

The First Verification Test of Space-Ground Collaborative Intelligence via Cloud-Native Satellites

Wang Shangguang^{1,2}, Zhang Qiyang¹, Xing Ruolin¹, Qi Fei³, Xu Mengwei^{1,*}

¹ State Key Laboratory of Networking and Switching, Beijing University of Posts and Telecommunications, Beijing 100876, China

² Star Network and Intelligent Computing Laboratory, Beiyou Shenzhen Institute, Shenzhen 518000, China

³ Huawei Technologies Co., Ltd, Shenzhen 518129, China

* The corresponding author, email: mwx@bupt.edu.cn

Cite as: S. Wang, Q. Zhang, *et al.*, "The first verification test of space-ground collaborative intelligence via cloud-native satellites," *China Communications*, 2024, vol. 21, no. 4, pp. 208-217. DOI: 10.23919/JCC.fa.2022-0422.202404

Abstract: Recent advancements in satellite technologies and the declining cost of access to space have led to the emergence of large satellite constellations in Low Earth Orbit (LEO). However, these constellations often rely on bent-pipe architecture, resulting in high communication costs. Existing onboard inference architectures suffer from limitations in terms of low accuracy and inflexibility in the deployment and management of in-orbit applications. To address these challenges, we propose a cloud-native-based satellite design specifically tailored for Earth Observation tasks, enabling diverse computing paradigms. In this work, we present a case study of a satellite-ground collaborative inference system deployed in the Tiansuan constellation, demonstrating a remarkable 50% accuracy improvement and a substantial 90% data reduction. Our work sheds light on in-orbit energy, where in-orbit computing accounts for 17% of the total onboard energy consumption. Our approach represents a significant advancement of cloud-native satellite, aiming to enhance the accuracy of in-orbit computing while simultaneously reducing communication cost.

Keywords: cloud-native satellite; orbital edge computing; satellite inference; verification test

I. INTRODUCTION

With the evolution of next-generation communication networks and advanced technologies, the number of end-user connections and application requirements are increasing. It is estimated that two-thirds of the population in the world will have access to the Internet and billions of devices will be connected to the Internet by 2023 [1]. However, there are still more than 80% of the land and 90% of the ocean on the Earth without network coverage. Moreover, real-time and high-resolution satellite imagery have vital use in getting along with nature. The terrestrial network is extremely vulnerable in the face of natural disasters, such as earthquakes and floods.

In recent years, however, the increasing investments in satellite-related technologies make the Low Earth Orbit (LEO) satellite constellation a strong complement to terrestrial networks [2]. Numerous competitors have disclosed efforts to deploy LEO constellations, including SpaceX [3], Telesat [4], and Amazon [5]. These LEO constellations are promising to blanket the globe with low-latency broadband Internet due to the declining costs of satellite manufacturing and launching today. Satellites are typically categorized into distinct types, each specifically designed to fulfill various tasks such as communication, guidance, and sensing. In this work, we focus on the Earth Observation satellites.

Received: Jun. 27, 2022

Revised: Jun. 21, 2023

Editor: Xiao Zhenyu

Since bent-pipe satellites do not have in-orbit processing capabilities, raw data is completely transmitted to the ground station, incurring a communication cost [6]. The architecture breaks down due to the limitations on downlink availability. Compression is useful because it reduces the resources required to store and transmit data. However, computational resources are consumed in compression and decompression. Orbital Edge Computing (OEC) is proposed to reduce the downlink transmission by placing computing resources at the LEO satellites constellation [7]. A promising cloud-native satellite operates by capturing an image and processing it locally instead of transmitting it to the ground completely. Though the satellite with computing capability has been developed [8, 9], satellite-ground collaborative intelligence is still required as follows: it is difficult for satellites to perform complex tasks independently, despite their limited computing capability; the models are one-size-fits-all for all scenarios because satellite data are inconsistently in spatial and temporal distribution. Based on these reasons, satellite-ground collaborative intelligence still plays a vital role now.

The traditional LEO satellite constellations supplement the terrestrial network and support 5G/6G communication, etc [10]. The satellites are customized for specific tasks, which will inevitably lead to the incompatibility of software and hardware. For this issue, domestic and foreign satellite manufacturers such as SmartSat [11] in the United States and TIANZHI 1 [12] in China propose software-defined satellites. Software-defined satellites support plug-and-play payloads, and the application software is loaded on demand to flexibly adapt to multi-tasking and multi-user scenarios [13]. However, it is still challenging to meet the demand of real-time deployment and continuous update of in-orbit services. We first introduce cloud-native technology to satellites to deal with these challenges, because cloud-native technology with container and microservice empowers satellites to build and run the scalable application in a modern, dynamic environment. These techniques enable loosely coupled systems that are resilient, manageable, and observable. Motivated by advanced techniques, cloud-native satellites address AI tasks through distributed cloud-edge collaboration paradigms (e.g., collaborative inference and federated learning, etc) to reduce deployment cost, improve performance and protect

privacy.

