Communication Technologies for Edge Learning and Inference: A Novel Framework, Open Issues, and Perspectives

[1*]Khan Muhammad, *Senior Member, IEEE*, [2,3]Javier Del Ser, *Senior Member, IEEE*, [4]Naercio Magaia, [5]Ramon Fonseca, [6]Tanveer Hussain, *Student Member, IEEE*, [7]Amir H. Gandomi, *Senior Member, IEEE, IEEE*, [8]Mahmoud Daneshmand, Senior Life Member, IEEE, [5]Victor Hugo C. de Albuquerque, *Senior Member, IEEE*

[1]Department of Applied AI, School of Convergence, College of Computing and Informatics, Sungkyunkwan University, South Korea
[2]TECNALIA, Basque Research & Technology Alliance (BRTA), Derio, Spain
[3]Department of Communications Engineering, University of the Basque Country (UPV/EHU), Bilbao, Spain
[4]School of Engineering and Informatics, University of Sussex, Brighton, UK
[5]University of Fortaleza, Fortaleza – CE, Brazil
[6]Institute for Transport Studies, University of Leeds, UK
[7]Faculty of Engineering & Information Technology, University of Technology Sydney, Australia
[8]Stevens Institute of Technology, Hoboken 07030, USA

Abstract

With the continuous advancement of smart devices and their demand for data, the complex computation that was previously exclusive to the cloud server is now moving towards the edge of the network. Due to numerous reasons (e.g., applications demanding low latencies and data privacy), data-based computation has been brought closer to the originating source, forging the Edge Computing paradigm. Together with Machine Learning, Edge Computing has turned into a powerful local decision-making tool, thus fostering the advent of Edge Learning. The latter, however, has become delay-sensitive as well as resource-thirsty in terms of hardware and networking. New methods have been developed to solve or, at least, minimize these issues, as proposed in this research. In this study, we first investigate representative communication methods for edge learning and inference (ELI), focusing on data compression, latency, and resource management. Next, we propose an ELI-based video data prioritization framework which only considers the data having events and hence significantly reduces the transmission and storage resources when implemented in surveillance networks. Furthermore, in this overview, we critically examine various communication aspects related to Edge Learning by analyzing their issues and highlighting their advantages and disadvantages. Finally, we discuss challenges and present issues that are yet to be overcome.

Keywords

Edge Learning, Communication Technologies, Video Summarization, Data Prioritization, Data Compression.

1. Introduction

The rapid growth of global data traffic is directly related to the accelerated popularization of edge devices. These typically low-powered embedded devices, which are often used for data collection, have led to unprecedented opportunities and innovative forms to improve our quality of life, serving as a stimulating substrate towards new scientific discoveries [1]. Indeed, combining Internet of Things (IoT) devices and data with the recent breakthroughs in machine learning (ML) has suggested academia and industry to pursue solutions in scenarios related to the smart city, intelligent transportation, e-Health, e-Banking, among others. Particularly, ML thrives in the domains of these applications [2]. Commonly, training procedures underneath ML models are computationally intensive, and thus only powerful cloud servers can support them effectively [3].

The limitation of ML to the cloud has propelled a recent research trend aimed at developing models to be trained at the edge devices of the network. However, the concept of distributed ML is designed by assuming a certain performance efficiency in relation to the computing and networking hardware. This is unlikely in practice due to the heterogeneous hardware and software of edge devices, which, along with the diversity of the data collected from different users contribute to significant differences between locally learned ML models. Consequently, the training of a model with satisfactory predictive performance and admissible computational efforts poses a challenging task that demands more intelligent solutions [4].

In this context, a recent trend has emerged from the evolution of mobile networks and the excellent capabilities of ML, called edge learning and inference (ELI). This concept is based on the idea that, instead of uploading all the data collected by the edge devices to a data center, the storage and computation resources of the edge network should be harnessed to provide a low-latency, reliable and intelligent learning service [5]. Furthermore, Federated Learning (FL) has gained momentum in the last years for privacy-preserving distributed ML, which is realized by collaboratively training a model across devices without sharing their data. This concept combined with the idea of ELI is called Federated Edge Learning and Inference (FELI). The main feature of FELI is to aggregate local models trained on devices to update the global model at a server.

Referring to communication aspects, an intelligent edge should provide adaptive and dynamic management and maintenance of communication resources at the edge. The progressive development of communication technologies has enabled increasingly diverse network access methods. At the same time, a more persistent and reliable connection between edge devices and the cloud server is provided by the edge computing infrastructure as an intermediate medium [6]. Thus, a continuum of shared resources is formed by the gradually merged end devices, edge, and the cloud. However, the complex and sizeable overall architecture of computing, wireless communication, storage, networking, and other resources can be a challenge to maintain and manage [2].

