adopted to streamline computations and reduce energy consumption. By minimizing redundancy, focusing on key targets, and lowering data volume, these methods enable real-time and accurate video analysis even under limited hardware conditions, as shown in Figure 2.

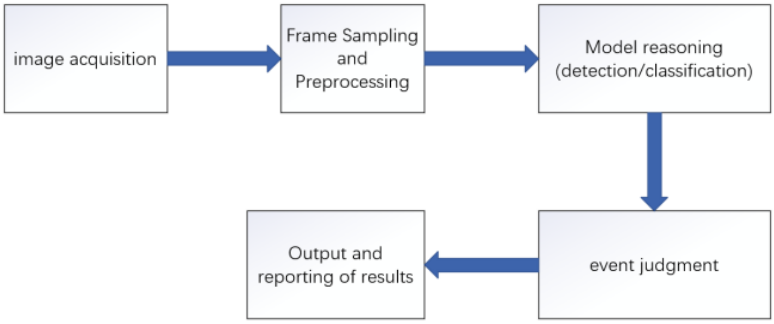

Figure 2. Edge-side video processing pipeline

4.2. Mainstream recognition algorithms and lightweight models

The advancement of edge video recognition is largely driven by deep learning, where lightweight convolutional neural network (CNN) architectures play a key role in enabling efficient deployment on embedded smart camera systems. In light of resource constraints inherent to edge computing, efficient neural architectures have been actively explored. For detection tasks, YOLO models have become a mainstream choice in traffic and surveillance applications, benefiting from anchor-free strategies and multi-scale feature fusion. In parallel, MobileNet variants achieve superior parameter efficiency and inference performance via depthwise separable convolutions and attention modules, making them ideal for embedded deployment. These models are particularly well-suited for edge applications such as face recognition and action classification, where computational efficiency and

DOI: 10.54254/2755-2721/2025.BJ26597

real-time responsiveness are essential. Furthermore, architectures such as ShuffleNet and GhostNet minimize computational load and memory bandwidth by using advanced convolutional strategies like channel grouping and feature reuse. These models are well-suited for multitask scenarios, such as pedestrian detection and behavior analysis, where accuracy and energy efficiency are critical. For tasks requiring multi-scale detection and segmentation, EfficientDet integrates the BiFPN structure to enhance small object recognition, while lightweight segmentation networks like PP-LiteSeg achieve an effective trade-off between speed and accuracy in applications such as industrial defect inspection and traffic sign recognition. In recent years, lightweight visual Transformer architectures like MobileViT and EfficientFormer have been developed to incorporate long-range dependencies into the local perception framework of CNNs, enhancing global modeling capabilities at the edge. Concurrently, neural architecture search (NAS) like TinyNAS and Once-for-All (OFA) enables fast generation of hardware-adaptive models, improving flexibility and deployment on edge devices.

4.3. Real-time performance optimization and model compression

In edge camera applications, achieving real-time and efficient performance for complex video tasks depends on model compression and structural optimization. To reduce computational and memory overhead, pruning is commonly employed in edge-optimized models like YOLO and ResNet by eliminating redundant neurons and channels. Quantization accelerates inference and improves computational efficiency by compressing weights and activations from 32-bit floating-point to 8-bit or lower precision integers, with minimal accuracy loss. Meanwhile, knowledge distillation, which leverages a high-capacity teacher network to supervise a compact student model, offers an effective trade-off between model size and representational power, particularly in recognition tasks requiring both efficiency and accuracy. In addition to manual optimization, NAS has become an important tool for edge model design in recent years, combining hardware constraints and multi-objective optimization to generate adaptive model architectures with a better balance of power, storage, and real-time performance. Meanwhile, event-driven inference and dynamic scheduling mechanisms are adopted in certain scenarios. By triggering inference only upon relevant visual changes or detected events, these methods reduce redundant computations. Besides, they cooperate with hardware-level load sensing and power management to support sustainable edge AI operations.

4.4. Video data security and privacy protection

Despite the advantages of reduced latency and bandwidth usage, the extensive deployment and open interfaces of smart cameras continue to pose risks to data, model, and hardware security, especially in sensitive environments such as transportation and healthcare. Empirical studies have indicated that physical attacks, such as adversarial patterns or light manipulation, can disrupt model inference, while insufficient local inference and weak transmission encryption may lead to leakage of facial data and location information [17]. To protect identity privacy, differential privacy techniques such as Differential Identity Protection (DIP) apply controlled perturbations to local features, striking a balance between recognition accuracy and anonymity [18]. However, threats like model inversion and leakage during collaborative inference still pose challenges. Though techniques such as task pruning and phased reasoning (e.g., PATROL, SecoInfer) help reduce exposure, they require careful balancing between system performance and security guarantees [19,20]. Complementing these software-level defenses, Trusted Execution Environments (TEE) and hardware-level isolation have emerged as critical safeguards against side-channel and architectural vulnerabilities such as Spectre and Meltdown [21]. A commonly adopted strategy is to perform identity recognition locally and

DOI: 10.54254/2755-2721/2025.BJ26597

transmit only structured results to the cloud, which necessitates lightweight models and efficient inference frameworks. Homomorphic encryption allows direct computation on encrypted data while ensuring data confidentiality. For example, dedicated system-on-chip architectures can boost energy efficiency up to 6,000 times, while CNNs with homomorphic encryption support secure inference with manageable trade-offs, though ciphertext computation remains a key bottleneck. Federated learning mitigates privacy risks by conducting local training and sharing only model parameters, thus eliminating the need to upload raw data; its applicability has been shown in multi-camera edge scenarios [22-24]. Moreover, the integration of differential privacy, fuzzy obfuscation, and trusted execution environments like ARM TrustZone and Intel SGX has increasingly reinforced edge video system security in public surveillance and behavioral analysis [25].