Cloud-native satellite technologies show great promise for in-orbit AI applications when the real-world deployment brings forth a set of unique challenges, as shown in Figure 1. One such challenge is the lack of flexibility in existing onboard inference architectures, which hinders the efficient deployment and management of in-orbit applications. Additionally, these architectures suffer from low accuracy levels that fail to meet the demanding requirements of such applications. To address these challenges, we propose a cloud-native-based collaborative intelligence architecture designed to cater to various in-orbit AI applications, including CNN-based image classification, object detection, and segmentation. By adopting this architecture, we aim to enhance the adaptability and accuracy of in-orbit AI applications, paving the way for their successful implementation in a real-world context.

We introduce a case study of satellite-ground collaborative inference system deployed in the experimental satellites of Tiansuan constellation, Baoyun and Chuangxingleishen. Our system has yielded a remarkable 50% improvement in in-orbit inference accuracy and a substantial 90% reduction in the amount of data returned by the satellite. However, in-orbit processing must be compatible with the space environment, which poses challenges due to the higher cost and lower capabilities of space-based systems compared to corresponding COTS units on the ground. Furthermore, we report our findings regarding the energy consumption of in-orbit computing, which accounts for approximately 17% of overall energy consumption. These observations have important implications for future satellite designs and energy management strategies. Our work represents a significant step towards realizing the potential of cloud-native satellites for improving the efficiency and accuracy of in-orbit computing.

The main contributions of this paper as follows:

- 1) We propose the concept of cloud-native satellites to cater to in-orbit computing and facilitate promising applications.
- 2) We present a case study of a satellite-ground collaborative inference system deployed in Tiansuan constellation.
- 3) We report that the inference system not only en-

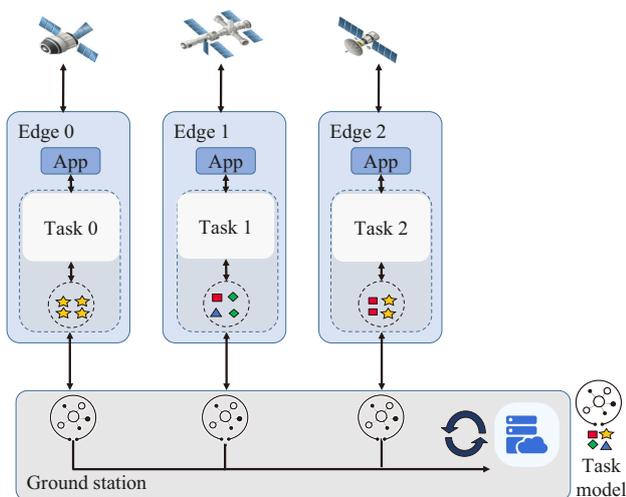


Figure 1. The framework of in-orbit AI applications. Cloud-native satellite can support diverse tasks.

hances the accuracy of in-orbit computing but also effectively reduces communication costs. Additionally, we provide insights into the energy consumption associated with in-orbit computing.

The rest of this paper is organized as follows. Section II introduces the background and motivation of the paper. Section III describes the overall system architecture and main components. Then Section IV details how the satellite-cloud collaborative inference works. Finally, Section V concludes the paper and discusses the potential directions.

II. BACKGROUND & MOTIVATION

Satellite data is exploding. It's reported that about 45% of the LEO satellites in orbit are used for earth observation. The data amount obtained by onboard cameras is too large and 60% of remote sensing images are highly similar in the same scenario [14]. For example, ZY-3 is the Chinese first civilian high-precision cartographic satellite, and it generates more than 10 of TB data every day [15]. One single remote sensing image dataset can reach tens of GB [16]. Such large data transmission to the ground station may take up most of the time and induce huge communication overhead under the bent-pipe architecture [17]. It is a fact that not all raw observations are worth exploring further. Especially in the southwest of China, 80%-90% of raw data is invalid due to cloud cover. There has been preliminary research in the field of onboard

image preprocessing for remote sensing satellites. Before satellites backhaul, redundant information such as cloud cover area can be eliminated in advance and the data returned can be greatly reduced.

Downlinks can be unreliable. Existing systems under a bent-pipe architecture stem from fundamental physical constraints simply downlink raw observations to ground stations. Surprisingly, one satellite task lost 80% of its data packets due to downlink instability [18]. The time-varying relationship between the orbital position of the satellite and the geographic location of ground stations imposes limitations on link availability and can incur high downlink latency between data collection and processing. Any viable orbital system must directly address the unique physical constraints. As an alternative paradigm, OEC collocates processing hardware with high-datarate sensors in low-cost satellites.

Onboard processing is on the rise. With the development of onboard chip integration, the processing performance of satellites has been improved. Onboard systems have integrated multiple computing modules such as CPUs, GPUs, DSPs and FPGAs [19]. Different from traditional CPUs, DSPs and GPUs have stronger computing capabilities. For example, based on embedded cluster computing, the next-generation onboard computer makes a system architecture that combines the master nodes (e.g., CPUs) and slave nodes (e.g., GPUs, FPGAs and DSPs) for parallel processing [20]. These processing units can work together to provide higher onboard computing and improve reconfiguration capabilities. In a word, the enhancement of onboard computing processing capability lays the foundation for OEC.