This overview brings together the above concepts by elaborating extensively on the problems derived from the deployment of ELI functionalities. Specifically, we focus on the video data generated by edge devices, which call for the deployment of effective Artificial Intelligence models that endow such edge devices with human-like intelligence to respond to real-time events. This survey highlights the potential of ELI and overviews state-of-the-art strategies for communication technologies, emphasizing their achievements and limitations. To shed empirical evidence on our analysis, a second contribution of this work is a novel deep learning-based framework that prioritizes video data containing human actions over the edge and then transmits it upstream for detailed analysis. Considering the restricted storage and processing capabilities of edge devices, we resort to a lightweight deep learning model to recognize coarse actions in vision sensor data. The present survey concludes by reviewing issues in this research domain and by outlining future research directions.

2. Edge Learning and Inference

A. Concept

The ELI concept comes from the aggregation of two main concepts: Edge Computing (EC) and ML. EC aims to bring data processing as close to the source as possible, reducing end-to-end latency and bandwidth usage. In other words, ELI runs learning and/or inference algorithms locally (i.e., at an IoT device, a user's computer, or an edge server), thereby demanding less network and cloud resources. Bringing computation close to the network edge minimizes the number of long-distance communications and reduces potential sources of risk regarding data privacy. On the other hand, ML tries to "learn" by using data, statistics, and trial-and-error to optimize a process to endow the machine with a robust decision-making ability. For instance, ML gives computers the capabilities to solve that even humans struggle to accomplish, such as climate change and cancer research [7]. The combination of these two definitions is what makes ELI so promising. It capitalizes on the agility, flexibility, and privacy capabilities of EC while harnessing the cognition in decision-making of ML. In other terms, the basic concept behind ELI is the idea of distributing learning and inference across an entire network instead of centralizing it in the cloud, as shown in Figure 1.

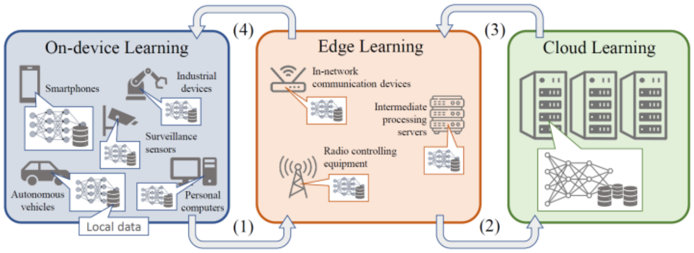

Figure 1: Representation of the ELI and FELI concepts, where the distribution of tasks is located across the entire network. Modules: (1) Download model parameters; (2) Update model with local data; (3) Upload new parameters; (4) Update global model.

B. Applications

ML applications at the edge flourish in almost all areas of computing. ELI is usually associated with security and surveillance, but it goes far beyond these application domains. Augmented Reality, Virtual Reality, or automatic pilot require real-time video analysis. The execution of these tasks in a cloud environment incurs some complications, such as inadmissible latency levels, exhausted bandwidth, and low reliability. All these complications can usually be resolved when video processing is implemented closer to the data origin at either an end device or an edge node [1]. Complementarily to the applications mentioned above, the Internet of Vehicles can also benefit significantly from ELI to reduce accidents, decrease traffic congestion, improve safety, and enhance efficiency in transportation systems. On one hand, EC can provide high-speed, low-latency communications for a fast response in inference time, and more plausible realization of autonomous driving. On the other hand, deep learning techniques have become central for vehicular perception and, hence, act as an enabler for safe autonomous driving [8].

IoT has a key role to bring intelligence to homes and offices, ranging from smart lighting control systems to smart access control. Nevertheless, it is necessary to make use of wireless IoT controllers and sensors in walls, floors, and corners. In order to protect sensitive data, IoT also relies on ELI extensively [9]. The same is true in the context of cities. Areas such as public facilities, transportation, and public safety can significantly benefit from ELI [10]. Cost and efficiency are the two primary considerations here, for the size and magnitude of a big metropolis, especially considering the natural and geographical characteristics of current urban environments. The only viable solution in the mid-to-long run is to adopt the ELI paradigm. Transitioning from personal to industrial IoT, two principles become critical. The first is automation, which is the main target of ML technology. The second is data analysis, whose output is essential for informed decision-making. Some requirements are vital in these environments, such as response latency, privacy protection, and risk control. As we will review next, the ELI-related contributions reported to date have addressed these points significantly.