5. Future directions and technical challenges

5.1. Future directions

The advancement of edge computing enables smart cameras to integrate multimodal sensing, which greatly strengthens their perception and recognition capabilities in complex environments. Notably, the integration of infrared and visible spectrum data markedly boosts the effectiveness of nighttime surveillance and boosts perception reliability. The multi-scale edge gradient enhancement method facilitates IR-assisted night surveillance on edge devices, demonstrating the practical advantages of multimodal fusion [26]. To further optimize performance under resource constraints, smart cameras are adopting adaptive reasoning mechanisms that adjust model structure and inference depth based on scene complexity. In particular, the Adaptive Reasoning Model reduces redundant inference, lowers token usage, and improves both training efficiency and resource utilization [27]. Besides, the evolution of the end-edge-cloud architecture has enabled edge devices to collaborate across nodes in both data processing and model execution. By sharing recognition results, multi-camera systems improve cross-view accuracy, while systems like CONVINCE reduce redundancy and bandwidth consumption via task offloading and collaborative inference [28]. Meanwhile, The cloud manages historical data and models, while a semantics-driven edge-cloud framework optimizes task division, allowing edge devices to offload tasks only when necessary [29]. Driven by regulations like GDPR, edge AI must ensure privacy and compliance, thus leading to the widespread adoption of federated learning, differential privacy, and cryptographic inference. Federated learning, differential privacy, and trusted hardware ensure over 90% accuracy in camera tasks with strong privacy protection [30].

5.2. Technical challenges

Edge smart cameras are challenged by limited computational capacity in the face of increasingly complex models, insufficient generalization across varying scenes, inconsistent standards due to hardware-software heterogeneity, and escalating system security requirements. Firstly, the rising complexity of video analysis tasks requires smart cameras to support multi-task recognition and object tracking with high accuracy and low latency. However, constrained by size and power, edge devices cannot directly run large-scale deep neural networks. Achieving accurate, low-power, and low-latency inference has thus become a core deployment challenge. Secondly, smart cameras are deployed in diverse environments such as urban roads, industrial plants, and commercial spaces, yet current models often struggle to generalize effectively across scenarios. Their limited cross-scene adaptability often results in performance degradation or even recognition failure under data-scarce conditions, making improved generalization a core challenge for large-scale deployment. Moreover,

DOI: 10.54254/2755-2721/2025.BJ26597

the heterogeneity of hardware and software lacks unified standards. Edge AI systems run on various hardware platforms (e.g., NPU, GPU, FPGA) and software frameworks (e.g., TensorRT, TFLite), with significant differences in chip architectures, arithmetic libraries, and deployment interfaces across vendors. This fragmentation hinders model deployment and cross-platform migration. The absence of unified model formats, API standards, and inference specifications severely limits system scalability and maintainability. Finally, system security remains a critical challenge. In edge deployment environments, smart cameras are vulnerable to physical attacks, malicious firmware tampering, and model theft. As models grow in complexity and scale, traditional protection methods are no longer sufficient to address all potential attack surfaces. Therefore, a secure edge system with authentication, traceability, and update mechanisms is essential for reliable operation.

6. Conclusion

This paper reviews smart camera video processing and recognition under edge computing, covering embedded platforms, AI acceleration hardware, model optimization techniques such as compression, quantization, and distillation, as well as security and privacy measures at the edge. The study shows that edge computing boosts the real-time performance and privacy protection of smart cameras but remains constrained by limited computing resources and the trade-off between model complexity and inference efficiency. Additionally, weak generalization across scenarios and the absence of unified standards for heterogeneous software and hardware hinder system versatility and scalability. In addition, system security remains a key challenge in edge intelligence deployment. Future efforts should investigate the synergistic optimization of multimodal perception, adaptive inference, and edge-cloud collaboration, integrated with trusted AI and privacy-preserving computing to enhance system intelligence and security. As a core driver of smart camera development, edge intelligent vision technology holds significant research value and broad application potential.

