# Model Collaboration at Network Edge: Feature-Large Models for Real-Time IoT Communications

Xinbo Yu, Shuhang Zhang, *Member, IEEE*, Hongliang Zhang, *Member, IEEE*, and Lingyang Song, *Fellow, IEEE*

***Abstract*—The growth of the Internet of Things (IoT) has reshaped the way devices, systems, and applications connect, leading to an enormous surge in data generation across various domains. This expansion, paired with the exponential increase in IoT devices, requires advanced data analysis capabilities to manage the multimodal sensory data collected by the massive IoT devices in real time, such as sensor outputs, visual data, audios, and videos. To address this challenge, large generative artificial intelligent (AI) models are designed, showing promise in processing multimodal data. However, deploying these models on IoT devices is constrained by limited computational power, memory, and energy resources, preventing full realization of their potential for real-time IoT systems. To address these limitations, we propose an innovative end-edge collaborative model framework between end nodes and edge servers, designed to balance computational load and optimize resource use. This approach transmits both extracted features and residual mapping data from end nodes to edge servers, allowing for spectrum efficient data handling across the network. Our work formulates an optimization strategy to enhance mean average precision (mAP) by adjusting task distribution, bandwidth, and data quantization in response to real-time network and device conditions. Comprehensive simulations demonstrate the proposed approach's superiority over conventional centralized edge model computing and distributed end model computing frameworks, achieving enhanced efficiency across various communication rates in real time.**

***Index Terms*—Internet of Things (IoT) networks, model collaboration, resource, task allocation.**

## I. Introduction

THE rapid development of the Internet of Things (IoT) has revolutionized the interaction among devices, systems, and applications, leading to an unprecedented expansion of IoT services and a substantial increase in multimodal data generation [1]. As we look toward the future, the number of IoT devices is expected to grow exponentially, encompassing everything from sensors and cameras to smart appliances [2]. This growth will result in vast amounts of heterogeneous data, including sensor readings, images, audios, and videos [3]. The heterogeneous data needs to be processed and analyzed in high accuracy to support advanced applications like smart cities [4], industrial automation [5], autonomous vehicles [6], and healthcare [7], which brings significant challenges in real-time data processing and analysis for IoT devices.

To address the complexity of processing such multimodal data, large generative artificial intelligent (AI) models have emerged as powerful tools capable of handling intricate tasks across different data modalities [8], [9]. Advanced large generative AI models like Llama-3 [10] and Gemini [11], which contain billions of parameters, are designed to interpret and analyze multimodal data simultaneously. By deploying these generative AI models in IoT systems, the multimodal sensory data can be processed and analyzed with higher accuracy [12]. Therefore, these models are essential for enabling next-generation IoT applications [13], [14], such as agriculture automation [15] and smart transportation [16], [17], [18], which require sophisticated data processing capabilities.

However, deploying these large-scale AI models directly on IoT devices is hindered by inherent limitations, such as computational power, limited memory, and energy constraints [19]. Due to the above limitations, most IoT devices can only support models with millions of parameters, like Yolov8 [20], restricting them to less complex inference tasks. This prevents IoT systems from fully leveraging the capabilities of large AI models for future applications, and processing the sensory data with high accuracy in real time. To overcome this challenge, end-edge collaboration for IoT systems has emerged as a viable solution [21]. In end-edge collaboration frameworks, IoT devices serve as end nodes, prioritizing low-latency and

Received 8 November 2024; revised 12 December 2024; accepted 26 December 2024. Date of publication 30 December 2024; date of current version 9 May 2025. This work was supported in part by the National Key Research and Development Project of China under Grant 2022YFB2902405; in part by the National Science Foundation under Grant 62371011 and Grant 62401302; in part by the Beijing Natural Science Foundation under Grant L243002; in part by the GuangDong Basic and Applied Basic Research Foundation under Grant 2023B0303000019; and in part by the Research Project of the Pengcheng Laboratory under Grant PCL2024Y02. *(Corresponding authors: Hongliang Zhang; Lingyang Song.)*

Xinbo Yu is with the School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: yxb2005@stu.pku.edu.cn).

Shuhang Zhang is with Pengcheng National Laboratory, Shenzhen 518055, China (e-mail: zhangshh01@pcl.ac.cn).

Hongliang Zhang is with the School of Electronics, Peking University, Beijing 100871, China (e-mail: hongliang.zhang@pku.edu.cn).

Lingyang Song is with the School of Electronics, Peking University, Beijing 100871, China, and also with the School of Electronic and Computer Engineering, Peking University Shenzhen Graduate School, Shenzhen 518055, China (e-mail: lingyang.song@pku.edu.cn).

Digital Object Identifier 10.1109/JIOT.2024.3523953

low-complexity data preprocessing. Meanwhile, edge servers with greater computational power handle the data received from IoT devices, employing large AI models to achieve high-accuracy data processing. Therefore, IoT systems perform real-time data processing in high accuracy through end-edge collaboration.

In this article, we propose an integrated end-edge collaborative model framework in which end nodes transmit extracted features and selective sensory data to the edge servers. The end nodes and edge servers process the data collaboratively using AI models, maximizing the accuracy in data processing. The proposed framework incorporates a feature stream transmitting key features of the data, and a data stream transmitting residual mapping data, thereby optimizing both computational and communication resources. We aim to maximize the mean average precision (mAP) of the proposed framework by optimizing the task allocation ratio, bandwidth allocation, and residual mapping data quantization. In this way, the end-edge collaboration allows end nodes and edge servers to adjust their transmission throughputs and distributions of computational power flexibly, thus delivering consistently high-accuracy data processing in real time.

Recently, many related works have explored the applications of distributed model collaboration in IoT networks. In [22], Sun et al. proposed a joint offloading scheme based on resource prediction to improve the efficiency of cloud–edge collaboration in industrial IoT. Based on deep reinforcement learning, Xiong et al. [23] proposed a resource allocation strategy for the IoT edge computing system to improve the efficiency of resource utilization. In [24], Han et al. discussed the challenges of using both resource-strenuous edge devices and elastic cloud resources in edge–cloud jobs, and proposed a method of fast scheduling algorithm tuning for dynamic edge–cloud workloads and resources. In [25], Yan et al. proposed a deeper multiscale encoding–decoding feature fusion network to handle remote sensing change detection tasks. However, existing works [22], [23], [24] primarily rely on raw data exchange among cooperative nodes, demanding high transmission throughput, which poses significant challenges in future 6G networks with massive IoT devices. Meanwhile, although [25] considered extracted feature transmission, it leads to poor performance for complex inference tasks with huge data. By contrast, our framework enables dynamic adjustment of feature transmission and data transmission between end nodes and edge servers, enabling massive IoT devices as end nodes to reach the optimal inference performance in various communication conditions, maintaining affordable bandwidth usage.

Therefore, the contributions of this article are summarized as follows.

1) We present an innovative end-edge collaborative model framework that empowers both end nodes and edge servers to jointly engage in real-time model computation. This framework seamlessly integrates streams of feature and data transmissions between end nodes and edge servers, enabling optimizations of computational and communication resources through cohesive collaboration.
2) We formulate an optimization problem aimed at maximizing the mAP of the proposed framework with multiple end nodes. To solve this problem, we propose a dynamic task and communication allocation algorithm based on real-time network conditions and IoT device requirements.
3) We demonstrate, through extensive simulations, that the proposed framework outperforms the centralized edge server model computing framework and the distributed end node model computing framework in terms of accuracy, efficiency, and resource utilization across various communication rates in real time.

This article is organized as follows. Section II introduces the system model of our integrated end-edge model framework. Section III formulates the mAP optimization problem and discusses its decomposition into subproblems. Section IV presents the proposed solution approach, while Section V provides the detailed simulation results and performance evaluations. Finally, Section VI concludes this article.

## II. System Model

### A. Integrated End-Edge Model Framework

In this section, we introduce the integrated end-edge model framework with feature and data streams, which is illustrated in Fig. 1. Our framework consists of end nodes (e.g., cameras) and edge nodes (e.g., edge servers). For clear illustration, we present two cameras as end nodes and one server as an edge node. The end nodes are able to capture image frames and do analysis like feature extraction and inference, and the edge server functions as a central processor and data storage. To be specific, each end node first captures sensory data, e.g., images, with onboard camera, and then extracts features of sensory data with onboard model. Finally, it transmits the extracted features of sensory data to the edge server by over-the-air (OTA) transmission, defined as the *feature stream*. Moreover, in order to get further improvement of the inference performance at the edge server, extra data beyond the extracted feature, which contains detailed visual information, known as the residual mapping data, can be transmitted to the edge server by idle OTA transmission resources with adjustable resolution, referred to as *data stream*.

Recently, many related works have given strong support to the implementation of the above two streams in our framework [26]. For feature stream, the compact feature representation technique guarantees the high-efficiency feature extraction and data compression, reducing the cost of feature stream to a few Kb/s [27]. For residual mapping data in the data stream, there is an intelligent coding technique facilitating the efficient representation of image, making the dynamic encoding of the video stream into a practical level [28].

Considering the intrinsic instability of the wireless communication bandwidth, we need to design a flexible communication paradigm to maximize the performance of the whole system. The paradigm includes two main parts: 1) allocating different bandwidth to each end node for its feature transmission due to the real need of end node and channel condition, called *dynamic task allocation* and 2) uploading

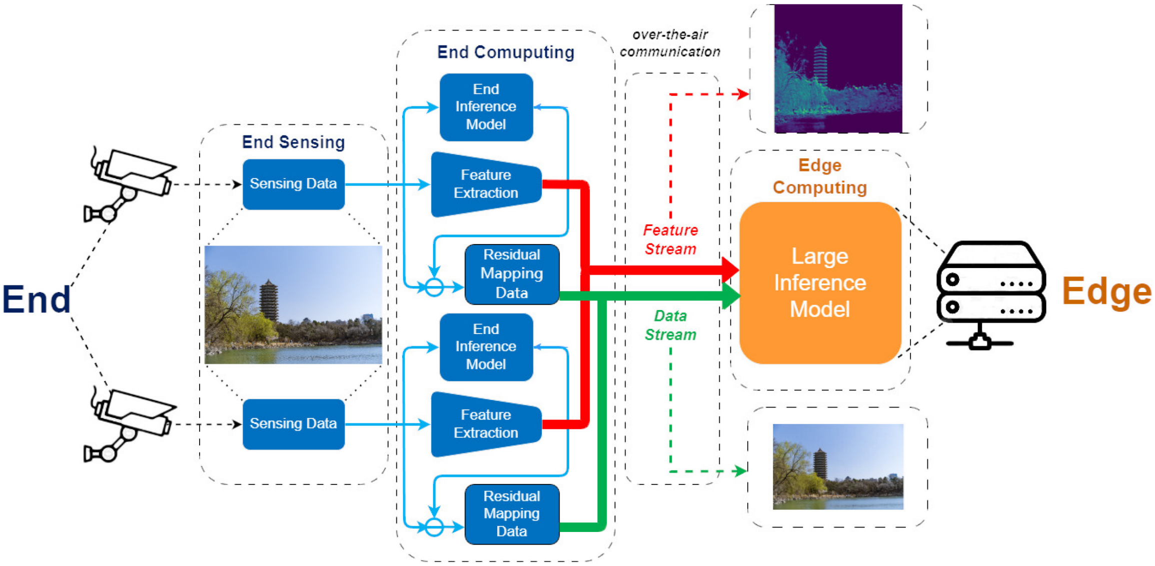


Fig. 1. Paradigm for an integrated end-edge model framework.