Enhancing OEC Landing with Cloud-Native Technology. While computing resources at the LEO satellite constellation enables OEC, meeting the demand for flexible deployment in orbit remains a significant challenge. Cloud-native techniques with container and microservice empowers satellites to build and run scalable applications in a modern, dynamic environment. These techniques facilitate the development of loosely coupled systems that exhibit resilience, manageability, and observability. Cloud-native satellite overcomes challenges for unprecedented success in real-world deployment.

Previous works related to this topic in the network/system community have been proposed. Denby

et al. [6] propose an orbital edge computing architecture on satellite constellations to reduce system edge processing latency. Vasisht et al. [21] present a scheduler for multi-satellite, multi-ground station configuration to reduce data downlink latency. Jinhyun et al. [22] focus on the federated learning framework, which dynamically schedules model aggregation based on the deterministic and time-varying connectivity according to satellite orbits. [23, 24] also present challenges of space-air-ground integrated network architecture. [25, 26] discuss opportunities and potential application areas for intelligence in space systems. Our work is orthogonal to and compatible with those above works. To our knowledge, we present a novel approach to achieve high inference accuracy through satellite-ground collaborative intelligence using cloud-native satellites. Our work marks the effort in harnessing the full potential of cloud-native satellite, thereby paving the way for enhanced efficiency and accuracy in in-orbit computing.

III. DESIGN OF CLOUD-NATIVE SATELLITES

With the expansion of LEO constellations and the explosive growth of satellite data, it is urgent to explore how to effectively utilize local satellite data. Additionally, local processing of satellite data is promising as AI capabilities migrate from the cloud to edge nodes [27]. However, each satellite has limited computing capability and energy resources, making it difficult to perform complex tasks independently. Satellite-ground collaboration is proposed to facilitate intelligent in-orbit computing. Also, motivated by cloud-native, we first provide an overview of an individual cloud-native satellite. We then describe key techniques involved in cloud-native satellite, which organizes a system that supports in-orbit computing.

3.1 Cloud-Native Satellite

As the first cloud-native satellite in the world, the satellite leverages advanced techniques such as containers and microservices to realize in-orbit computing. As shown in Figure 2, we integrate communication, computing, and storage capabilities into cloud-native satellites. The satellite provides open interfaces that allow third parties to deploy applications. The

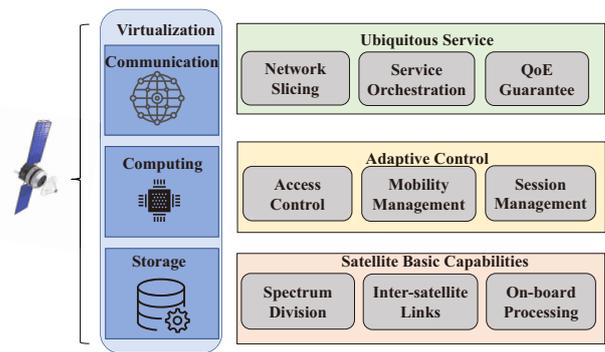


Figure 2. The design of the cloud-native satellite. The communication, computing, and storage capabilities are integrated into the cloud-native satellite.

satellite also monitors and manages the operational status and applications. Based on the storage capacity, computing module provides onboard control capabilities for basic and third-party ubiquitous applications. Cloud-native satellites guarantee in-orbit service execution that mainly benefits from:

Rapid deployment Cloud-native satellite leverages agile development and DevOps project management models [28]. It not only enables users to deploy applications quickly and automatically but also continuously updates onboard applications to keep up with changes in demand.

Elastic expansion The satellite manages satellite-ground clusters by automatically increasing or decreasing edge nodes. It also shrinks or expands resources according to the real-world situation to improve the utilization of onboard resources.

Highly autonomous Based on container technology, cloud-native satellite encapsulates service functions to achieve isolation and abstraction independent of location and environment. Moreover, EdgeMesh, as the data plane component of the KubeEdge cluster, provides simple service discovery and traffic proxy functions for satellite service, thereby shielding the complex network structure in edge scenarios [29].

Safe and reliable Container orchestration is crucial for large and dynamic production environments [8]. With container orchestration for microservices, onboard applications can be automatically scaled, fault-tolerant which copes with the complex environment of space and keeps onboard applications available at all times.

The emergence of cloud-native satellites bene-

fits from the application management framework in which satellites and the ground cloud work together. KubeEdge [30] and its sub-project Sedna [31] provide unified resource and application management for cloud-native satellites and improve application collaboration capabilities. The system architecture is shown in Figure 3. Collaborative AI framework breaks model training and inference space constraints. In the following, we mainly introduce the two major frameworks: KubeEdge and Sedna.

3.2 KubeEdge

KubeEdge is an open-source platform based on Kubernetes that provides infrastructure support for applications by extending containerized application orchestration to satellites [30]. KubeEdge also synchronizes deployments and metadata between central cloud and satellite, which strongly supports:

Extremely lightweight The global resources of the satellites are discrete and the local resources are limited to one satellite. The instability of the edge network leads to frequent disconnection of edge satellites and management systems. The platform extracts the core functions of Kubelet and builds a lightweight management software named EdgeCore to cover different scenarios. EdgeCore and CloudCore in the central cloud provide data centers to meet exacting requirements without compromise [32].