3. Communication Methods for ELI

A. Data Compression

For a fast ML model training (*learning*), it is necessary to enable rapid access to the enormous real-time data captured by edge devices. However, there is a bottleneck in the communication over networks due to the extensive overhead. One approach to reduce the overhead is to compress data through stochastic gradients. To further alleviate the compression bandwidth demand, most existing schemes focus on scalar gradient quantization to efficiently deal with its high dimensionality. With this challenge in mind, Y. Du et al. [3] presented a low-complexity Grassmannian quantization scheme that, besides communication-efficient, guarantees convergence. This method aims to minimize the deviation in direction between a line and its quantized version, or the deviation in orientation for the case of a subspace, becoming particularly useful when compressing data that contain information on a subspace orientation or vector direction.

Specific data compression approaches have also been reported in the FL realm, especially around the Federated Average (FedAvg) algorithm, aimed to reduce the data communication requirements of learning edge nodes and support data privacy. FedAvg is designed for user devices, such as smartphones, which also suffer from the same data compression problems. To reduce and help ease network usage, J. Mills et al. developed a communication-Efficient FedAvg approach to reduce the number of rounds towards FedAvg convergence and the total/compressed data uploaded per round [1]. A summarized form of literature is given in Table I, representing data compression, latency, resource management, and adaptive transmission approaches.

B. Latency

Other properties can be targeted to achieve efficient communication in both learning and inference, such as the reduction of latency. In [4], a broadband analog aggregation technique was used for FELI in a wireless network. The scheme's idea is to exploit the simultaneous transmission of a broadband multi-access channel waveform superposition property. The reduction of the latency when compared to other orthogonal access is significant, but there is still room for improvement, such as to enhance the aggregation. D. Zhang et al. developed a solution to batch tasks, called EdgeBatch, to identify the optimal batch size for Deep Neural Network (DNN) of GPU-intensive tasks, leading to a significant optimization between the delay of tasks in end-to-end devices [10]. M. Elbamby et al. analyzed techniques of ultra-reliable low-latency communication enablers that consist of a more direct attack on the leading cause of the latency, that is, the distance to the server [8].

C. Resource Management

There is also a tradeoff between communication and computation at the edge inference system in the downlink, especially when the nodes perform model learning tasks. A higher quality-of-service can be attained by performing inference at the edge for delivering the output results to mobile users through a cooperative downlink transmission. Building upon this tradeoff, K. Yang et al. proposed to jointly decide on the task allocation strategy at edge nodes and designed downlink beamforming vectors by minimizing the sum of transmission and computation power consumption [11]. Y. Liu et al. introduced an IoT-based energy management system that deploys edge computing with Deep Reinforcement Learning [9]. This system can improve energy management performance as well as reduce the execution time. S. Yu et al. proposed a framework capable of taking precise offloading decisions in the MEC network. In the framework, both the variance of the network conditions and the execution cost of the task on the MEC side are analyzed [12].

D. Adaptive Transmission

The edge of the network in smart IoT devices and surveillance cameras has the task of capturing and compressing video information, processing frames in real-time, and depending on the application itself, performing image segmentation, annotation and/or captioning, among other learning-based functionalities alike. To reduce the potential overhead caused by the data size, one can adaptively acquire data and/or selectively send it to the cloud. In this context, Wang et al. introduced various Edge DL methods to adapt to various Deep Neural Networks' architectures, hardware platforms, wireless connections, and server load levels, to identify the partition point for best latency and best mobile energy consumption [2].

Table I: Data compression, latency, resource management, and adaptive transmission approaches with their key properties.

Issue	Citation	Solution	Advantages	Disadvantages
Data compression	[3]	Quantization of low complexity of Grassmannian	Light-computation requirement and highly scalable	Network requirements seem to be limiting
	[1]	Federated Average (FedAvg)	Greatly reduces the number of rounds of convergence, reducing data sent	Does not reflect the real environment
Latency	[4]	Broadband Analog Aggregation (BAA)	The reduction of the latency compared to other orthogonal access is significant	Although the latency is reduced, the complexity is increased
	[10]	EdgeBatch	Significant optimization between the delay of tasks in end-to-end devices	Do not reflect the real environment
	[8]	Ultra-Reliable, Low-Latency Communications	A great number of enablers in ELI	Lack of any test with the enablers mentioned
Resource management	[11]	Design Downlink Beamforming Vectors	Minimizes the sum of transmission power consumption and computation power consumption	Restrictive assumptions in the proposed technique
	[9]	Deep Reinforcement Learning	Significantly reduces scheduling time.	It is computational demanding
	[12]	Framework Capable Offloading Decisions	Greatly optimizes the communication resource and energy saving	It is not clear if it would run in other types of environments
Adaptive transmission	[2]	Adaptive framework for reducing the delay and power consumption	Highly adaptive method that analyzes the various device conditions in one DNN method	Can be outperformed by a specialized method