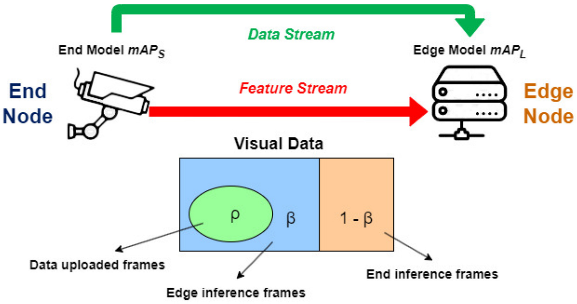


Fig. 2. System model for an integrated end-edge model framework.

residual mapping data of initial frames to help edge server for inference enhancement, called *on demand data transmission*.

In the next part, we propose an end-edge collaborative model to demonstrate the proposed integrated end-edge model framework.

### B. System Model Description

In this section, we introduce a basic system model of the integrated end-edge model framework. The system model includes an edge server as the edge node and $M$ end nodes as sensing, computing and communication modules, as shown in Fig. 2. Due to the limitation of the end nodes in terms of computation capability, it is challenging to support high accuracy computing tasks, e.g., precise visual classification, independently. The edge server with larger computing capability gives support to the end nodes in computing tasks, e.g., target classification, by its large scale onboard model. As we mentioned in Section II-A, two streams, i.e., feature stream and data stream, can be transmitted from the end nodes to the edge server via the OTA transmission. For clear depiction to the proposed model framework, we deliver the notations used in this article in the Table I.

We assume the number of the end nodes is $M$, each of which captures image frames at a rate of $N$ per second and each frame captured has $x$ pixels. For the collected data of the $i$th end node, as shown in Fig. 2, a fraction $\beta_i$ of the frames are uploaded to the edge server via OTA transmissions, while the rest fraction $1-\beta_i$ of the frames are analyzed by the local model of the end node. We use $\Psi_i$ to represent the set of images analyzed at the edge server, satisfying $|\Psi_i| = \beta_i \cdot N$. The feature of the frames transmitted to the edge server (i.e., the frames in $\Psi_i$), are extracted at the end node for the subsequent analysis at the edge server. Here we define $\bar{F}$ as the size of the features extracted from a frame of the image. Therefore, we present the transmission rate of the feature stream of the $i$th end node as

TABLE I
NOTATION

| | |
|---|---|
| $N$ | Number of frames captured per second |
| $x$ | Pixels per frame |
| $\beta$ | Task allocation ratio |
| $\Psi$ | Set of frames analysed on the edge |
| $\Phi$ | Set of frames with residual mapping data transmission |
| $F$ | Average size of extracted feature per frame |
| $R_F$ | Transmission data rate of feature stream |
| $\rho$ | Fraction of frames in $\Psi$ with residual mapping data transmission |
| **b** | Residual mapping data quantization bit |
| $R_D$ | Transmission data rate of data stream |
| $B_u$ | Uplink transmission bandwidth |
| $S$ | Spectrum efficiency |
| $mAP$ | mAP of the integrated end-edge model framework |
| $mAP_L$ | mAP of the edge model |
| $mAP_S$ | mAP of the end model |
| $r_L^k$ | Recall value of the edge model with the $k$th IoU |
| $r_S^k$ | Recall value of the end model with the $k$th IoU |
| $p_L^k$ | Precision value of the edge model with the $k$th IoU |
| $p_S^k$ | Precision value of the end model with the $k$th IoU |

$$R_{Fi} = \bar{F} \cdot \beta_i \cdot N. \tag{1}$$

For further analysis and better inference performance at the edge server, a proportion $\rho_i$ of the frames will be transmitted to the edge server from the $i$th end node, as shown in Fig. 2. The set of images for this residual mapping data at the edge server can be represented as $\Phi_i$, satisfying $|\Phi_i| = \rho_i \cdot N$. Meanwhile, for the residual mapping data needs to be jointly processed with the extracted feature mentioned above for image reconstruction, the image set $\Phi_i$ only includes the frames analyzed at the edge server, satisfying: $\Phi_i \subseteq \Psi_i$. Due to the OTA transmission bandwidth constraints, we need to quantize the residual mapping data properly to maximize using the limited bandwidth resource. We define $\boldsymbol{b} = \widehat{b}_{ij}\ \forall j \in \Phi_i$. Here, $\widehat{b}_{ij}$ refers to the number of quantization bits for the $j$th frame captured by the $i$th end node, and this quantization parameter comes from a discrete set of values defined as $\Omega$. Building upon this, we define the transmission rate of the data stream as

$$R_{Di} = \sum_{j\in\Phi_i} x \cdot \widehat{b}_{ij} \quad \forall \widehat{b}_{ij} \in \Omega. \tag{2}$$

Both the feature stream and the data stream are subject to bandwidth constraints on the uplink channel, i.e.,

$$R_{Fi} + R_{Di} \le B_i \cdot S_i \quad \forall i \in \{1, \ldots, M\}. \tag{3}$$

Here, $B_i$ represents the bandwidth occupancy of the uplink channel for the $i$th end node, and $S_i$ represents the spectral efficiency of uplink transmission for the $i$th end node. The total transmission bandwidths of the M end nodes should not exceed the total transmission bandwidth $B_u$, i.e.,

$$\sum_{i=1}^{M} B_i \le B_u. \tag{4}$$

And the spectrum efficiency for the $i$th end node (i.e., $S_i$), can be considered as available value, obtained through appropriate channel measurement techniques as discussed in [29], [30], independent of the wireless OTA propagation environments.

## III. Problem Formulation and Decomposition

In this section, we formulate the mAP maximization problem for the integrated end-edge model framework described in Section II-B, then decompose the problem into two subproblems for further study.

### A. Problem Formulation

The overall mAP of the integrated end-edge model framework is mainly determined by three factors: 1) the mAP of the edge server's model $\mathrm{mAP}_L$; 2) the set of mAPs of the end nodes' models $\mathbf{mAP}_{\boldsymbol{S}} = \{\mathrm{mAP}_{S1}, \mathrm{mAP}_{S2}, \ldots, \mathrm{mAP}_{SM}\}$, where $\mathrm{mAP}_{Si}$ represents the mAP value of the $i$th end node's model; and 3) the set of fractions of the transmitted frames analyzed at the edge server $\boldsymbol{\beta} = \{\beta_1, \beta_2, \ldots, \beta_M\}$, where the $\beta_i$ represents the task allocation ratio of the $i$th end node.

Because the value of $\mathrm{mAP}_L$ can be influenced by the task allocation ratio $\beta$ and the quantization bits $\boldsymbol{b}$, the edge server's inference performance to different end nodes can be different, referred to as $\mathrm{mAP}_{L1}, \ldots, \mathrm{mAP}_{LM}$, respectively. And we denote the mAP of the edge server by

$$\mathrm{mAP}_L = h(\mathrm{mAP}_{L1}, \mathrm{mAP}_{L2}, \ldots, \mathrm{mAP}_{LM}). \tag{5}$$

As related studies [31] have demonstrated, the mAP of the feature based inference converges to a stable level with an overhead much smaller than the residual mapping data size. Therefore, we can consider that each frame in set $\Psi$ has a fixed overhead. Building upon this, $\mathrm{mAP}_{Li}$ is only influenced by the residual mapping data allocation ratio $\rho_i$ and the corresponding quantization bits $\widehat{b}_{ij}$, which can be expressed as

$$\mathrm{mAP}_{Li} = g\big(\rho_i, \widehat{b}_{ij}\big) \quad \forall j \in \Phi_i \quad \forall i \in \{1, \ldots, M\}. \tag{6}$$

The value of the $\mathrm{mAP}_S$ for each end node here is constant.

To maximize the mAP of the integrated end-edge model framework, we need to jointly optimize the end-edge task allocation, multiuser uplink bandwidth allocation and residual mapping data transmission design. The problem can be formulated as

$$\max_{\beta_i, \rho_i, \boldsymbol{b}_i, B_i} \quad \mathrm{mAP} = f(\mathrm{mAP}_L, \boldsymbol{\beta}) \tag{7a}$$
$$\textbf{s.t.} \quad \mathrm{mAP}_{Li} = g\big(\rho_i, \widehat{b}_{ij}\big) \quad \forall j \in \Phi_i \quad \forall i \in \{1, \ldots, M\} \tag{7b}$$
$$0 \le \rho_i \le \beta_i \le 1 \quad \forall i \in \{1, \ldots, M\} \tag{7c}$$
$$\widehat{b}_{ij} \in \Omega \quad \forall j \in \Phi_i \quad \forall i \in \{1, \ldots, M\} \tag{7d}$$
$$R_{Fi} + R_{Di} \le B_i \cdot S_i \quad \forall i \in \{1, \ldots, M\} \tag{7e}$$
$$\sum_{i=1}^{M} B_i \le B_u \tag{7f}$$
$$\mathrm{mAP}_L = h(\mathrm{mAP}_{L1}, \mathrm{mAP}_{L2}, \ldots, \mathrm{mAP}_{LM}). \tag{7g}$$

Objective function (7a) represents the mAP maximization problem for the integrated end-edge model framework, which is a function of variables $\mathrm{mAP}_L$ and $\boldsymbol{\beta}$. Constraint (7b) captures the mAP of the edge model for each end node. Constraint (7c) notes that the fraction of frames with residual mapping data transmission should not exceed the fraction of frames analysed at the edge server. Constraint (7d) shows the quantization constraint for residual mapping data transmission. Constraints (7e) and (7f) involve the constraints of transmission data size for the feature stream and data stream, respectively.

Problem (7) faces challenges for direct solution for two main reasons. First, it is a mixed-integer programming problem that encompasses both discrete variables in $\boldsymbol{b}$ and continuous variables $\beta$, $\rho$, and $B_i$, which is NP hard. Second, the convexity of this problem cannot be guaranteed, as the convexity of the experimentally fitted functions g$(\cdot)$, $h(\cdot)$, and $f(\cdot)$ remains uncertain.

### B. Problem Decomposition

In this part, we decompose the problem into subproblems to separate the discrete variable $\boldsymbol{b}$ from the parameters $\boldsymbol{\beta}$ and $B_i$, and discuss the function $g(\cdot)$, $h(\cdot)$, and $f(\cdot)$ in two independent subproblems.