Reliable connection The network between satellites and ground station often suffers from low bandwidth and serious packet loss. The platform manages edge-cloud messages in the same way, and the data is still reliably transmitted in weak network scenarios.

Offline autonomous A lightweight management component named MetaManager stores metadata [33]. When edge nodes go offline, applications are managed and restored based on storage metadata. This method not only brings less resource overhead but also ensures the security of other satellite nodes in the cluster.

Cloud-edge collaboration Edge and cloud always work together to achieve end-to-end applications. KubeEdge provides an edge-cloud data exchange mode. More importantly, EdgeMesh provides unified service discovery and traffic proxying between microservices. For example, the capabilities of EdgeMesh-Server are merged into the tunnel module of EdgeMesh-Agent, so that EdgeMesh-Agent

with relay capability can automatically become a relay server, providing other nodes with the functions of assisting hole punching and relaying.

3.3 Sedna

Sedna's edge-cloud synergy is implemented for collaborative inference based on the capabilities provided by KubeEdge. The component Global Manager in Figure 3 unifies satellite-ground synergy AI task management and collaboration by well-supporting current popular frameworks including TensorFlow/ Pytorch/ PaddlePaddle/ MindSpore, etc [31]. Sedna helps AI applications migrate seamlessly to satellites by using cloud-edge collaboration protocols to reduce cost, improve model performance, and protect data privacy, etc. In the following section, we discuss the major components of Sedna.

Global Manager This component implements an edge AI controller based on Kubernetes. Users create CRD [34] to achieve model/dataset management, AI task management for edge-cloud collaboration, and edge-cloud coordination.

Local Controller This component performs local process control of edge-cloud collaborative AI tasks. Moreover, it also performs local general management: models, datasets, state synchronization of AI tasks, etc.

Worker This component performs AI tasks based on the training/ inference procedures of existing AI frameworks. Workers can be deployed on the edge or in the cloud and they work together with each other.

Lib This component provides an edge-cloud collaboration interface for applications. Users can implement training/ inference and aggregation based on this Lib.

3.4 Promising Application Scenarios

To address different tasks and challenges, cloud-native satellites can support of diverse paradigms. Currently supported computing protocols are as follows:

Collaborative inference Satellites cannot independently perform complex inference tasks because of limited resources and weak computing capability on the satellite. The satellite performs one part of the whole inference and offloads some tasks to the cloud to improve the overall inference performance.

Incremental training Changes in data distribution,

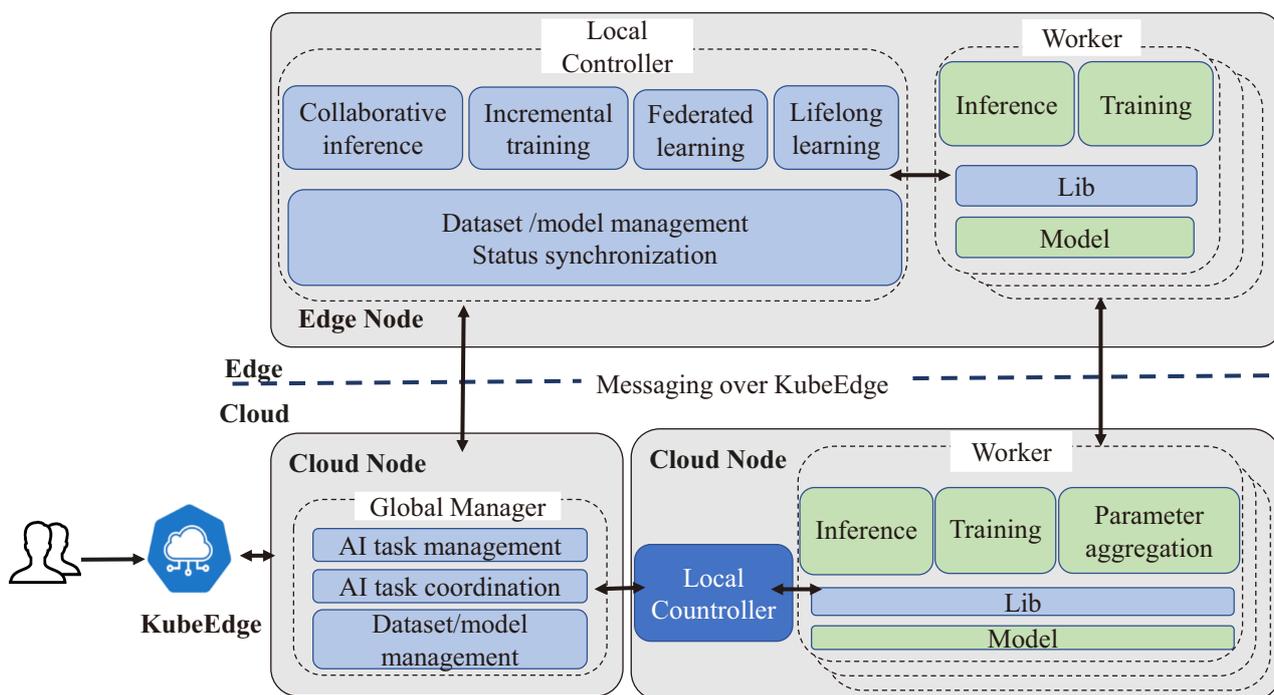


Figure 3. The workflow of satellite-ground collaborative inference system architecture.

such as weather, make the deployed model unable to adapt to the current environment and the detection accuracy decreases. For this challenge, satellites continuously collect newly generated data and train models in the cloud. The satellite nodes regularly fine-tune the model from the cloud to improve accuracy.