4. Proposed ELI-based Video Data Prioritization Framework

In this section, we build upon the previously reviewed state-of-the-art strategies by focusing on video surveillance in IoT environments with learning functionalities deployed in EC nodes. Indeed, the distributed vision sensors in IoT networks are functional 24/7, continuously creating big volumes of data, that are used in many surveillance applications, ranging from indoor office monitoring to outdoor public places and roads. Following the issues of ELI scenarios discussed previously, automated analysis and purposeful utilization of vision sensors' data are hindered by: (1) the high latency rate of wireless sensor networks (WSNs) that create transmission delays and reduce the swiftness of responsive actions in IoT environments; and (2) the large-sized complex image data that require powerful graphics processing units (GPUs) and cloud handling services. Instant analysis of video data results in efficient responsive engagements, thereby preventing or delaying anomalous events. In our proposed edge intelligence-based video data prioritization framework, these challenges are tackled intelligently by considering resource-constrained devices that are functional over the edge nodes, thereby showcasing the enormous potential of ELI for this application niche.

The proposed framework has several functional modules, such as input acquisition from resource-constrained device over the edge, equipped with vision sensor, followed by its propagation to the lightweight trained model and optimally to the prioritization decision-making module. Once the data is selected for prioritization, they are forwarded to the data analysis cloud servers over WSNs in a compressed format and then used for detailed events analysis, including video classification, anomaly analysis, video retrieval, etc. The resource-constrained device deployed in IoT environment is also functional 24/7, generating video frames. However, our framework ensures transmission of data containing events, which is possible due to our fine-tuned deep learning model for events classification. The deep learning model is trained over images from video summarization and anomaly recognition datasets comprising a diverse set of events, including walking, running, or jogging, among other actions alike. The baseline concept of our framework is to utilize the limited computational capabilities effectively; therefore, we skip the details of events, such as their types (walking, running), and integrate them together in a single class, named "event," against the videos without any event, termed "no-event." Furthermore, as the video data are 3-dimensional, 3-D learning models are required for effectively accounting for correlations in time and image domains. The enormous parametric space of 3D learning models requires training and implementation over GPUs and our motive is to execute lightweight models over resource-constrained devices. To tackle this issue, the proposed framework incorporates the "time intervals" concept over video frames to enhance the decision-making of "event" and "no-event" classes for a series of continuous frames belonging to the same event. A recent state-of-the-art lightweight deep learning model, "EfficientNetB3" [13], was utilized for our problem. The learning abilities of "EfficientNetB3" over "ImageNet" dataset are transformed into the binary classification task of "events/no-events" probabilities computation in continuous video frames.

The working mechanism of the proposed data prioritization framework is four-fold: i) acquire a single frame, ii) propagate it to the trained model to generate output probabilities, iii) analyze the probabilities in intervals then decide whether frames of an interval should be prioritized, and iv) send over WSN to the data analysis centers. The proposed framework employs a vision sensor attached to the resource-constrained device that continuously generates video frames. The key module of our framework is related to the fine-tuned deep learning models prediction, where we pass each frame in real-time from the vision sensor to acquire an output probabilities array, indicating the chances of each class's occurrence. In case the probability for the "event" class is higher than a certain threshold, the frame is stored temporarily alongside its probability value. The same process continues until the structural similarity index measurement (SSIM) distance between two successive frames is lower than a certain threshold, indicating a different "event". The SSIM measurement helps distinguish one event from another; therefore, the proposed framework transmits the "prioritized" frames of a single event over the WSN.

Saving Network Resources:

To ensure an extensively efficient and flexible transmission over the WSN, our framework implements an encoding mechanism over the frames of a single event interval before its transmission. The encoding mechanism involves compression of portable network graphics of each frame in the prioritized interval, which decreases the data size (approximately 17.02% smaller size per frame), increasing the speed and safety of transmission as the frames over the WSN are propagated in an intermediate structure i.e., encoded format. To achieve extensively reduced transmission, it is also possible to transmit only a single encoded frame from each interval, as given in the experiments (see Table II). Most of the time, an event occurs in sequence of frames, and thus the frame with highest probability of ongoing event is representative of the whole event (current interval), which means dependency on this specific frame is also a reliable decision. Therefore, based on our experimental results, we suggest applying the concept of single frames selection from an interval, while transmitting prioritized contents over communication channels with huge traffic or limited potential. The overall working pipeline is given in Figure 2.

To verify the effective prioritization potentials of our framework, we examined its performance over a multi-view surveillance videos summarization dataset "Office dataset". The generated prioritized contents and other statistics are given in Table II. We applied our algorithm over the whole dataset, which contains four different videos, and reported the prioritized data results against the actual contents' transmission. As shown in Table II, the Office-0 video has a total number of 26955 frames, each of 20 KB size (average), resulting in 539.1 MB for the overall frames. If transmitted over the WSN, this video demands a huge bandwidth and yields unaffordable transmission latencies. On the other hand, if we apply the proposed algorithm, the prioritized content containing events shrinks the number of frames to 776 and 15.52 MB, thereby relaxing the usage of transmission resources.