1) *Data Stream Subproblem:* In this subproblem, we optimize the set of frames for residual mapping data transmission $\rho_i$, and the quantization bits of each residual mapping data frame $\widehat{b}_{ij}\ \ \forall j \in \Phi_i$. Parameters about the task and transmission allocation are treated as fixed values. This subproblem aims to maximize the mAP of the edge server, which is associated with the independent

mAP$_{Li}$ of the $i$th end node, by optimizing the proportion of frames with the residual mapping data and their quantization bit numbers. We denote the subproblem as

$$\begin{aligned}
&\max_{\rho_i, \boldsymbol{b}_i} \quad \mathrm{mAP}_L && \text{(8a)}\\
&\textbf{s.t.} \quad \mathrm{mAP}_L = h(\mathrm{mAP}_{L1}, \mathrm{mAP}_{L2}, \ldots, \mathrm{mAP}_{LM}) && \text{(8b)}\\
&\qquad \mathrm{mAP}_{Li} = g\big(\rho_i, \widehat{b}_{ij}\big) \;\; \forall j \in \Phi_i \;\; \forall i \in \{1, \ldots, M\} && \text{(8c)}\\
&\qquad 0 \le \rho_i \le \beta_i \le 1 \;\; \forall i \in \{1, \ldots, M\} && \text{(8d)}\\
&\qquad \widehat{b}_{ij} \in \Omega \;\; \forall j \in \Phi_i \;\; \forall i \in \{1, \ldots, M\} && \text{(8e)}\\
&\qquad R_{Fi} + R_{Di} \le B_i \cdot S_i \;\; \forall i \in \{1, \ldots, M\}. && \text{(8f)}
\end{aligned}$$

2) *Feature Stream Subproblem:* In this subproblem, parameters related to the data stream are already optimized. This subproblem is formulated to maximize the mAP of the framework, which is the function of the mAP of the edge server and a series of $\beta_i$, i.e., the proportion of the target classification frames of the $i$th end node. The second subproblem can be formulated as below

$$\begin{aligned}
&\max_{\beta_i, B_i} \quad mAP = f(\mathrm{mAP}_L, \boldsymbol{\beta}) && \text{(9a)}\\
&\textbf{s.t.} \quad 0 \le \rho_i \le \beta_i \le 1 \quad \forall i \in \{1, \ldots, M\} && \text{(9b)}\\
&\qquad R_{Fi} + R_{Di} \le B_i \cdot S_i && \text{(9c)}\\
&\qquad \sum_{i=1}^{m} B_i \le B_u. && \text{(9d)}
\end{aligned}$$

## IV. Solutions and Analysis for Integrated End-Edge Model Framework

In this section, we solve the mAP maximization optimization problem in (7).

### A. Solution to Data Stream Subproblem

In this part, we focus on solving the subproblem presented in (8). Parameters related to the feature stream (i.e., $\beta_i$ and $B_i$) are considered fixed. Due to the independence of each frame, the quantization bits can vary in different frames. We denote the mAP of the edge model for the $i$th end node and the $k$th frame as mAP$^k_{Li}$.[1] To solve (8), we list the following key properties and assumptions about the function $g(\cdot)$ in constraint (8c).

*Remark 1:* The mAP of the $k$th frame and $i$th end node at the edge server, i.e. mAP$^k_{Li}$, monotonically increases with respect to the quantization bits of the residual mapping data $\widehat{b}_{ik}$.

*Assumption 1:* The mAP of the $k$th frame and $i$th end node at the edge server, i.e. mAP$^k_{Li}$, is a concave function of $\widehat{b}_{ik}$. Here mAP$^k_{Li}$ can be considered as a continuous variable with $\widehat{b}_{ik} \in \Omega$.

*Assumption 2:* The mAP of the $k$th frame and $i$th end node at the edge server, i.e. mAP$^k_{Li}$, is a concave function of $\widehat{b}_{ik}$. Here mAP$^k_{Li}$ can be considered as a continuous variable with $\widehat{b}_{ik} \in \Omega \cup \{0\}$, where $\widehat{b}_{ik} = 0$ corresponds to the case without residual mapping data transmission.

*Remark* 1 illustrates that precise quantization of the residual mapping data leads to improved mAP performance at the edge server, regardless of the specific frame or end node.

Assumption 1 has been validated through numerous experiments on various datasets [32], [33], [34], and it can be considered nearly accurate in our context, as it holds true in most current studies. Assumption 2 is an extension of Assumption 1 that includes the scenario where no residual mapping data is transmitted, i.e., the corresponding quantization bit is zero. Assumption 2 specifically addresses the situation where only the extracted features are transmitted to the edge server.

Additionally, since the mAP of a single frame mAP$^k_{Li}$ is not equal to the overall mAP at the edge server mAP$_{Li}$, Remark 1 and Assumption 1 do not guarantee the convexity of subproblem (8). Therefore, according to [35], we have two theorems below.

*Theorem 1:* Without the discrete quantization bits constraint (8e), the solution that maximizes mAP$_{Li}$ satisfies $\widehat{b}_1 = \widehat{b}_2 = \cdots = \widehat{b}_k \;\; \forall k \in \Psi_i$.

*Theorem 2:* When Assumption 2 is satisfied, the residual mapping data of all the frames in $\Psi_i$ should be sent to the edge server with the same quantization bits, i.e., $\widehat{b}_1 = \widehat{b}_2 = \cdots = \widehat{b}_k \;\; \forall k \in \Psi_i$.

With the two theorems and the given values of $R_{Fi}$, $B_i$, and $S_i$ (which are considered as fixed parameters), we maximize the mAP$_{Li}$ for the $i$th end node on the edge server. The (8f) for the $i$th end node can be rewritten as $\sum_{j \in \Psi_i} x\widehat{b}_{ij} \le B_i \cdot S_i - R_{Fi}$. When Assumption 2 is satisfied, we set $\rho_i = \beta_i$ and define $\widehat{b}^{opt}_{i1} = \widehat{b}^{opt}_{i2} = \cdots = \widehat{b}^{opt}_{ik} = ([B_i \cdot S_i - R_{Fi}]/[\mid \Psi_i \mid]) \;\; \forall k \in \Psi_i$. If $\widehat{b}^{opt}_{i1}$ does not satisfy (8e), we select the two closest values to $\widehat{b}^{opt}_{i1}$ within the set $\Psi_i \cup \{0\}$, denoted as $\widehat{b}^{\mathrm{low}}_i$ and $\widehat{b}^{\mathrm{high}}_i$. A proportion of $\lceil([\widehat{b}^{opt}_i - \widehat{b}^{\mathrm{low}}_i]/[\widehat{b}^{\mathrm{high}}_i - \widehat{b}^{\mathrm{low}}_i])\rfloor$ frames are quantized with $\widehat{b}^{\mathrm{high}}_i$ bits for the residual data, while the remaining frames are quantized with $\widehat{b}^{\mathrm{low}}_i$ bits. Here, $\lceil\cdot\rfloor$ represents rounding to the nearest integer.

However, when Assumption 2 is not satisfied, an additional method is required to solve (8). Let $E$ represents the increment of mAP with unit increment of data quantization bits for each frame. We employ a method that prioritizes allocating communication resources to the frame with the highest value of $E$. This approach is a heuristic algorithm and continues until all communication resources are fully allocated in this manner. In this way, we obtain the corresponding maximum mAP$_{Li}$ for the $i$th end node.

However, the same mAP$_{Li}$ value can correspond to different precision-recall pairs, leading to different values of mAP$_L$. Therefore, the relationship between the mAP$_L$ and the independent mAP$_{Li}$ is not a function. In order to solve (7), we fit $h(\cdot)$ with simulation experiments for closed-form polynomial expression, and study the properties of the fitted function.

Based on the results of simulation, function $h(\cdot)$ increases monotonically and shows concave property with respect to the single mAP$_{Li}$. Therefore, maximizing mAP$_L$ is equivalent to maximizing mAP$_{Li}$ for each end node. The method of mAP$_{Li}$

[1] The mAP of a single frame serves as a metric to measure the inference accuracy of a model, different from the mAP definition statistically.

maximization by optimizing $\rho_i$ and $\widehat{b}_{ij}$ has been introduced above. In that way, we solve (8).

### *B. Solution to Feature Stream Subproblem*

In this section, we solve (9). Parameters related to the data streams, which were optimized in Section IV-A, are treated as fixed values. The objective is to optimize the parameters associated with the feature stream to maximize the mAP of the framework.

The mAP of the framework can be viewed as the combined mAP of an edge model and multiple identical end models. To simplify subproblem (9), we present some properties in what follows.

*Remark 2:* The mAP performance of the end models, $\mathrm{mAP}_S$, is identical on different end nodes. The precision and recall values of a given threshold for all end nodes are also the same.

*Theorem 3:* The precision and recall values of the framework for the $t$-th IoU threshold, denoted by $p^t$ and $r^t$ satisfies

$$p^t = \frac{1}{\frac{1}{M}\sum_{i=1}^{M}\left(\frac{\beta_i}{p_L^t} + \frac{1-\beta_i}{p_S^t}\right)} \tag{10}$$

$$r^t = \frac{1}{\frac{1}{M}\sum_{i=1}^{M}\left(\frac{\beta_i}{r_L^t} + \frac{1-\beta_i}{r_S^t}\right)} \tag{11}$$

$$\begin{aligned} mAP = \frac{1}{2}\sum_{t=1}^{T}&\left[\frac{1}{\frac{1}{M}\sum_{i=1}^{M}\left(\frac{\beta_i}{p_L^t} + \frac{1-\beta_i}{p_S^t}\right)} + \frac{1}{\frac{1}{M}\sum_{i=1}^{M}\left(\frac{\beta_i}{p_L^{t-1}} + \frac{1-\beta_i}{p_S^{t-1}}\right)}\right] \\ &\times\left[\frac{1}{\frac{1}{M}\sum_{i=1}^{M}\left(\frac{\beta_i}{r_L^t} + \frac{1-\beta_i}{r_S^t}\right)} - \frac{1}{\frac{1}{M}\sum_{i=1}^{M}\left(\frac{\beta_i}{r_L^{t-1}} + \frac{1-\beta_i}{r_S^{t-1}}\right)}\right] \end{aligned} \tag{12}$$

where $p_L^t$ and $r_L^t$ are the precision and recall values of the edge model with the $t$-th IoU threshold; $p_S^t$ and $r_S^t$ are the precision and recall values of the end models with the $t$-th IoU threshold.

*Proof:* See in Appendix A. ■

However, although we have established the relationship between mAP and the precision-recall pairs, parameters to be optimized in (9) do not directly relate to precision and recall values. Therefore, we need to further explore the relationship between mAP and the optimization variables, leading to Theorem 4.

*Theorem 4:* The joint mAP of the framework satisfies

$$mAP \geq \frac{\mathrm{mAP}_L \cdot \mathrm{mAP}_S}{\frac{1}{m}\sum_{i=1}^{m}[\beta_i \mathrm{mAP}_S + (1-\beta_i)\mathrm{mAP}_L]}. \tag{13}$$

*Proof:* See in Appendix B. ■

Theorem 4 provides a lower bound for the mAP of the framework as a function of $\mathrm{mAP}_L$ and $\beta_i$ (where $\mathrm{mAP}_S$ is considered as a fixed parameter). By calculating the second derivative of the mAP with respect to $\mathrm{mAP}_L$, we get

$$\frac{\partial^2(\mathrm{mAP})}{\partial(\mathrm{mAP}_L)^2} = \frac{-2M \cdot {\mathrm{mAP}_S}^2 \sum_{i=1}^{M}\beta_i \sum_{i=1}^{M}(1-\beta_i)}{\sum_{i=1}^{M}[\beta_i \mathrm{mAP}_S + (1-\beta_i)\mathrm{mAP}_L]^3} \tag{14}$$

the second derivative of mAP with respect to $\mathrm{mAP}_L$ is nonpositive, which means mAP is a concave function with respect to $\mathrm{mAP}_L$. Similarly, we can prove the convexity of the function $f(\cdot)$ with respect to the $\beta_i \;\; \forall i \in \{1, \ldots, M\}$.