Federated learning From the perspective of privacy protection, federated learning can effectively alleviate data silos because raw data is reluctant to share to the cloud. The satellite trains the model and transmits the parameters (i.e., training weights) to the cloud responsible for parameter aggregation. The system also provides a secure data channel through KubeEdge and supports encrypted transmission to ensure data security. Such a protocol may train high-precision models for privacy protection.

Lifelong learning Satellites suffer from data drift and catastrophic forgetting of onboard models. Combining incremental training and multi-task training, satellite model enables knowledge transfer across time and scenarios. Based on the knowledge library in the cloud, satellite model can be continuously updated to address unknown tasks.

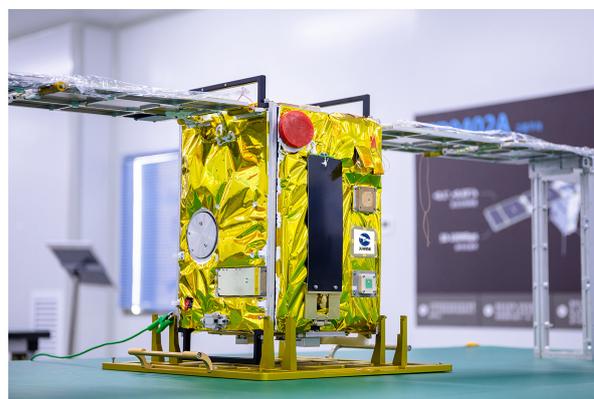


Figure 4. A physical appearance of the Baoyun satellite.

IV. CASE STUDY: THE FIRST IN-ORBIT AI COLLABORATIVE INFERENCE

We've successfully launched two new satellites, Baoyun and Chuangxingleishen, and implemented collaborative inference on these satellites using KubeEdge and Sedna. Earth Observation satellites incorporate cloud-native computing capabilities. Table 1 shows the real-world specifications including vital parameters such as mass, size for both satellites. Additionally, Figure 4 provides a physical appearance of Baoyun satellite. It's worth noting that these satel-

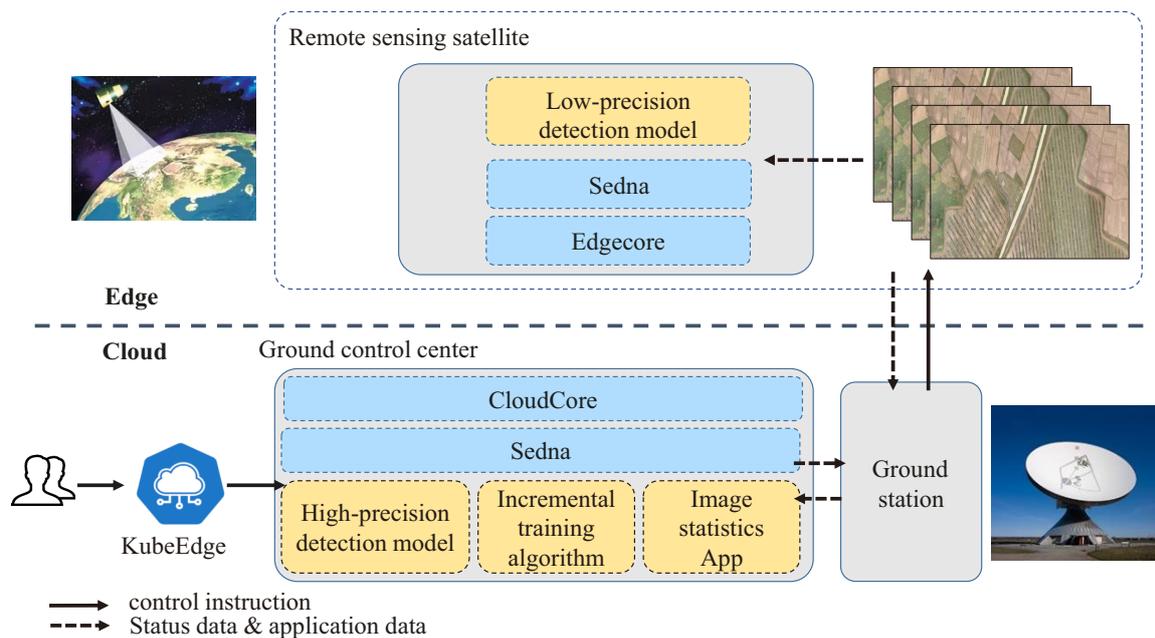


Figure 5. The workflow of satellite-ground collaborative inference system architecture.