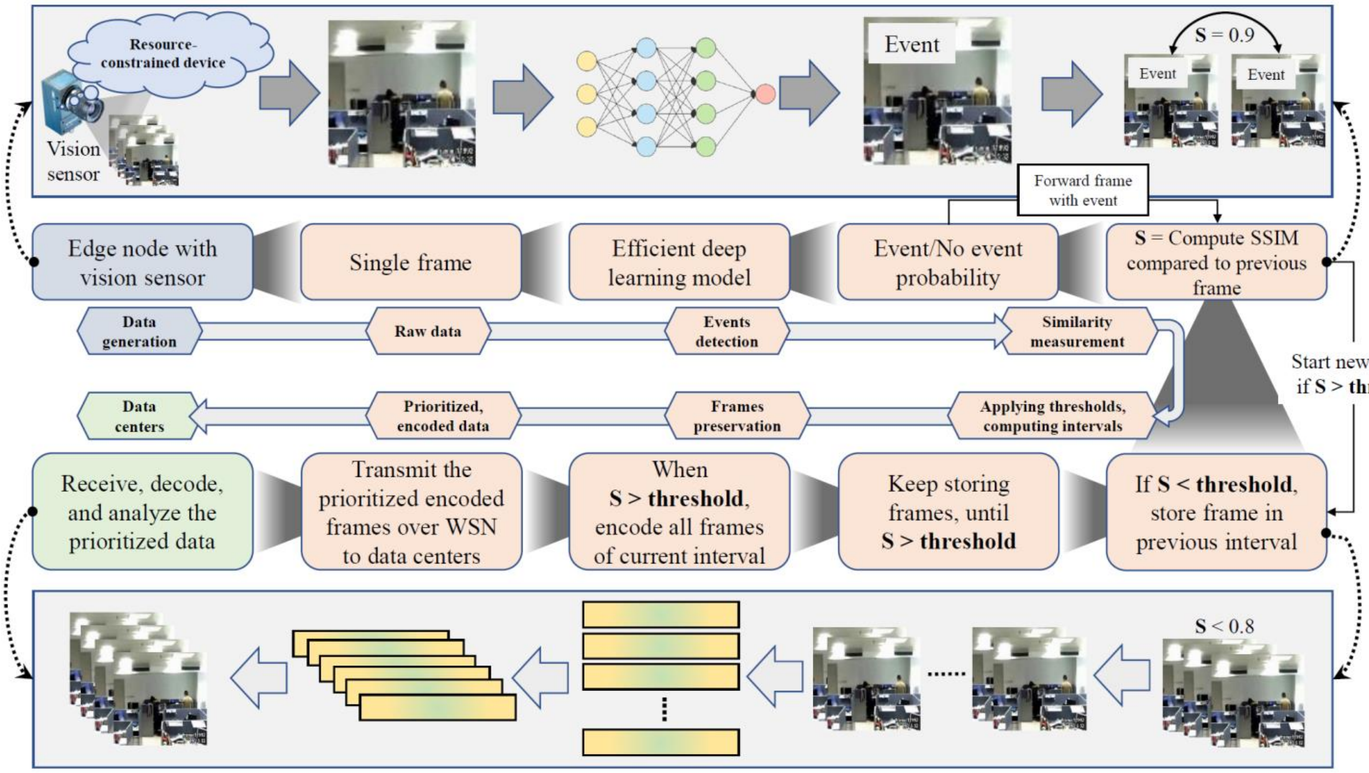
Resource-constrained device
Vision sensor
Event
S = 0.9
Event
Event
Forward frame with event
Edge node with vision sensor
Single frame
Efficient deep learning model
Event/No event probability
S = Compute SSIM compared to previous frame
Data generation
Raw data
Events detection
Similarity measurement
Start new interval if S > threshold
Data centers
Prioritized, encoded data
Frames preservation
Applying thresholds, computing intervals
Receive, decode, and analyze the prioritized data
Transmit the prioritized encoded frames over WSN to data centers
When S > threshold, encode all frames of current interval
Keep storing frames, until S > threshold
If S < threshold, store frame in previous interval
S < 0.8
......

Figure 2: The proposed edge-intelligence based data prioritization framework for surveillance in IoT environments.