To examine the convexity of the constraints in (9), we need to analyze the convexity of $\mathrm{mAP}_L$ with respect to $R_{Di} + R_{Fi}$. By our denotion in Section IV-A, $\mathrm{mAP}_L = h(g(\rho_1, \widehat{b}_{1j}, \ldots, g(\rho_M, \widehat{b}_{Mj})))$. For single $\widehat{b}_{ij}$, the second derivative of $\mathrm{mAP}_L$ can be written as $h''(g(\cdot))\cdot g'(\cdot)^2 + h'(g(\cdot))\cdot g''(\cdot)$. As the function $g(\cdot)$ and $h(\cdot)$ are concave and the $h(\cdot)$ increases monotonically, the second derivative of $\mathrm{mAP}_L$ is nonpositive, which means the $\mathrm{mAP}_L$ is a concave function with respect to the $\widehat{b}_{ij}$. That implies $\mathrm{mAP}_L$ is concave with respect to $R_{Di}$. Since $R_{Fi}$ does not affect $\mathrm{mAP}_L$, $\mathrm{mAP}_L$ can also be considered as a concave function with respect to $R_{Di}+R_{Fi}$. Furthermore, $R_{Di} + R_{Fi}$ is a linear combination of the corresponding bandwidth $B_i$, indicating that $\mathrm{mAP}_L$ is concave with respect to $B_i$. In summary, subproblem (9) is proved to be concave.

Based on the lower bound in (13), subproblem (9) is concave with respect to $B_i$ and can be optimized using convex optimization methods. However, since we only have a lower bound of the mAP value, the optimization result will be suboptimal. To obtain the optimal result, we need additional strong constraints to derive a closed-form expression for the mAP, which can be written as follows:

*Theorem 5:* If constraints $p_L^t - p_S^t \ll p_L^t$, $r_L^t - r_S^t \ll r_L^t$, and $\forall 1 \leq t \leq T$ holds, the mAP of the framework can be approximated by

$$mAP \approx \frac{\mathrm{mAP}_L \cdot \mathrm{mAP}_S}{\frac{1}{M}\sum_{i=1}^{M}[\beta_i \mathrm{mAP}_S + (1-\beta_i)\mathrm{mAP}_L]}. \tag{15}$$

*Proof:* See in Appendix C. ■

With Theorem 5, function $f(\cdot)$ can be approximated by the expression in (15) and subproblem (9) can be solved by convexity optimization methods. If Theorem 5 can not be satisfied, the lower bound of mAP in (15) can be adopted for approximation to solve subproblem (9).

### *C. Overall Algorithm*

In this part, we give the overall algorithm for solving the optimization problem (7) of the integrated end-edge model framework, which is shown below.

With each available $B_i$, we first solve the subproblem (8) to obtain the optimal $\mathrm{mAP}_L$. Then, we deal with subproblem (9) to get the optimal solution to variables $\beta_i$, and the $B_i$ and $\beta_i$ will be substituted back into subproblem (8) to get the final solution to the $\beta_i$, $\rho_i$ and quantization bits $\mathbf{b}_i$.

If constraints $p_L^t - p_S^t \ll p_L^t$, $r_L^t - r_S^t \ll r_L^t$, and $\forall 1 \leq t \leq T$ are satisfied, Theorem 5 holds and we can get the optimal solution. Otherwise, we consider the mAP equals to its lower bound $([\mathrm{mAP}_L \cdot \mathrm{mAP}_S]/[(1/M)\sum_{i=1}^{M}[\beta_i \mathrm{mAP}_S + (1-\beta_i)\mathrm{mAP}_L]])$ to get the suboptimal solution.

Next, we deliver Theorem 6 to show the complexity of Algorithm 1 as follows.

*Theorem 6:* The complexity of Algorithm 1 is $O(N \cdot M^2 \cdot {B_u}^2)$.

**Algorithm 1:** Joint Model Design for the Integrated End-Edge Model Framework

```
Input: Variables $B$, $S_i$, $\Omega$, functions $g(\cdot)$, $h(\cdot)$;
For available values of $B_i$, solve subproblem (8) to obtain
the function of maximized $\mathrm{mAP}_L$ with respect to $B_i$;
if $p_L^t - p_S^t \ll p_L^t$, $r_L^t - r_S^t \ll r_L^t$ then
    Based on the solutions for the subproblem (8), solve
    subproblem (9) to obtain values of $\beta_i$;
end
else
    Obtain the lower bound of mAP with a suboptimal
    solution $\beta_i$;
end
Solve for $\rho_i$ and the corresponding optimal quantization
bits $\mathbf{b}_i$ corresponds to the $B_i$ and $\beta_i$ in problem (9);
Output: Task allocation variables $\beta_i$, data quantization
variables $\rho_i$, $\mathbf{b}_i$, communication variable $B_i$;
```

*Proof:* In Algorithm 1, the subproblems (8) and (9) are solved sequentially with enumerations of $B_i$, which is proportional to the $B_u$. For each available $B_i$, subproblem (8) is solved by the method in Section IV-A. If Assumption 2 is satisfied, the complexity of subproblem (8) is $O(M \cdot N)$, otherwise subproblem (8) can be solved with a complexity of $O(M \cdot N \cdot B_u)$ with heuristic method. In Section IV-B, the subproblem (9) can be solved by convex optimization. Since the optimization parameters $\beta_i, \rho_i, B_i$ and $\boldsymbol{b}_i$ are all elements, the optimization complexity can be seen as a constant value $C$. Thus, the complexity of Algorithm 1 is $M \cdot B_u \cdot O(M \cdot N \cdot B_u + C) = O(N \cdot M^2 \cdot {B_u}^2)$. ■

### *D. Algorithm Analysis*

In this part, we analysis the properties of the multiple end nodes in the framework.

The spectrum efficiency of transmissions from end nodes to the edge server may differ due to various channel gains, which in turn impacts the allocation of bandwidth. To clearly demonstrate the effect of spectrum efficiency on resource allocation, we present Theorem 7, which establishes the relationship between bandwidth and spectrum efficiency for each end node in the framework.

*Theorem 7:* When the total bandwidth for the framework is sufficient, the spectrum efficiency and optimal bandwidth allocation should satisfy

$$\frac{\partial h}{\partial \mathrm{mAP}_{Li}} \frac{\partial s(S_i \cdot B_i)}{\partial B_i} \cdot S_i = \frac{\partial h}{\partial \mathrm{mAP}_{Lj}} \frac{\partial s\left(S_j \cdot B_j\right)}{\partial B_j} \cdot S_j \quad \forall i, j \in \{1, \ldots, M\} \tag{16}$$

where the function $s(\cdot)$ satisfies $\mathrm{mAP}_{Li} = s(S_i \cdot B_i)$ and can be seen as an identity to the marginal utility of the optimal bandwidth for each end node.

*Proof:* See in Appendix D. ■

Theorem 7 show a property of the optimal bandwidth allocation for each end node with the sufficient bandwidth condition. The function $s(\cdot)$, as discussed in Appendix D, is a concave function, meaning that its derivative (i.e., $s'$) decreases as $B_i \cdot S_i$ increases. In Section IV-A, the function $h(\cdot)$ has been considered to be concave with respect to each $\mathrm{mAP}_{Li}$ $\forall i$, based on experimental fitting. Therefore, $(\partial h / \partial \mathrm{mAP}_{Li})$ is a decreasing function, which implies that the contribution of $\mathrm{mAP}_{Li}$ to $\mathrm{mAP}_L$ diminishes as Theorem 7 is a linear function of the quantization bits of the $i^{th}$ end node, i.e., $\boldsymbol{b}_i$, $\boldsymbol{b}_i$ also satisfies the properties of $B_i \cdot S_i$. As a result, analysis on the optimal resource allocation strategy for the proposed integrated end-edge model framework are listed below.

1) When the total bandwidth is sufficient, a higher quantization bit rate is allocated to the end node with better communication channel condition (i.e., higher spectrum efficiency), allowing for more quantization bits for residual mapping data transmission.
2) When the total bandwidth is insufficient, as an end node's allocated bandwidth increases, the contribution of additional bandwidth to $\mathrm{mAP}_L$ decreases. Therefore, bandwidth allocation among end nodes should ensure that the marginal utility for each end node is equal, i.e.,

$$\frac{\partial h}{\partial \mathrm{mAP}_{Li}} \cdot \frac{\partial s(S_i \cdot B_i)}{\partial B_i} \cdot S_i = \lambda \tag{17}$$

This implies that the allocation should avoid concentrating bandwidth excessively on a few specific end nodes. Instead, the framework should balance performance across all end nodes to ultimately achieve the highest possible mAP.

Moreover, we can further analyze the scenario where the number of frames generated per second varies for each end node. For example, let $N_i$ represent the number of frames captured per second by $i$th end node. The relationship between bandwidth allocation and the number of frames per second is described in the following theorem.

*Theorem 8:* When Theorem 7 holds, if the number of frames per second for different end nodes is independently adjustable, we have such equation holds

$$\frac{\partial h}{\partial \mathrm{mAP}_{Li}} \frac{\partial s\left(B_i \cdot \frac{S_i}{N_i}\right)}{\partial B_i} \cdot \frac{S_i}{N_i} = \frac{\partial h}{\partial \mathrm{mAP}_{Lj}} \frac{\partial s\left(B_j \cdot \frac{S_j}{N_j}\right)}{\partial B_j} \cdot \frac{S_j}{N_j} \tag{18}$$

where $N_i$ and $N_j$ refer to the number of frames generated per second for the $i$th and the $j$th end node.

*Proof:* According to the proof of Theorem 7 in Appendix D, function $s(\cdot)$ satisfies $s(\cdot) = g([B_i \cdot S_i - R_{Fi}]/[N \cdot x \cdot \beta]) = g(([B_i \cdot S_i - N \cdot F \cdot \beta]/[N \cdot x \cdot \beta])) = g(([B_i \cdot S_i]/[N \cdot x \cdot \beta]) - (F/x))$. The parameters $N$, $\beta$, $x$, and $F$ are fixed value, so the function $s(\cdot)$ can be written as $s(B_i \cdot S_i)$.

When $N$ can be different among different end nodes, referred to as $N_i$ for the $i$th end node, the function $s(\cdot)$ can be adopted as $g(([B_i \cdot S_i]/[N_i \cdot x \cdot \beta]) - (F/x))$. The corresponding expression of $s(\cdot)$ can be described as $s(B_i \cdot (S_i/N_i))$. With the same method in Appendix G, we can get (20), and Theorem 8 holds. ■

Theorem 8 incorporates parameter $N_i$ (the number of frames captured per second by the $i$th end node) into the bandwidth allocation optimization. Notably, $N_i$ has an opposite effect to $S_i$ (spectrum efficiency), and we can combine these two variables into one when analyzing the problem, which simplifies the optimization process. For example, if the $i$th end node doubles the number of frames captured per second, it has the same impact of halving its spectrum efficiency $S_i$ in terms of mAP.