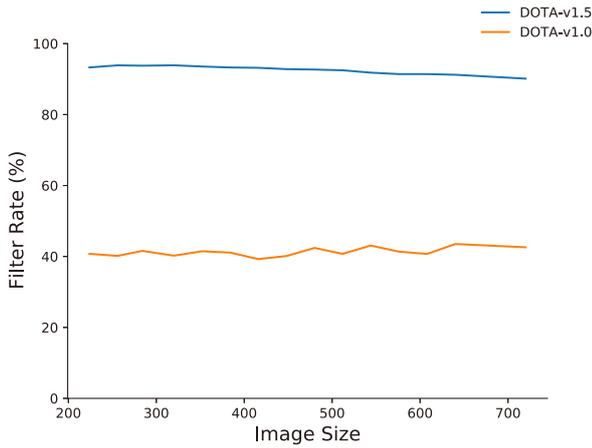
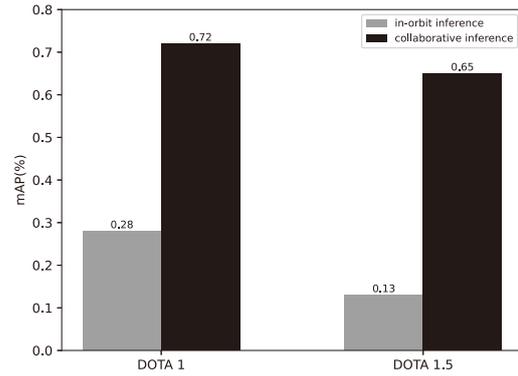
lites adhere to the CubeSat standard for design. The control center operates on a Linux server, sending remote instructions to the satellites. We've deployed two different precision detection models, consisting of lightweight, low-precision models onboard the satellite for preprocessing to obtain preliminary results. On the ground, large, high-precision models are situated with high computing capabilities, which facilitate accurate detection. The workflow of our collaborative inference system is shown in Figure 5. The satellites perform a preliminary detection using a lightweight model to identify features of interest. If confidence threshold in the results is high, the processed results are sent back to the ground directly. However, if confidence threshold is low, the satellite transmits the images to the ground, where the high-precision detection model is used for exact detection. This system enables efficient detection of captivating images and information, ensuring that only essential data is transmitted to the ground. Our experiments are based on the fact that onboard inference can be performed anytime, when satellite computation are available. The handover between them only occurs during the contact time between the satellite and the ground. Note that the shown evaluations are conducted in Baoyun satellite. Interestingly, the experimental results on different satellites show that the inference system can achieve consistent

results on satellites of different specifications.

Onboard image splitting significantly reduces the occurrence of redundant images caused by cloud cover. We conducted experiments on the widely-used DOTA [35], object detection dataset. We propose a strategy to split the images into smaller images before performing in-orbit inference. This approach is necessary due to the limited computing power of the satellite, which cannot handle high-resolution images. Figure 6 illustrates the reduction in redundant images during in-orbit operations. We observe that splitting the large images into smaller fragments, irrespective of the fragment size, leads to a remarkable 90% and 40% decrease in images for the two versions of the datasets, respectively. As a result, in-orbit processing approach offers the dual benefits of reducing the number of images and conserving significant bandwidth. Collaborative inference yields a significant improvement in accuracy. Our satellite-ground collaborative inference system incorporates YOLOv3-tiny and YOLOv3 object detection models for onboard and ground detection, respectively. To assess the accuracy of our system, we use the mean average precision (mAP) metric, which compares ground-truth bounding boxes to detected boxes and returns a score. A higher score indicates more accurate detection [36]. We analyzed the accuracy of our system through in-orbit and collabora-

Table 1. Satellite platform specifications.

Name	Launch Time	Orbital Altitude (km)	Mass (kg)	Load Size (U)	Size (U)	Operating System	Uplink Rate (Mbps)	Downlink Rate (Mbps)
Baoyun	Dec. 7 2021	500±50	20	0.25	12	Ubuntu Server 20.04 arm	0.1~1	≥40
Chuangxing leishen	Feb. 27 2022	500±50	20	0.25	6	Debian Buster with Raspberry Pi	0.1~1	≥40

**Figure 6.** The filter rate of redundant data in orbit on DOTA.**Figure 7.** Accuracy (mAP in object detection task) of in-orbit vs. collaborative inference.

tive inference methods, as shown in Figure 7. The results demonstrate a remarkable accuracy enhancement of 44% and 52% respectively when employing collaborative inference compared to in-orbit inference. On average, our collaborative inference system achieves an approximate 50% improvement in accuracy. Additionally, a significant number of redundant images were detected. By performing preprocessing onboard and directly transmitting the inference results to the ground station, our system successfully reduced the amount of data returned by 90%.