After applying the intervals concept, i.e., selecting only a single frame with the highest probability from a single event, the number of frames for the Office-0 video becomes only 52 with 1.04 MB size of transmission. Note that in our experiment, we considered a sequence of 15 frames per event, so as not to lose any important information related to the frames with events. The significantly lowest transmission size is observable for the frames when sent after encoding, i.e., 0.104 MB, while utilizing the intervals concept. Thus, a huge transmission gap is observable for a video of only 9- to 14-minute duration and is elevated exponentially if applied to a whole-day surveillance video and even more for week- and month-long videos. Despite the high-level of transmission relaxation, the resulting frames with events are effective and useful for future analysis, such as video retrieval, detailed events' investigation, etc. For detailed information about the prioritization concepts when applied to distributed sensors, readers are referred to our recent research, where inter-view correlations among different views are also considered while generating the final summary, but their mechanism is not deployable over resource-constrained devices. A detailed transmission latencies evaluation over the Office dataset is provided in Figure 3, where a huge latency difference (in seconds) is observable for prioritized frames using intervals concept. The latency values are calculated using the generic formula that sums up propagation, transmission, and queuing time for a packet transmission. However, since we consider the ideal transmission situation, the queue delay time is ignored.

Table II: Experimental results of the proposed framework over Office multi-view dataset videos. The total number of frames and their corresponding size (in MBs) after applying the proposed prioritization framework and encoding scheme are marginally decreased.

Video #	**Total Frames**			**Overall prioritized data transmission**			**Single frame selection from an interval of 15 frames**		
	# Frames	Size (MB)		# Frames	Size (MB)		# Frames	Size (MB)	
		Normal	Encoded		Normal	Encoded		Normal	Encoded
Office-0	26955	539.1	53.91	776	15.5	1.55	52	1.04	0.104
Office-1	16329	318.4	32.65	175	3.4	0.006	12	0.2	0.024
Office-2	20486	307.2	40.97	12641	189.6	25.28	843	12.6	1.686
Office-3	26948	538.9	53.89	14000	14	28.00	940	0.9	1.880

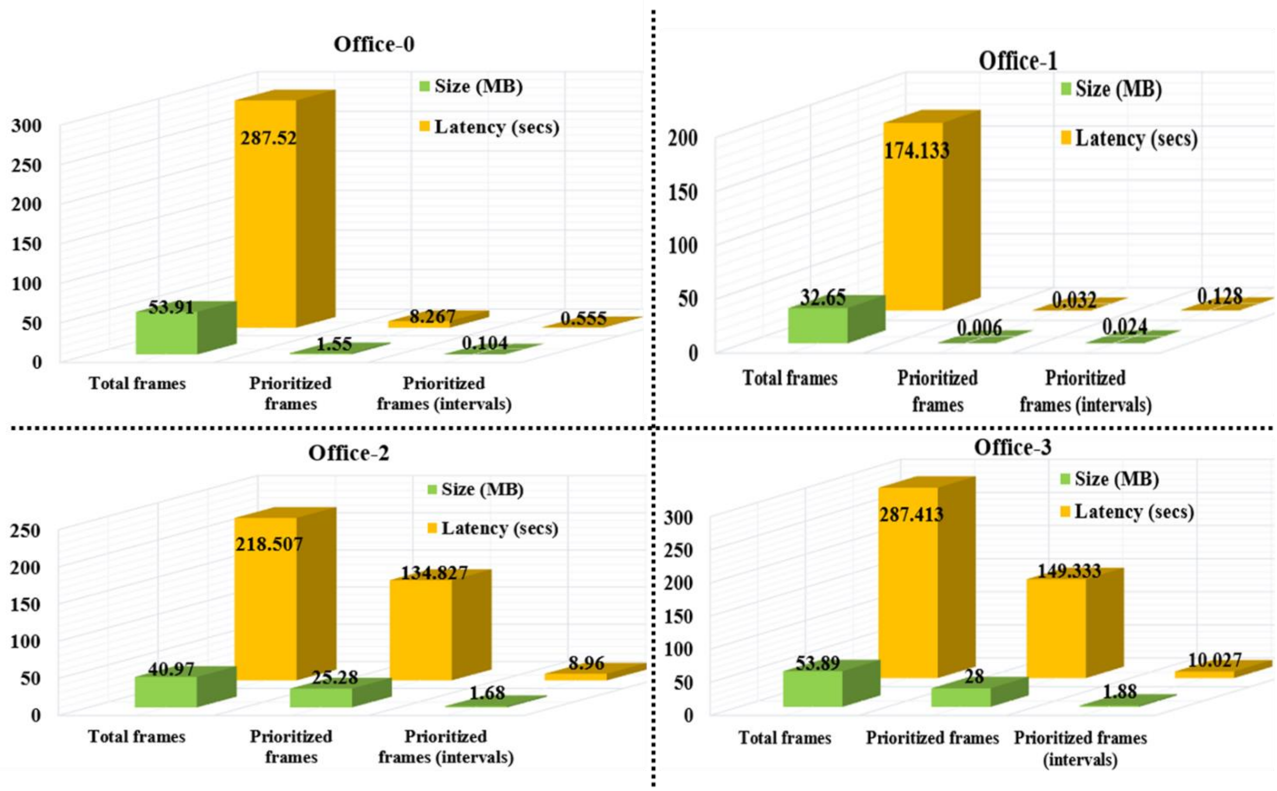

Figure 3: Performance evaluation of the proposed prioritization framework latency rates over Office dataset videos.