However, when the total bandwidth of the system is insufficient, Theorems 7 and 8 mentioned above are not applicable. The reason lies in the significant discrepancies in the task allocation ratios, i.e., $\beta$, among different end nodes, which are influenced by their channel conditions. This, in turn, makes $s(\cdot)$ be a function with respect to $\beta_i$, expressed as $s(([B_i \cdot S_i]/[N_i \cdot \beta_i]))$. As $\beta_i$ is not a fixed parameter like $N_i$ and $S_i$, the analysis method in Theorems 7 and 8 can not work.

Due to the complex interdependencies among the optimization variables in (7), it is necessary to analyze the impact of the variables on the mAP. Such analysis enables a rapid qualitative evaluation of the optimization problem under different conditions, which prominents the priority and key metrics of communication resource allocation in the framework with multiple end nodes.

*Theorem 9:* The properties and impact of optimization variables on the mAP are described as follows.

1) variable $\beta_i$ directly influence the mAP value between the mAP$_S$ and mAP$_L$. With a high value of $\beta$, mAP is close to the value of mAP$_L$. A higher average value of $\beta$ does not necessarily imply better system performance.
2) Variables $B_i$, $S_i$, $\beta_i$, and $N_i$ determine the value of optimal quantization bits $\boldsymbol{b}_i$ together, equal to $([B_i \cdot S_i]/[N_i \cdot \beta_i \cdot x]) - (F/x)$.
3) Variable $\boldsymbol{b}_i$ impacts the value of mAP$_L$ at the edge server, and the function relationship shows concave property by fitting. With limited quantization bits, the value of mAP$_L$ reaches relatively higher when the quantization bits for each end node get closer.

Theorem 9 shows the method of communication resource allocation among the end nodes, which can be described as: each end node trades off between improving its ratio of frames processed at the edge server $\beta$, or increasing its quantization bits for residual mapping data $\boldsymbol{b}_i$. Meanwhile, as the concave relation between the mAP$_L$ and $\boldsymbol{b}_i$, the proper allocation is inclined to set the quantization bit rate in close values, to save the limited bandwidth. Based on Theorem 9, we draw several conclusion as follows.

*Corollary 1:* When the total bandwidth for the framework is insufficient, the end node with better communication condition(i.e., a higher spectrum efficiency or fewer number of frames generated per second) gets a higher task allocation ratio(i.e., $\beta$).

*Corollary 2:* In the conditions that Corollary 1 holds, the ratio of task allocation $\beta$ is positively correlated to the corresponding channel condition.

## V. Simulation Results

In this section, we evaluate the performance of the proposed integrated end-edge model framework, which includes task allocation, data quantization, and communication resource allocation. To better demonstrate the effectiveness of the proposed framework, we compare it with two baseline model framework: a centralized edge model framework and a distributed end model framework.

TABLE II
Simulation Parameters

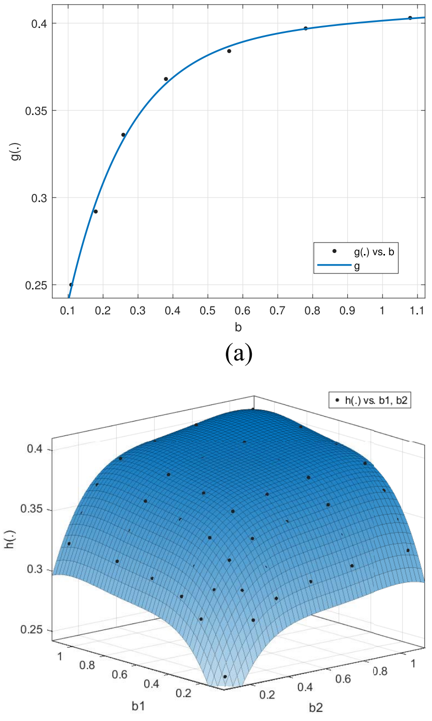
| parameter | value |
|---|---|
| Number of frames captured per second $N$ | 10 |
| Number of pixels per frame $x$ | $10^7$ |
| Average data size of the extracted feature $F$ | 0.86kbps |
| OTA transmission bandwidth $B$ | 20 MHz |
| Spectrum efficiency for $1^{st}$ end node $SE_1$ | 2.55 bit/s/Hz |
| Spectrum efficiency for $2^{nd}$ end node $SE_2$ | 5 bit/s/Hz |

In the centralized edge model framework, only the edge server possesses the computational capacity for data analysis. As a result, the end nodes transmit the extracted features and residual mapping data of all the frames to the edge server for processing. The number of quantization bits for the residual mapping data is determined by the available bandwidth and the spectrum efficiency of each end node. In the distributed end model framework, the end nodes perform all data analysis locally, and no data is transmitted to the edge server.

In this simulation, we use a visual-based target detection task with two end nodes, with the related parameters provided in Table II. The data for the experiment is based on the COCO2017 dataset [36] for target detection. Both the end nodes and the edge server run Faster R-CNN model [37], with 49M and 71M parameters, respectively. The training process for these models follows the methodology proposed in [38]. The function $g(\cdot)$ is fitted using the approach outlined in [39], and the function $h(\cdot)$ is tailored for the proposed framework. The proposed integrated model framework can be applied to various tasks, such as classification or semantic segmentation, as well as different modal datasets, including single modal datasets like ImageNet [40], CIFAR-10 [41], or PASCAL VOC [42], and multiple modal datasets like CMU-MOSEI [43] and How2 [44].

As shown in Fig. 3, we display the fitted form of functions $g(\cdot)$ and $h(\cdot)$ with respect to quantization bits. Furthermore, we can obtain the function value of $f(\cdot)$ with the closed form expression in Theorem 5, where the value of mAP$_L$ can be calculated by the fitted $g(\cdot)$ and $h(\cdot)$.

In Fig. 4, the mAP of the integrated end-edge model framework is shown for different numbers of frames captured by each UAV per second. With the total transmission bandwidth fixed, capturing more frames per second reduces the available bandwidth per frame for residual mapping data transmission, leading to a decline in mAP. The proposed framework can leverage the local processing capacity of the end nodes to handle the increased data when $N$ rises. As a result, the quantization bits for the residual mapping data is higher than that of the centralized edge model framework, resulting in a better mAP performance. Although an increase of $N$ leads to a decrease of mAP, the mAP of the proposed framework is strictly bounded below by the performance of the distributed end model framework. This ensures that the integrated end-edge model framework consistently outperforms both the baseline model framework for any given value of $N$. For the distributed end model framework, its mAP performance remains unaffected by transmission bandwidth or the number



(a)

(b)

Fig. 3. Fitted $g(\cdot)$ and $h(\cdot)$ functions. (a) Fitted function value of $g(\cdot)$ versus quantization bits number $b$. (b) Fitted function value of $h(\cdot)$ versus 1st and 2nd end node's quantization bits number b1 & b2.

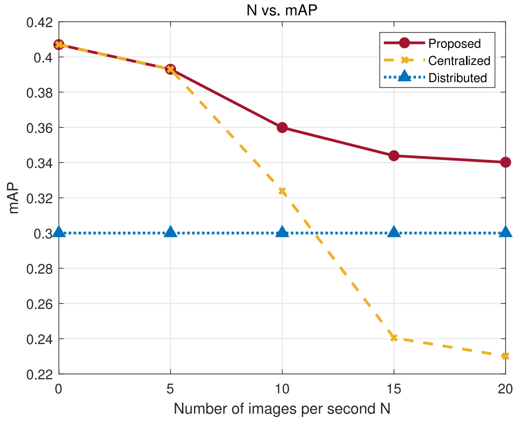


Fig. 4. Number of frames per second $N$ versus mAP .

of generated frames, so the mAP value remains constant across different values of $N$.

In Fig. 5, we evaluate the mAP of the integrated end-edge model framework under different total transmission bandwidths. The performance of the proposed framework improves as the transmission bandwidth increases, eventually converging to a stable value when the total bandwidth $B$ exceeds 60 MHz. The convergence of mAP suggests that when the bandwidth is sufficiently large, all residual mapping data can be transmitted to the edge server with an optimal quantization bit rate, leading to consistently high mAP performance. When the bandwidth is below 20 MHz, only a small portion of residual mapping data can be transmitted, leaving a significant amount to be analyzed locally by the end nodes, and the proposed framework outperforms the centralized edge framework in this case. On the other hand, the distributed end model framework processes all features and residual mapping data locally, without any need for transmission, resulting in a stable mAP regardless of the total bandwidth. When the total bandwidth is extremely limited, such as less than 1 MHz, the performance of the integrated framework approaches that of the distributed end model framework, as both rely primarily on local processing. Overall, the integrated end-edge model framework dynamically allocates communication resources between the end nodes and the edge server, consistently outperforming the two baseline frameworks across various bandwidths, as demonstrated in Fig. 5.

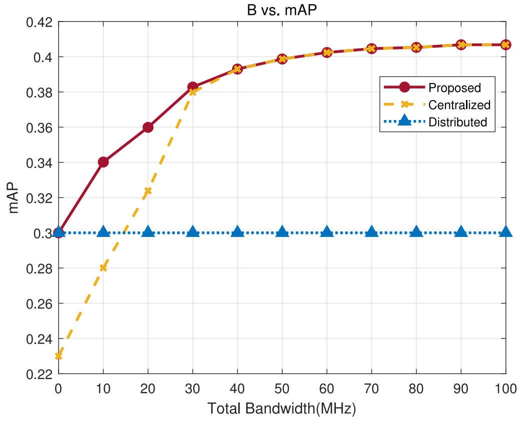


Fig. 5. Total bandwidth $B$ versus mAP.

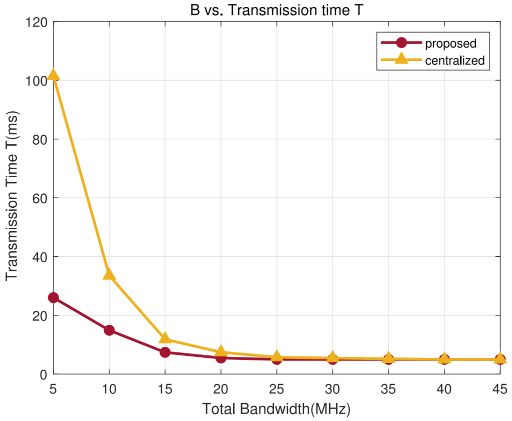


Fig. 6. Total bandwidth $B$ (MHz) versus transmission time $t$.