According to the energy consumption analysis presented in Table 2, Baoyun satellite's energy system primarily powers its payloads, accounting for approximately 53% of the total energy consumption. The remaining energy is utilized for basic satellite functions such as communication and propulsion. Further delving into the energy consumption of the satellite's payloads, Onboard equipment measures the voltage and current of each power system and records the telemetry data, which is then transmitted to the ground. The reported power consumption is recalculated based on the collected data. Table 3 presents a breakdown

of the subsystems responsible for satellite computing and their corresponding power consumption, including the camera that captures remote sensing images. Among these power subsystems, the Raspberry Pi system accounts for the largest proportion of energy consumption, representing 33% of the total energy consumed by the payloads. In summary, computing on the Baoyun satellite accounts for about 17% of energy consumption. The presented data sheds light on the energy allocation and consumption patterns of the satellite, offering value for optimizing its operational efficiency and ensuring its long-term sustainability.

V. CONCLUSION

This paper introduces the emergence of a cloud-native satellite that directly leverages onboard computing for remote sensing tasks. It explores various execution protocols such as collaborative inference, federated learning, incremental learning, and lifelong learning. Additionally, we launched the first cloud-native satellite and implemented the collaborative inference system on the satellite, with the ultimate objective of achieving high inference accuracy through the use of

Table 2. Real power distribution of energy consumption system in Baoyun satellite.

Item	Electrical Subsystem	Propulsion	Guidance	Avionics	Comm.	Payloads	Sum
Power(W)	1.47	7.00	5.43	4.81	5.43	26.93	51.07

Table 3. The power of payloads subsystem of Baoyun satellite.

Item	Camera	Occultation	Tribology	Mems	Adsbs	Raspberry Pi
Power(W)	0.09	6.26	5.68	0.95	6.12	8.78

cloud-native satellites.

In the future, we plan to deploy this project in batches across other satellites of Tiansuan constellation, thereby establishing a collaborative computing network in space. We also expect cloud-native satellites to unlock new capabilities in precision agriculture, infrastructure monitoring, humanitarian assistance, etc [37].

VI. ACKNOWLEDGEMENT

This work was supported by National Natural Science Foundation of China (62032003).

References

- [1] "Cisco annual internet report (2018–2023) white paper," <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>, 2020.
- [2] Y. Shi, J. Liu, *et al.*, "Jamming-aided secure communication in ultra-dense leo integrated satellite-terrestrial networks," *China Communications*, 2023, vol. 20, no. 7, pp. 43–56.
- [3] "SpaceX starlink." <https://www.spacex.com/webcast.>, 2017.
- [4] "Telesat: Global satellite operators." <https://www.telesat.com/>, 2020.
- [5] "Amazon to offer broadband access from orbit with 3,236- satellite project kuiper constellation." <https://www.geekwire.com/>, 2019.
- [6] B. Denby and B. Lucia, "Orbital edge computing: Nanosatellite constellations as a new class of computer system," in *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, 2020, pp. 939–954.
- [7] Z. Zhang, W. Zhang, *et al.*, "Satellite mobile edge computing: Improving qos of high-speed satellite-terrestrial networks using edge computing techniques," *IEEE Network*, 2019, vol. 33, no. 1, pp. 70–76.
- [8] Z. Zhong, M. Xu, *et al.*, "Machine learning-based orchestration of containers: A taxonomy and future directions," *ACM Computing Surveys (CSUR)*, 2022.
- [9] Y. Sun and M. Peng, "Edge intelligence assisted resource management for satellite communication," *China Communications*, 2022, vol. 19, no. 8, pp. 31–40.
- [10] M. Sheng, D. Zhou, *et al.*, "6g service coverage with mega satellite constellations," *China Communications*, 2022, vol. 19, no. 1, pp. 64–76.
- [11] "Smartsat." <https://www.smartsat.com/>, 2022.
- [12] "Tianzhi 1." https://space.skyrocket.de/doc_sdat/tianzhi-1.htm, 2018.
- [13] J. Bao, B. Zhao, *et al.*, "Opensan: A software-defined satellite network architecture," *ACM SIGCOMM Computer Communication Review*, 2014, vol. 44, no. 4, pp. 347–348.
- [14] R. Avtar, A. P. Yunus, *et al.*, "Multi-temporal remote sensing data to monitor terrestrial ecosystem responses to climate variations in ghana," *Geocarto International*, 2022, vol. 37, no. 2, pp. 396–412.
- [15] "Zy-3 01, 02, 03, 04." https://space.skyrocket.de/doc_sdat/zy-3.htm, 2022.
- [16] "Processing, challenges and thinking of earth observation big data." http://www.sinoprobe.org/news_show.aspx?id=1811, 2022.
- [17] J. Park, S. Samarakoon, *et al.*, "Wireless network intelligence at the edge," *Proceedings of the IEEE*, 2019, vol. 107, no. 11, pp. 2204–2239.
- [18] C. Nogales, B. Grim, *et al.*, "Makersat-0: 3d-printed polymer degradation first data from orbit," 2018.
- [19] K. Schwenk, M. Ulmer, *et al.*, "Scosa: Application development for a high-performance space qualified onboard computing platform," in *High-Performance Computing in Geoscience and Remote Sensing VIII*. SPIE, 2018, vol. 10792, pp. 32–44.
- [20] D. Lütke, K. Westerdorff, *et al.*, "Obc-ng: Towards a reconfigurable on-board computing architecture for spacecraft," in *2014 IEEE Aerospace Conference*, 2014, pp. 1–13.
- [21] D. Vasisht, J. Shenoy, *et al.*, "L2d2: Low latency distributed downlink for leo satellites," in *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*, 2021, pp. 151–164.
- [22] J. So, K. Hsieh, *et al.*, "Fedspace: An efficient federated learning framework at satellites and ground stations," *arXiv preprint arXiv:2202.01267*, 2022.
- [23] H. Cui, J. Zhang, *et al.*, "Space-air-ground integrated net-