The propagation time is computed by dividing the distance (500 m in our situation) over speed of light in optical fiber (2×10^8 m/s m/s) and the output value is negligible, i.e., 25×10^{-7}. Transmission time is computed by dividing the data size over bandwidth (considered constant at 1.5 Mbps), which plays a key role in the overall latency rate. The data size is directly proportional to the transmission time, indicating that a big data packet (total frames) poses higher latency compared to the lower-sized data packet (prioritized frames) transmission. Table II presents the experimental results of the proposed framework over the Office multi-view dataset videos. It can be seen that the total number of frames and their corresponding size (in MBs) marginally decreased after applying the proposed data prioritization framework and encoding scheme.

5. Open Challenges and Research Directions

The methods mentioned above still require further improvement to ensure efficient communication. To fully leverage the benefits of ELI, it is essential to establish a connection with a cloud server, divide the heavy computation model of DL into sub-tasks, and then, with an effective method, spread these tasks between the edge devices collaboration. There are various critical features to take into consideration for communication optimization, cache resources, heterogeneous data, high-dimensional parameters, and real-time joint optimization. Specifically, most communication methods struggle to sustain real-time processing, and to pipeline tasks for an optimized connection. *Further reducing the communication overhead* still largely remains an open issue, particularly in models comprising millions of trainable parameters.

Benefiting ELI via Cloud Services. Many existing techniques, such as recently proposed by Hussain et al. [14] for video summarization employ features extraction, activities recognition at the edge devices. Although this approach has the benefit of instant decision making but it also comes with certain limitations such as the computational cost at edge devices consume resources at higher ratio. A more sophisticated approach is to analyze the data in a detailed manner at cloud analysis centers to save the edge resources. A strong communication between the edge source and cloud ensures instant decision as well as distributing the workload among the adjacent edge devices for efficient processing.

Federated ELI. Mainstream existing techniques for edge devices are focused on centralized data acquisition and training. In ELI, federated learning can play a significant role that needs to be explored. For instance, ELI can be effectively used in intelligent transportation, surveillance systems, and smart cities to acquire data continuously at any spot, that can be the main cloud server. While the model at the edge device can train and update the model's parameters using federated learning concepts without acquiring the data locally at the specific machine. Therefore, it can extend the real-world scenarios and diversity of the existing data at any edge device.

ELI combines the best of EC with ML to solve problems that cloud servers currently face (high latency, reliability, privacy); yet this also poses new challenges that must be addressed (e.g., *battery* *consumption, real-time processing, distributed/scalable ML pipelines over the network*). Some methods have been proposed to diminish these problems, but they should be deployed together. For instance, data compression, latency reduction, resource management, and adaptive transmission are some approaches that can help to mitigate these issues. For the future, we firmly advocate for more consistent and principled studies concentrating on these problems to acquire the rigor and empirical evidence needed to make ELI a driver of the next technological decade.

6. Conclusions

The arrival, development, and massive adoption of ELI can surely buttress hundreds of new applications soon. Functionalities, as showcased in this review for video data, are promising, yet require further sophistication to achieve a sufficient level of maturity for practical use. In this survey, we investigated various communication methods for ELI, focusing on data compression, latency, and resource management. Towards saving communication resources and better management, we presented an ELI-based video data prioritization framework which only considers the data having events. Our framework's ability to save communication resources is experimentally proved. We also listed the major limitations, challenges, and directions for further research in this domain.

Acknowledgments

This work was supported by the National Research Foundation of Korea grant funded by the Korea government (No. 2019R1A2B5B01070067) and in part by the Basque Government through the ELKARTEK program (ref. KK-2022/00119) and the consolidated research group MATHMODE (ref. IT1456-22).

References

[1] J. Mills, J. Hu, and G. Min, "Communication-efficient federated learning for wireless edge intelligence in IoT," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 5986-5994, 2019.

[2] X. Wang, Y. Han, V. C. Leung, D. Niyato, X. Yan, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 869-904, 2020.

[3] Y. Du, S. Yang, and K. Huang, "High-dimensional stochastic gradient quantization for communication-efficient edge learning," *IEEE transactions on signal processing*, vol. 68, pp. 2128-2142, 2020.

[4] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 491-506, 2019.

[5] M. Dai, Z. Su, R. Li, Y. Wang, J. Ni, and D. Fang, "An edge-driven security framework for intelligent internet of things," *IEEE Network*, vol. 34, no. 5, pp. 39-45, 2020.