In Fig. 6, we present the transmission time required to achieve the same stable inference performance between the centralized edge model framework and our proposed framework. As shown in the figure, when the total bandwidth is below 40 MHz, the transmission time of the centralized edge model framework exceeds that of our proposed framework. This is because the limited bandwidth restricts the available quantization bits, leading to a decline in inference performance. To maintain stable performance, the centralized edge model framework requires more time to support high-quantization-bit data transmission, resulting in increased latency. Furthermore, the latency increase becomes more pronounced as total bandwidth decreases, posing significant challenges for real-time analysis applications. In contrast, the

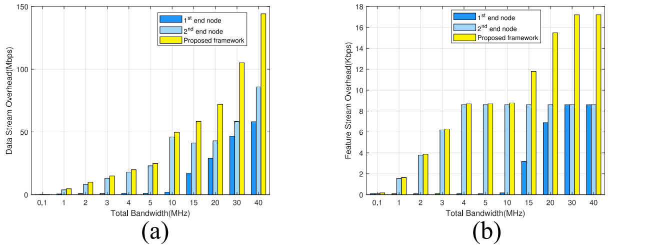


Fig. 7. Total bandwidth $B$ versus overhead of each stream. (a) Total bandwidth $B$ versus feature stream overhead. (b) Total bandwidth $B$ versus data stream overhead.

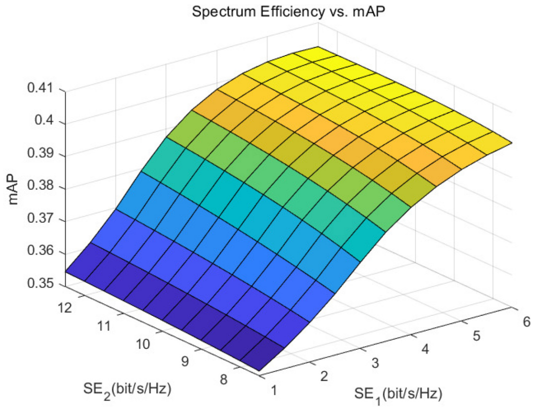


Fig. 8. Spectrum efficiency versus mAP.

proposed framework demonstrates a clear advantage in terms of latency over the centralized edge model framework.

In Fig. 7, we present the overhead of the data and feature streams under various total transmission bandwidths. The overhead represents the communication resource cost, i.e., the number of bits per second required to transmit the feature and data streams. As shown in Fig. 7(a), the data stream overhead increases as the total bandwidth expands. When bandwidth is limited, the second end node, with higher spectrum efficiency, receives dominant bandwidth for residual mapping data transmission. However, when total bandwidth reaches 15 MHz, it becomes beneficial to allocate additional bandwidth to the first end node. Consequently, the first end node's bandwidth increases significantly when the total bandwidth exceeds 10 MHz. The tendency of the feature stream overhead is similar to that of the data stream overhead. Initially, the second end node's feature stream overhead rises and reaches its upper bound, as the value of $\beta$ reaches 1. As the total bandwidth continues to expand, the first end node begins to transmit more frames captured, leading to an increment in its feature stream overhead. Overall, the overheads of both the data and feature streams are positively correlated with the total transmission bandwidth.

Fig. 8 illustrates the impact of each end node's spectrum efficiency on the mAP of the integrated end-edge model framework. In this scenario, we assume that the first end node consistently has a poorer transmission channel condition compared to the second end node. As shown in Fig. 8, the spectrum efficiency of both end nodes is positively correlated with the mAP of the integrated framework. As the second end node's spectrum efficiency increases, the influence of the first end node on the mAP decreases, because a larger portion of the bandwidth is allocated to the second end node for residual mapping data transmission. This indicates that, to maximize the mAP of the integrated model framework, bandwidth should be prioritized for the end node with higher spectrum efficiency.

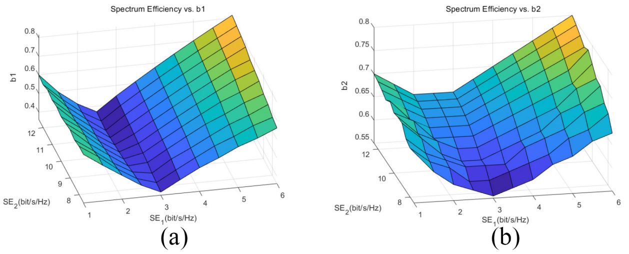


Fig. 9. Spectrum efficiency versus quantization bits $b_i$. (a) Spectrum efficiency SE versus the optimized quantization bits number for the 1st end node b1. (b) Spectrum efficiency SE versus the optimized quantization bits number for the 2nd end node b2.

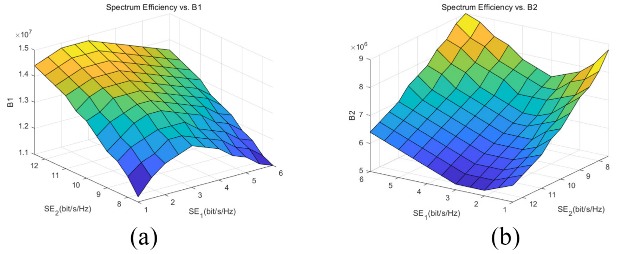


Fig. 10. Spectrum efficiency versus bandwidth allocation $B_i$. (a) Spectrum efficiency SE versus the optimized bandwidth cost for the 1st end node B1. (b) Spectrum efficiency SE vs. the optimized bandwidth cost for the 2nd end node B2.

In Figs. 9 and 10, we present the allocation of communication resources for two end nodes under different spectrum efficiencies. Among them, the second end node exhibits significantly better spectrum efficiency compared to the first end node. Based on this, we analyze the simulation results as follows.

1) In Fig. 9, we present the impact of spectrum efficiency on the quantization bits for residual mapping data across two end nodes. In both Fig. 9(a) and (b), we can observe a consistent trend in the changes of $b_1$ and $b_2$ as spectrum efficiency varies. Moreover, both values of quantization bits(i.e., $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$) remain close for any pair of spectrum efficiencies(i.e., $SE_1$ and $SE_2$), which validates the content discussed in Theorem 9, stating that the optimal quantization bits should remain close. Additionally, it can be observed that as the first end node's spectrum efficiency increases, the quantization bits for both end nodes exhibit a trend of first decreasing and then increasing. The reason for this behavior is that, when the first end node's spectrum efficiency is very low (e.g., 1 bit/s/Hz), bandwidth resources are scarce for all end nodes. Therefore, as the first end node's bandwidth utilization efficiency improves, the marginal gain of increasing $\beta_1$ on mAP is greater than the loss caused by the decrease in $b_1$. As a result, $\beta_1$ increases while $b_1$ decreases. However, as $\beta_1$ approaches its upper bound of 1, any additional spectrum efficiency can be fully utilized to improve the quantization bits. When considering the increase in the second end node's spectrum efficiency, with a low spectrum efficiencies

of the first end node, the bandwidth resources saved by the second end node cannot effectively enhance the first end node's data capacity due to the very limited spectrum efficiency of the first end node. As a result, the improvement in the first end node's quantization bits is not significant. However, as the first end node's spectrum efficiency gradually improves, the increment in quantization bits $\boldsymbol{b}_1$ accelerates significantly.

2) Fig. 10 describes the relationship between spectrum efficiency and bandwidth allocation. When the second end node's spectrum efficiency increases, with $\beta_2 = 1$, a large amount of bandwidth can be saved and allocated to the first end node, with lower spectrum efficiency, resulting in a monotonically decreasing trend, as shown in Fig. 10(b). However, when the first end node's spectrum efficiency increases, Theorem 9 indicates that $B_i$ is proportional to $b_i\beta_i/S_i$. When $\beta_1 < 1$, the $b_i\beta_i/S_i$ increases monotonically with respect to the corresponding $S_i$, so the first end node's bandwidth allocation increases monotonically. Once $\beta_1 = 1$, it keeps constant and $B_i$ is proportional to $b_i/S_i$ instead. Since the increment rate of $b_1$ is lower than that of spectrum efficiency $SE_1$, the first end node's bandwidth allocation starts to decrease.

## VI. Conclusion

This article has introduced a novel integrated end-edge model framework that enables concurrent data analysis by both the end nodes and edge servers. We have derived a closed-form expression for the lower bound of the overall mAP of the proposed framework with multiple end nodes. Then, we have addressed the mAP maximization problem through joint optimization of task allocation, transmission resource allocation, and data quantization. Simulation results have demonstrated the superior mAP performance of the proposed framework across a range of communication bandwidths and data sizes, consistently outperforming both the centralized edge model and the distributed end model frameworks. The key conclusions of this work are summarized as follows.

1) The performance gain of the proposed framework arises from dynamically adjusting the mAP of both the end model and the edge model through data uploading and task allocation, thereby maximizing the overall mAP of the framework.
2) When the total bandwidth is insufficient, the inference tasks are mostly allocated to the end nodes with high spectrum efficiency. While the transmission bandwidth is sufficient, each end node transmits most of its extracted feature and residual mapping data with similar quantization bits to the edge server for task.
3) In the framework with multiple end nodes, the optimal solution prioritizes allocating communication resources to the end nodes with higher spectrum efficiency and lower data size.

## Appendix A
## Proof to Theorem 3

According to the definition of the mean-average precision (i.e., mAP), the mAP value equals to the area under the precision-recall curve (i.e., PRC curve). And the PRC curve corresponds to the paired precision values and recall values in different IoU thresholds for the model. The precision value can be calculated by the number of the true positive samples (TP) and false positive samples (FP), written as $(\mathrm{TP}/\mathrm{TP}+\mathrm{FP})$. Similarly, the recall value is determined by the TP samples and false negative samples (FN), written as $(\mathrm{TP}/\mathrm{TP}+\mathrm{FN})$.

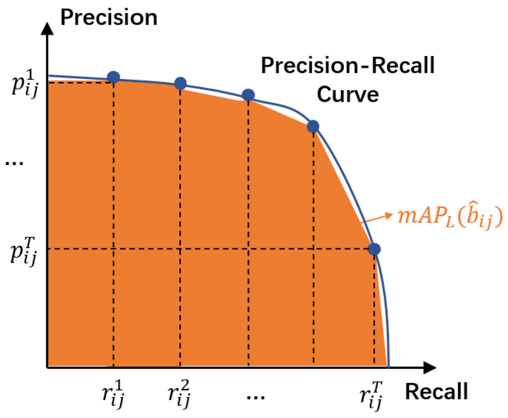


Fig. 11. Illustration for mAP and PRC.

Now we consider the situation for the $i$th end node. We assume the quantization bit of its $j$th frame and $k$th frame as $b_{ij}$ and $b_{ik}$, which is independent to each other. With independent quantization bits, we get independent mAP performances of the edge model for the classification task, written as $\mathrm{mAP}_L(b_{ij})$ and $\mathrm{mAP}_L(b_{ik})$. As the limited thresholds for measuring, we cannot get the accurate expression of the PRC curve. Therefore, we approximate the mAP value by calculating and adding the area of the trapezium which is surrounded by the adjacent points on the curve (i.e., the precision-recall value pairs for adjacent thresholds) and the $X$-axis. We take the $j$th frame as an example, and the expression of mAP is given as

$$\mathrm{mAP}_L(b_{ij}) \approx \frac{1}{2}\sum_{t=1}^{T}\left(p_{ij}^t + p_{ij}^{t-1}\right)\left(r_{ij}^t - r_{ij}^{t-1}\right) \tag{19}$$

where $p_{ij}^t$ and $r_{ij}^t$ correspond to the precision and recall value with the $t$-th threshold of the $j$th frame for the $i$th end node, and $T$ equals to the total number of the threshold, as shown in Fig. 9. And written in a more precise form, the $p_{ij}^t$ can be expressed as $p_{ij}^t = ([\mathrm{TP}_{ij}^t]/[\mathrm{TP}_{ij}^t + \mathrm{FP}_{ij}^t])$, and $r_{ij}^t$ equals $r_{ij}^t = ([\mathrm{TP}_{ij}^t]/[\mathrm{TP}_{ij}^t + \mathrm{FN}_{ij}^t])$, where $\mathrm{TP}_{ij}^t$ is the number of the true positive samples with the $t$-th threshold of $j$th frame for the $i$th end node, and so on. By fraction reduction, we get $p_{ij}^t$ as $p_{ij}^t = (1/[1 + x_{ij}^t])$, and $r_{ij}^t$ as $r_{ij}^t = (1/[1 + y_{ij}^t])$, with $x_{ij}^t = ([\mathrm{FP}_{ij}^t]/[\mathrm{TP}_{ij}^t])$ and $y_{ij}^t = ([\mathrm{FN}_{ij}^t]/[\mathrm{TP}_{ij}^t])$.