- work (sagin) for 6g: Requirements, architecture and challenges,” *China Communications*, 2022, vol. 19, no. 2, pp. 90–108.
- [24] Z. Xiao, J. Yang, *et al.*, “Leo satellite access network (leosan) towards 6g: Challenges and approaches,” *IEEE Wireless Communications*, 2022.
- [25] N. Kato, Z. M. Fadlullah, *et al.*, “Optimizing space-air-ground integrated networks by artificial intelligence,” *IEEE Wireless Communications*, 2019, vol. 26, no. 4, pp. 140–147.
- [26] G. Furano, G. Meoni, *et al.*, “Towards the use of artificial intelligence on the edge in space systems: Challenges and opportunities,” *IEEE Aerospace and Electronic Systems Magazine*, 2020, vol. 35, no. 12, pp. 44–56.
- [27] Q. Zhang, S. Wang, *et al.*, “Cost-aware edge server placement,” *International Journal of Web and Grid Services*, 2022, vol. 18, no. 1, pp. 83–98.
- [28] A. Damiani, G. Fiscoletti, *et al.*, “Blastfunction: A full-stack framework bringing fpga hardware acceleration to cloud-native applications,” *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, 2022, vol. 15, no. 2, pp. 1–27.
- [29] “Edgemesh.” <https://github.com/kubeedge/edgemesh>, 2022.
- [30] “Kubeedge.” <https://github.com/kubeedge>, 2022.
- [31] “Sedna.” <https://github.com/kubeedge/sedna>, 2022.
- [32] “Edgecore.” <https://www.edgecore.com/>, 2022.
- [33] M. Rani and K. Ahuja, “Interface management in multi-interface mobile communication: A technical review,” *International Journal of System Assurance Engineering and Management*, 2022, pp. 1–13.
- [34] “Extend the kubernetes api with customresourcedefinitions.” <https://kubernetes.io/docs/tasks/extend-kubernetes/custom-resources/custom-resource-definitions/>, 2022.
- [35] “Dota.” <https://captain-whu.github.io/DOTA/dataset.html>, 2017.
- [36] S. Ren, K. He, *et al.*, “Faster r-cnn: Towards real-time object detection with region proposal networks,” *Advances in Neural Information Processing Systems*, 2015, vol. 28.
- [37] S. Lang, P. Füreder, *et al.*, “Earth observation tools and services to increase the effectiveness of humanitarian assistance,” *European Journal of Remote Sensing*, 2020, vol. 53, no. sup2, pp. 67–85.

Biographies



Wang Shangguang is a professor at the School of Computer Science, Beijing University of Posts and Telecommunications, China. He is the founder&chief scientist of Tiansuan Constellation. He is also the director of Star Network and Intelligence Computing Laboratory, and vice director of Sate Key Laboratory of Networking and Switching at BUPT. His research interests include service computing, mobile edge computing, cloud computing, and satellite computing. He is currently

serving as chair of IEEE Technical Committy on Services Computing (TCSVC), and vice chair of IEEE Technical Committee on Cloud Computing. He also served as general chairs or program chairs of 10+ IEEE conferences, advisor/associate editors of several journals such as *Journal of Cloud Computing*, *Journal of Software: Practice and Experience*, *International Journal of Web and Grid Services*, *China Communications*, and so on. He is a senior member of the IEEE, and Fellow of the IET.



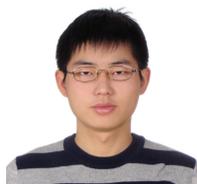
Zhang Qiyang received the bachelor’s degree in Network Engineering from Henan Normal University in 2018. Currently, he is a Ph.D candidate in computer science at the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. His research interests include satellite edge computing and edge intelligence.



Xing Ruolin received the bachelor’s degree from Beijing University of Posts and Telecommunications in 2019. Currently, he is a Ph.D candidate in computer science at the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. His research interests include satellite edge computing and edge intelligence.



Qi Fei is the Principal Engineer of Huawei Cloud Computing Technologies and the team leader of Huawei edge-cloud collaboration OS technology innovation team. He joined Huawei in 2012, and he has been engaged in the research of cloud computing systems and key technologies for a long time, and has participated in the innovation and incubation of multiple cloud computing products such as Huawei hybrid cloud and edge cloud.



Xu Mengwei received the bachelor’s and Ph.D degrees from Peking University, Beijing, China. He is an Assistant Professor with the Computer Science Department, Beijing University of Posts and Telecommunications, Beijing. His research interests cover the broad areas of mobile computing, edge computing, and operating systems.