[6] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Toward an intelligent edge: Wireless communication meets machine learning," *IEEE communications magazine*, vol. 58, no. 1, pp. 19-25, 2020.

[7] N. Skatchkovsky and O. Simeone, "Optimizing pipelined computation and communication for latency-constrained edge learning," *IEEE Communications Letters*, vol. 23, no. 9, pp. 1542-1546, 2019.

[8] M. S. Elbamby *et al.*, "Wireless edge computing with latency and reliability guarantees," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1717-1737, 2019.

[9] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang, "Intelligent edge computing for IoT-based energy management in smart cities," *IEEE network*, vol. 33, no. 2, pp. 111-117, 2019.

[10] D. Zhang, N. Vance, Y. Zhang, M. T. Rashid, and D. Wang, "EdgeBatch: Towards AI-Empowered Optimal Task Batching in Intelligent Edge Systems," in *2019 IEEE Real-Time Systems Symposium (RTSS)*, 2019: IEEE, pp. 366-379.

[11] K. Yang, Y. Shi, W. Yu, and Z. Ding, "Energy-efficient processing and robust wireless cooperative transmission for edge inference," *IEEE internet of things journal*, vol. 7, no. 10, pp. 9456-9470, 2020.

[12] S. Yu, X. Chen, L. Yang, D. Wu, M. Bennis, and J. Zhang, "Intelligent edge: Leveraging deep imitation learning for mobile edge computation offloading," *IEEE Wireless Communications*, vol. 27, no. 1, pp. 92-99, 2020.

[13] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*, 2019: PMLR, pp. 6105-6114.

[14] T. Hussain *et al.*, "Multiview summarization and activity recognition meet edge computing in IoT environments," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9634-9644, 2020.

Khan Muhammad [S'16, M'18, SM'22] is an assistant professor at the Department of Applied AI, Sungkyunkwan University, South Korea. His research interests include video summarization, computer vision, big data analytics, IoT, and intelligent surveillance. He has published over 220 papers in peer reviewed international journals and conferences in these research areas. He is Associate Editor/Editorial Board member of over 14 journals and a highly cited researcher for 2021 (as per Clarivate).

Naercio Magaia is a Lecturer in the School of Engineering and Informatics of the University of Sussex, UK. His current research interests include Computer Networks, Cybersecurity, IoT, and Artificial Intelligence.

Ramon Fonseca is a graduate student in Computer Engineering at the University of Fortaleza (UNIFOR). His research interests include Computer Vision, Machine Learning, and Robotics.

Tanveer Hussain [S'16] is a Post Doc Fellow at Univeristy of Leeds, UK. His research interests include video analytics, embedded vision, edge learning, video summarization, and IoT.

Amir H. Gandomi is a Professor of Data Science and an ARC DECRA Fellow at the Faculty of Engineering & Information Technology, University of Technology Sydney, Australia. He has published over 200 journal papers and seven books which collectively have been cited 33700+ times (H-index=83). He has been named as one of the most influential scientific mind and Highly Cited Researcher (top 1% publications and 0.1% researchers) for five consecutive years, 2017 to 2021. His research interests are global optimization and (big) data analytics using machine learning and evolutionary computations.

Javier Del Ser [M'07, SM'12] is a Research Professor at TECNALIA, and a professor at the Department of Communications Engineering of the University of the Basque Country (UPV/EHU). His research interests include all forms of Artificial Intelligence, with a focus on DL applications, explainability techniques (XAI), meta-heuristic optimization and randomization-based modeling.

Mahmoud Daneshmand (Senior Life Member, IEEE) received the Ph.D. degree in statistics from the University of California at Berkeley, Berkeley, CA, USA. He is a Professor with the Department of Computer Science, Stevens Institute of Technology, Hoboken, NJ, USA. He has over 40 years of industry and university experience as a Professor, a Researcher, an Assistant Chief Scientist, the Executive Director, a Distinguished Member of Technical Staff, a Technology Leader, the Chairman of Department, and the Dean of School with Bell Laboratories, Murray Hill, NY, USA; AT&T Shannon Labs—Research, Florham Park, NJ, USA; the University of California at Berkeley; the University of Texas at Austin, Austin, TX, USA; the Sharif University of Technology, Tehran; the University of Tehran; New York University, New York, NY, USA; and the Stevens Institute of Technology.

Victor Hugo C. de Albuquerque [M'17, SM'19] is a full professor at the University of Fortaleza, Brazil. He has experience in the research fields of applied computing, intelligent systems, visualization and interaction, with specific interest in artificial intelligence, Internet of Things, and Internet of Health Things. He is Associate Editor of several reputed journals.