According to the definition of precision and recall value, we get $p^t = (\mathrm{TP}^t/[\mathrm{TP}^t + \mathrm{FP}^t])$ and $r^t = (\mathrm{TP}^t/[\mathrm{TP}^t + \mathrm{FN}^t])$, where $\mathrm{TP}^t$ is the number of the true positive samples with the $t$-th IoU threshold, $\mathrm{FP}^t$ is the number of the false positive samples with the $t$-th IoU threshold and $\mathrm{FN}^t$ is the number of the false negative samples with the $t$-th IoU threshold. For the precision value, we have

$$p_L^t = \frac{1}{1 + x_L^t}, x_L^t = \frac{\mathrm{FP}_L^t}{\mathrm{TP}_L^t} \tag{20}$$

$$p_{Si}^t = \frac{1}{1 + x_{Si}^t}, x_{Si}^t = \frac{\mathrm{FP}_{Si}^t}{\mathrm{TP}_{Si}^t} \tag{21}$$

where $p_{Si}^t$ is the precision value of the $i$th end node's model with the $t$-th IoU threshold.

Equations (20) and (21) show that one TP sample at the edge server corresponds to $x_L^t$ FP samples, and one TP samples at the $i$th end node is accompanied with $x_{Si}^t$ FP samples. Considering the ratio of samples at the edge server is $(1/M) \sum_{i=1}^{M} \beta_i$, we calculate $p^t$ by

$$\begin{aligned} p^t &= \frac{\mathrm{TP}^t}{\mathrm{TP}^t + \mathrm{FP}^t} \\ &= \frac{1}{1 + \frac{1}{M}\sum_{i=1}^{M} \beta_i x_L^t + \frac{1}{M}\sum_{i=1}^{M}(1-\beta_i) x_{Si}^t} \\ &= \frac{1}{1 + \frac{1}{M}\sum_{i=1}^{M} \beta_i \left(\frac{1}{p_L^t} - 1\right) + \frac{1}{M}\sum_{i=1}^{M}(1-\beta_i)\left(\frac{1}{p_{Si}^t} - 1\right)} \\ &= \frac{1}{\frac{1}{M}\sum_{i=1}^{M}\left(\frac{\beta_i}{p_L^t} + \frac{1-\beta_i}{p_{Si}^t}\right)}. \end{aligned} \tag{22}$$

As *Remark* 2 shows, the precision values of all end models are the same. Therefore, a fixed value $p_S^t$ can replace all the $p_{Si}^t$ of different end nodes, and equation in (10) can be obtained. Similarly, the equation for the recall value can be expressed as (11).

To get the approximated joint mAP value, we can substitute (10) and (11) into (19), thus Theorem 3 holds.

## Appendix B
## Proof to Theorem 4

As (19) shows, the mAP is a linear function of different precision-recall pairs. Therefore, we can analyze the relation among mAP, $\mathrm{mAP}_L$ and $\mathrm{mAP}_S$ with the precision-recall pairs, and the corresponding properties still hold with necessary linear transformations.

Define $\xi^t = p^t r^t$ as a variable with relation to the precision-recall pairs of the framework. Similarly, we can define $\xi_L^t = p_L^t r_L^t$ and $\xi_S^t = p_S^t r_S^t$ as the variables of the edge model and end models, respectively. By (10) and (11), we can expand $\xi^t$ as

$$\begin{aligned} \xi^t &= p^t r^t \\ &= \frac{1}{\frac{1}{M}\sum\limits_{i=1}^{M}\left(\frac{\beta_i}{p_L^t} + \frac{1-\beta_i}{p_S^t}\right) \cdot \frac{1}{M}\sum\limits_{i=1}^{M}\left(\frac{\beta_i}{r_L^t} + \frac{1-\beta_i}{r_S^t}\right)} \\ &= \frac{M^2 \xi_L^t \xi_S^t}{\sum\limits_{i=1}^{M}\sum\limits_{j=1}^{M}\left[\beta_i\beta_j\xi_S^t + (1-\beta_i)(1-\beta_j)\xi_L^t\right] + \sum\limits_{i=1}^{M}\beta_i\sum\limits_{j=1}^{M}(1-\beta_j)\left(p_S^t r_L^t + r_S^t p_L^t\right)} \\ &= \frac{M^2 \xi_L^t \xi_S^t}{M\sum\limits_{i=1}^{M}\beta_i\xi_S^t + M\sum\limits_{j=1}^{M}(1-\beta_j)\xi_L^t - \sum\limits_{i=1}^{M}\beta_i\sum\limits_{j=1}^{M}(1-\beta_j)\left(p_L^t - p_S^t\right)\left(r_L^t - r_S^t\right)} \\ &= \frac{M^2 \xi_L^t \xi_S^t}{M\sum\limits_{i=1}^{M}\beta_i\xi_S^t + M\sum\limits_{j=1}^{m}(1-\beta_j)\xi_L^t - \sum\limits_{i=1}^{M}\beta_i\sum\limits_{j=1}^{M}(1-\beta_j)\Delta} \end{aligned} \tag{23}$$

where $\Delta = (p_L^t - p_S^t)(r_L^t - r_S^t)$ shows the inference capacity difference between the edge model and the end models. We make a proper assumption that the edge model has a better performance of inference capacity than the end models, i.e., $p_L^t - p_S^t > 0$ and $r_L^t - r_S^t > 0$. That makes $\Delta > 0$ holds. As a result, if $\Delta$ in (23) is reduced, the following relation holds:

$$\begin{aligned} \xi^t &> \frac{M\xi_L^t\xi_S^t}{\sum\limits_{i=1}^{M}\beta_i\xi_S^t + \sum\limits_{j=1}^{M}(1-\beta_j)\xi_L^t} \\ &= \frac{\xi_L^t\xi_S^t}{\frac{1}{M}\sum\limits_{i=1}^{M}\left(\beta_i\xi_S^t + (1-\beta_i)\xi_L^t\right)}. \end{aligned} \tag{24}$$

As mAP is a linear combination of a series of $\xi^t$, (24) also holds for mAP, which can be written as

$$mAP > \frac{\mathrm{mAP}_L \cdot \mathrm{mAP}_S}{\frac{1}{M}\sum_{i=1}^{M}[\beta_i \mathrm{mAP}_S + (1-\beta_i)\mathrm{mAP}_L]} \tag{25}$$

thus Theorem 4 holds.

## Appendix C
## Proof to Theorem 5

According to Theorem 4, we have

$$\xi^t = \frac{M^2\xi_L^t\xi_S^t}{M\sum\limits_{i=1}^{M}\beta_i\xi_S^t + M\sum\limits_{j=1}^{M}(1-\beta_j)\xi_L^t - \sum\limits_{i=1}^{M}\beta_i\sum\limits_{j=1}^{M}(1-\beta_j)\Delta} \tag{26}$$

where $\Delta$ equals $(p_L^t - p_S^t)(r_L^t - r_S^t)$. If $p_L^t - p_S^t \ll p_L^t$ and $r_L^t - r_S^t \ll r_L^t$ are satisfied, $\Delta \ll \xi_L^t$ holds. Therefore, we have

$$\xi^t \approx \frac{\xi_L \cdot \xi_S}{\frac{1}{M}\sum_{i=1}^{M}[\beta_i\xi_S + (1-\beta_i)\xi_L]}. \tag{27}$$

As the mAP is a linear combination of a series of $\xi^t$, (27) also holds for mAP, thus Theorem 5 holds.

## Appendix D
## Proof to Theorem 7

In Section V-A, we get the mAP of the $i$th end node at the edge server $\mathrm{mAP}_{Li}$ as function $g(\cdot)$, defined as $g(\rho_i, \widehat{b_{ij}})$. When the total bandwidth is sufficient for data transmission for $M$ end nodes, parameter $\beta$ can be seen as fixed, and we have the optimal $\rho_i = \beta_i = \beta$ and $b_{ij} = b_i = ([B_i \cdot S_i]/[N \cdot x \cdot \beta]) - (F/x)$.

For the $i$th end node, we have the relation $\mathrm{mAP}_{Li} = g(b_i) = g(([B_i \cdot S_i]/[N \cdot x \cdot \beta]) - (F/x))$. As the value of $N$, $x$, $F$, and $\beta$ can be seen as fixed, we define $\mathrm{mAP}_{Li}$ as a function of $B_i \cdot S_i$, described as $\mathrm{mAP}_{Li} = s(B_i \cdot S_i)$. Since the relationship between the $b_i$ and $B_i \cdot S_i$ is linear, the function $s(\cdot)$ inherits the properties of $g(\cdot)$, such as the concave property.

As introduced in Section IV-A, the mAP of the edge server $\mathrm{mAP}_L$, is defined as $\mathrm{mAP}_L = h(\mathrm{mAP}_{L1}, \ldots, \mathrm{mAP}_{LM})$. The mAP optimization problem can be converted to

$$\max_{B_1,\ldots,B_M} \quad \mathrm{mAP}_L = h(s(B_1 \cdot S_1), \ldots, s(B_M \cdot S_M)) \tag{28}$$

with constraint $\sum_{i=1}^{M} B_i = B_u$.

To solve this problem, we define the Lagrange multiplier $\lambda$, and construct the target function $L(B_1, \ldots, B_M, \lambda) = h(s(B_1 \cdot S_1), \ldots, s(B_M \cdot S_M)) + \lambda(B_u - \sum_{i=1}^{M} B_i)$.

For each $B_i$, we calculate the partial derivative of $L$, satisfying

$$\frac{\partial L}{\partial B_i} = \frac{\partial h}{\partial \mathrm{mAP}_{Li}} \cdot \frac{\partial s(B_i \cdot S_i)}{\partial B_i} - \lambda = 0. \tag{29}$$

So, we can get the optimization condition

$$\frac{\partial h}{\partial \mathrm{mAP}_{Li}} \cdot s'(B_i \cdot S_i) \cdot S_i = \lambda. \tag{30}$$

For any two random end nodes, the equation always holds, thus the Theorem 7 holds.

## REFERENCES

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015.

[2] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani, "Deep learning for IoT big data and streaming analytics: A survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2923–2960, 4th Quart., 2018.