References

[1] Edge AI Technology Report. (2023). Edge AI 2023 Report: The Era of AI Inference. Edge Industry Review. https: //www.wevolver.com/article/2023-edge-ai-technology-report
[2] Hu, M., Luo, Z., Pasdar, A., et al. (2023). Edge-based video analytics: A survey. arXiv: 2303.14329.
[3] Reddi, V.J. (2023). Generative AI at the edge: Challenges and opportunities. ACM Queue, 23(2), 79-137.
[4] Xu, R., Razavi, S., & Zheng, R. (2023). Edge video analytics: A survey on applications, systems and enabling techniques. IEEE Communications Surveys & Tutorials, 25(4), 2951-2982.
[5] Swaminathan, T.P., Silver, C., Akilan, T., & Kumar, J. (2025). Benchmarking deep learning models on nvidia jetson nano for real-time systems: An empirical investigation. Procedia Computer Science, 260, 906-913.
[6] Gong, L., Yang, H., Fang, G., et al. (2025). A survey on video analytics in cloud-edge-terminal collaborative systems. arXiv: 2502.06581.
[7] Hudson, N., et al. (2024). QoS-aware edge AI placement and scheduling with multiple implementations in FaaS-based edge computing. Future Generation Computer Systems, 157, 250-263.
[8] Nan, Y., Jiang, S., & Li, M. (2023). Large-scale video analytics with cloud-edge collaborative continuous learning. ACM Transactions on Sensor Networks, 20(1), 1-23.
[9] Sarvajcz, K., Ari, L., & Menyhart, J. (2024). AI on the road: NVIDIA Jetson Nano-powered computer vision-based system for real-time pedestrian and priority sign detection. Applied Sciences, 14(4), 1440.
[10] Choe, C., Choe, M., & Jung, S. (2023). Run your 3D object detector on NVIDIA Jetson platforms: A benchmark analysis. Sensors, 23(8), 4005.
[11] NVIDIA. (2023). TensorRT: High-performance deep learning inference optimizer and runtime. Retrieved from https: //developer.nvidia.com/tensorrt
[12] Hung, C.C., et al. (2018). VideoEdge: Processing camera streams using hierarchical clusters. In 2018 IEEE/ACM Symposium on Edge Computing (SEC), 115-131).

DOI: 10.54254/2755-2721/2025.BJ26597

[13] Zeng, L., Chen, X., Zhou, Z., Yang, L., & Zhang, J. (2020). CoEdge: Cooperative DNN inference with adaptive workload partitioning over heterogeneous edge devices. IEEE/ACM Transactions on Networking, 29(2), 595-608.
[14] Chen, Y.H., Krishna, T., Emer, J.S., & Sze, V. (2016). Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks. IEEE Journal of Solid-State Circuits, 52(1), 127-138.
[15] Mehta, S., & Rastegari, M. (2021). MobileViT: Light-weight, general-purpose, and mobile-friendly vision transformer. arXiv: 2110.02178.
[16] Li, Y., Yuan, G., Wen, Y., et al. (2022). EfficientFormer: Vision transformers at MobileNet speed. In Advances in Neural Information Processing Systems, 35, 12934-12949.
[17] Guesmi, A., et al. (2023). Physical adversarial attacks for camera-based smart systems: Current trends, categorization, applications, research challenges, and future outlook. IEEE Access, 11, 109617-109668.
[18] Nagasubramaniam, P., Wu, C., Sun, Y., et al. (2024). Privacy-preserving live video analytics for drones via edge computing. Applied Sciences, 14(22), 10254.
[19] Ding, S., et al. (2024). PATROL: Privacy-oriented pruning for collaborative inference against model inversion attacks. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 4716-4725.
[20] Yao, Y., et al. (2024). SecoInfer: Secure DNN end-edge collaborative inference framework optimizing privacy and latency. ACM Transactions on Sensor Networks, 20(6), 1-29.
[21] Sehatbakhsh, N., et al. (2025). Secure artificial intelligence at the edge. Philosophical Transactions A, 383(2288).
[22] Azad, Z., Yang, G., Agrawal, R., et al. (2023). RISE: RISC-V SoC for en/decryption acceleration on the edge for homomorphic encryption. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 31(10), 1523-1536.
[23] Ye, Z., Huang, T., Wang, T., et al. (2024). HTCNN: High-throughput batch CNN inference with homomorphic encryption for edge computing. Cryptology ePrint Archive.
[24] Xu, R., Razavi, S., & Zheng, R. (2023). Edge video analytics: A survey on applications, systems and enabling techniques. IEEE Communications Surveys & Tutorials, 25(4), 2951-2982.
[25] Li, Y., et al. (2025). Adaptive differential privacy in asynchronous federated learning for edge video analytics. Journal of Network and Computer Applications, 235, 104087.
[26] Tang, H., et al. (2024). EdgeFusion: Towards edge gradient enhancement in infrared and visible image fusion with multi-scale transform. IEEE Transactions on Computational Imaging.
[27] Wu, S., Xie, J., Zhang, Y., et al. (2025). ARM: Adaptive reasoning model. arXiv: 2505.20258.
[28] Pasandi, H.B., & Nadeem, T. (2020). CONVINCE: Collaborative cross-camera video analytics at the Edge. IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops).
[29] Gao, Y., & Zhang, B. (2023). Semantics-driven cloud-edge collaborative inference (arXiv: 2309.15435).
[30] Padala, V.A., & Nagothu, S.L. (2025). Federated learning with differential privacy for edge AI. University of Massachusetts Lowell.