[3] D. Lahat, T. Adali, and C. Jutten, "Multimodal data fusion: An overview of methods, challenges, and prospects," *Proc. IEEE*, vol. 103, no. 9, pp. 1449–1477, Sep. 2015.

[4] M. Ilyas, "IoT applications in smart cities," in *Proc. Int. Conf. Electron. Commun., Internet Things Big Data (ICEIB)*, Yilan, Taiwan, 2021, pp. 44–47.

[5] P. Nirmala, S. Ramesh, M. Tamilselvi, G. Ramkumar, and G. Anitha, "An artificial intelligence enabled smart industrial automation system based on Internet of Things assistance," in *Proc. Int. Conf. Adv. Comput., Commun. Appl. Informat. (ACCAI)*, 2022, pp. 1–6.

[6] Y. Wang and X. Bi, "The application of computer vision target recognition technology in autonomous driving," in *Proc. 3rd Int. Conf. Artif. Intell. Auton. Robot Syst. (AIARS)*, Bristol, U.K., 2024, pp. 519–524.

[7] S. S. Priya, M. H. Al-Fatlawy, N. Khare, V. Mahalakshmi, and S. S. Ganesh, "Machine and deep learning classifications for IoT-Enabled healthcare devices," in *Proc. 4th Int. Conf. Comput., Autom. Knowl. Manag. (ICCAKM)*, Dubai, UAE, 2023, pp. 1–7.

[8] M. Xu et al., "Sparks of generative pretrained transformers in edge intelligence for the metaverse: Caching and inference for mobile artificial intelligence-generated content services," *IEEE Veh. Technol. Mag.*, vol. 18, no. 4, pp. 35–44, Dec. 2023.

[9] J. Wen et al., "From generative AI to generative Internet of Things: Fundamentals, framework, and outlooks," *IEEE Internet Things Mag.*, vol. 7, no. 3, pp. 30–37, May 2024.

[10] A. Dubey et al., "The llama 3 herd of models," 2024, *arXiv:2407.21783*.

[11] T. G. Gemini, "Gemini: A family of highly capable multimodal models," 2023, *arXiv:2312.11805*.

[12] Y. Cao et al., "A comprehensive survey of ai-generated content (AIGC): A history of generative ai from GAN to ChatGPT," 2023, *arXiv:2303.04226*.

[13] J. Wang et al., "Generative AI based secure wireless sensing for ISAC networks," 2024, *arXiv:2408.11398*.

[14] J. Wang et al., "Generative AI for integrated sensing and communication: Insights from the physical layer perspective," *IEEE Wireless Commun.*, vol. 31, no. 5, pp. 246–255, Oct. 2024.

[15] V. Puranik, Sharmila, A. Ranjan, and A. Kumari, "Automation in agriculture and IoT," in *Proc. 4th Int. Conf. Internet Things, Smart Innovat. Usages (IoT-SIU)*, 2019, pp. 1–6.

[16] M. Xu et al., "Generative AI-empowered simulation for autonomous driving in vehicular mixed reality metaverses," *IEEE J. Sel. Topics Signal Process.*, vol. 17, no. 5, pp. 1064–1079, Sep. 2023.

[17] Z. Yue et al., "Generative diffusion-based contract design for efficient AI twins migration in vehicular embodied AI networks," 2024, *arXiv:2410.01176*.

[18] Y. Kang et al., "Hybrid-generative diffusion models for attack-oriented twin migration in vehicular metaverses," 2024, *arXiv:2407.11036*.

[19] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, "Federated learning for Internet of Things: Recent advances, taxonomy, and open challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1759–1799, 3rd Quart., 2021.

[20] R. Varghese and M. Sambath, "YOLOv8: A novel object detection algorithm with enhanced performance and robustness," in *Proc. Int. Conf. Adv. Data Eng. Intell. Comput. Syst. (ADICS)*, 2024, pp. 1–6.

[21] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. Vincent Poor, "Federated learning for Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1622–1658, 3rd Quart., 2021.

[22] Z. Sun et al., "Cloud-edge collaboration in Industrial Internet of Things: A joint offloading scheme based on resource prediction," *IEEE Internet Things J.*, vol. 9, no. 18, pp. 17014–17025, Sep. 2022.

[23] X. Xiong, K. Zheng, L. Lei, and L. Hou, "Resource allocation based on deep reinforcement learning in IoT edge computing," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 6, pp. 1133–1146, Jun. 2020.

[24] R. Han, S. Wen, C. H. Liu, Y. Yuan, G. Wang, and L. Y. Chen, "EdgeTuner: Fast scheduling algorithm tuning for dynamic edge-cloud workloads and resources," in *Proc. IEEE Conf. Comput. Commun.*, London, U.K., 2022, pp. 880–889.

[25] J. Yang, L. Ran, J. Dang, Y. Wang, and Z. Qu, "Deeper multiscale encoding–decoding feature fusion network for change detection of VHR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, Nov. 2023, Art. no. 6012105.

[26] W. Gao et al., "Digital retina: A Way to make The city brain more efficient by visual coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 11, pp. 4147–4161, Nov. 2021.

[27] L. Duan, J. Liu, W. Yang, T. Huang, and W. Gao, "Video coding for machines: A paradigm of collaborative compression and intelligent analytics," *IEEE Trans. Image Process.*, vol. 29, pp. 8680–8695, 2020.

[28] S. Ma, X. Zhang, C. Jia, Z. Zhao, S. Wang, and S. Wang, "Image and video compression with neural networks: A review," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 6, pp. 1683–1698, Jun. 2020.

[29] F. D. L. Coutinho, H. S. Silva, P. Georgieva, and A. S. R. Oliveira, "A novel CNN-based architecture for Over-the-Air 5G OFDM channel estimation," in *Proc. IEEE/MTT-S Int. Microw. Symp.*, Washington, DC, USA, 2024, pp. 98–101.

[30] A. S. Doshi, M. Gupta, and J. G. Andrews, "Over-the-Air design of GAN training for mmWave MIMO channel estimation," *IEEE J. Sel. Areas Inf. Theory*, vol. 3, no. 3, pp. 557–573, Sep. 2022.

[31] W. Yang, H. Huang, Y. Hu, L.-Y. Duan, and J. Liu, "Video coding for machines: Compact visual representation compression for intelligent collaborative analytics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 7, pp. 5174–5191, Jul. 2024.

[32] S. Wang, Z. Wang, S. Wang, and Y. Ye, "End-to-end compression toward machine vision: Network architecture design and optimization," *IEEE Open J. Circuits Syst.*, vol. 2, pp. 675–685, 2021.

[33] W. Li, W. Sun, Y. Zhao, Z. Yuan, and Y. Liu, "Deep image compression with residual learning," *Appl. Sci.*, vol. 10, no. 11, p. 4023, Jun. 2020.

[34] M. Akbari, J. Liang, J. Han, and C. Tu, "Learned variable-rate image compression with residual divisive normalization," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, London, U.K., Jul. 2020, pp. 1–6.

[35] S. Zhang et al., "Large models for aerial edges: An edge-cloud model evolution and communication paradigm," *IEEE J. Sel. Areas Commun.*, vol. 43, no. 1, pp. 21–35, Jan. 2025.

[36] T. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. 13th Eur. Conf. Comput. Vis. (ECCV)*, Zurich, Switzerland, 2014, pp. 740–755.

[37] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Toward real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[38] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick. "Detectron2." 2019. [Online]. Available: https://github.com/facebookresearch/detectron2

[39] C. Zhengxue, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Seattle, WA, USA, 2020, pp. 7939–7948.

[40] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, pp. 211–252, Dec. 2015.

[41] "The CIFAR-10 dataset." 2022. [Online]. Available: https://www.cs.toronto.edu/~kriz/cifar.html

[42] "The PASCAL visual object classes homepage." Accessed: Oct. 24, 2024. [Online]. Available: http://host.robots.ox.ac.uk/pascal/VOC/

[43] A. Zadeh, P. Liang, S. Poria, E. Cambria, and L. P. Morency, "Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph," in *Proc. Annu. Meeting Assoc. Comput. Linguist.*, Melbourne, VIC, Australia, 2018, pp. 2236–2246.

[44] "The How2 dataset." Accessed: Jan. 11, 2025. [Online]. Available: https://github.com/srvk/how2-dataset

**Xinbo Yu** is currently pursuing the B.S. degree in electronic information engineering with the School of Electronics Engineering and Computer Science, Peking University, Beijing, China.

His research interests mainly focuses on wireless communication and machine learning.

**Shuhang Zhang** (Member, IEEE) received the Ph.D. and B.S. degrees in electronic engineering from the School of Electrical Engineering and Computer Science, Peking University, Beijing, China, in 2021 and 2016, respectively.

Since 2023, he was with Pengcheng Laboratory, Shenzhen, China, as an Assistant Researcher. Before joining Pengcheng Laboratory, he was with Huawei Technology Company Ltd., Shenzhen, from 2021 to 2023. He has published over 20 papers in IEEE/ACM Journals, including multiple ESI hot papers and ESI highly cited papers. His current research interests include space-air-ground integrated network, artificial intelligence, and signal processing.

Dr. Zhang has won the 2021 IEEE Communication Society Heinrich Hertz Award, the 2021 IEEE Communication Society Asia–Pacific Outstanding Paper Award, and the 2019 First Prize of IEEE Communication Society Student Competition.

**Hongliang Zhang** (Member, IEEE) received the B.S. and Ph.D. degrees from the School of Electrical Engineering and Computer Science, Peking University, Beijing, China, in 2014 and 2019, respectively.

He is currently an Endowed Boya Young Fellow Assistant Professor with the School of Electronics, Peking University. His current research interests include intelligent surfaces, aerial access networks, and Internet of Things.

Dr. Zhang received the Best Doctoral Thesis Award from the Chinese Institute of Electronics in 2019. He is the Winner of the Outstanding Leadership Award as the Publicity Chair for IEEE EUC in 2022. He is also the recipient of the 2024 IEEE GLOBECOM Best Paper Award, the 2024 IEEE/CIC ICCC Best Demo Award, the 2023 IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award, the 2021 IEEE Comsoc Heinrich Hertz Award for Best Communications Letters, and the 2021 IEEE ComSoc Asia–Pacific Outstanding Paper Award. He has served as a TPC member and a workshop co-chair for many IEEE conferences. He is currently an Editor of IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING, IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE COMMUNICATIONS LETTERS, and IET COMMUNICATIONS. He is an Exemplary Editor of IEEE COMMUNICATIONS LETTERS in 2023.

**Lingyang Song** (Fellow, IEEE) received the Ph.D. degree from the University of York, York, U.K., in 2007.

He was a Research Fellow with the University of Oslo, Oslo, Norway, until rejoining Philips Research, Cambridge, U.K., in March 2008. In May 2009, he joined the Department of Electronics, School of Electronics Engineering and Computer Science, Peking University, Beijing, China, where he is currently a Boya Distinguished Professor. His current research interests include wireless communication and networks, signal processing, and machine learning.

Dr. Song received the K. M. Stott Prize from the University of York for excellent research. He was a recipient of the IEEE Leonard G. Abraham Prize in 2016 and the IEEE Asia–Pacific Young Researcher Award in 2012. He has been an IEEE Distinguished Lecturer since 2015.