

# Collaborative Hyperspectral Image Processing using Satellite Edge Computing

Botao Zhu, Siyuan Lin, Yifei Zhu, *Member, IEEE*, and Xudong Wang, *Fellow, IEEE*

**Abstract**—The advancement of nanosatellite techniques has boosted the growth of satellite-originated data and applications. Satellite edge computing (SEC) is envisioned to provide in-orbit processing of the sensed data to save the scarce terrestrial-satellite communication resources and support mission-critical services. While most of the existing SEC studies mainly focus on general computing tasks, we present a two-tier collaborative processing framework for the important and unique hyperspectral image (HSI) processing task. Our framework carefully selects bands out of the collected HSIs and sends them back for further analysis. We first conduct a comprehensive data analysis to reveal the non-trivial relationship between the band selection and the eventual analytic performance. We then formulate the band selection problem in this collaborative setting as a utility maximization problem that jointly considers the analytic, energy, and communication factors. A novel multi-agent reinforcement learning approach, named MaHSI, is proposed to solve it in the dynamic SEC environment. Our multi-agent design judiciously embeds the complex correlations among bands as collaborations among agents and significantly reduces the exploration space. Extensive experiments on real-world HSI datasets prove that our approach not only outperforms the existing classical band selection algorithms in accuracy and inference speed but also brings the highest utility to the satellites.

**Index Terms**—satellite edge computing, collaborative computing, multi-agent reinforcement learning, hyperspectral image processing, band selection

## 1 INTRODUCTION

OVER the past two decades, low earth orbit (LEO) nanosatellite launches have grown significantly due to the resurgence of the space industry and declining cost of accessing space [1]. The number of LEO satellites has reached nearly 4000 and is expected to grow 110% by 2025 [2] [3]. These satellites play an integral part in the next-generation wireless networks and continuously provide networking or monitoring services. Currently, about 45% of LEO satellites conduct earth observing tasks [2]. Among these tasks, the hyperspectral imaging task is one of the most important. Hyperspectral imaging is a remote sensing technique that collects the electromagnetic spectrum from the visible to the near-infrared wavelength ranges, usually from 400 to 2500 nm. The generated hyperspectral image (HSI) is a 3D image cube that contains hundreds of bands at different wavelengths along with the usual spatial information. With the resourceful spectral and spatial information captured, HSIs have been widely applied in fields such as agriculture [4], mineral mapping [5], and landscape change detection [6]. In particular, HSIs are increasingly deployed for mission-critical applications that require in-time data processing, e.g., forest fire detection [7] and gas leakage detection [8].

However, there exists a huge gap between the amount of data generated and the amount of data that can be transmitted. The current satellites adopt a bent-pipe archi-

ture to communicate with the ground station [9]. In this architecture, ground stations send commands to satellites, and satellites reply with raw data. For any ground station observer, an LEO satellite usually has 3 to 5 good contacts every day with the ground station; each contact lasts for around 10 minutes [10]. Compared to the traditional monolithic satellites, nanosatellites have even scarcer energy and communication capabilities. A limited-sized nanosatellite can only harvest 7.1W at peak time from the onboard solar panel [11]. As for communication, existing hyperspectral imaging nanosatellites can only allow a maximum 7.5 Mbps data rate for downlink transmission [12]. In contrast, a single high-resolution HSI can even reach more than 1 GB per scene [13]; hundreds of Gigabytes of data can be generated every day by a satellite [10].

Constrained by the limited energy and communication resources, and driven by the demand for in-time intelligence extraction from the growing mission-critical applications, satellite edge computing has started to gain researchers' attention recently. For satellite edge computing, computing power is deployed on a satellite, so that data can be processed locally on the satellite, and only the extracted information is transmitted to the ground station [11] [14] [15] [16]. For example, GomX-4B launched in 2018 carries CPU, GPU, and VPU to provide in-orbit data processing [12]. While it is desirable to in-orbit process hyperspectral image, hyperspectral image processing tasks differ from ground-based image processing tasks in that it is fully satellite-originated and exhibit unique characteristics. First, hyperspectral images have far more channel information than classical RGB images. A typical hyperspectral image can have up to 200 channels, whose high dimensions make it impractical to directly apply classical machine learning approaches. Second, both spatial and spectral features of HSI

- B. Zhu, S. Lin, and X. Wang are with the UM-SJTU Joint Institute, Shanghai Jiao Tong University, China. Y. Zhu is with the UM-SJTU Joint Institute, Shanghai Jiao Tong University, China, and Cooperative Medianet Innovation Center (CMIC), Shanghai Jiao Tong University, China.  
Email: {zhubotao, Jeremy.l, yifei.zhu, wxudong}@sjtu.edu.cn,  
Corresponding author: Yifei Zhu

need to be considered for accurate analysis. Furthermore, a strong correlation exists among the spectral bands. Third, satellite communication differs from current on-ground cellular communications in that it is essentially intermittent, and highly emphasizes energy consumption.

Due to the overwhelming band information, after transmitted to the ground, the raw data usually go through another dimensionality reduction process, like band selection [17] [18], to reduce the dimensions so that the downstream analytic tasks are possible. In band selection operations, a set of representative bands are selected to include most information of the original band set. Compared with other feature extraction techniques that are based on complex data transformation, band selection preserves the main semantic information in an HSI without losing its spectral property. With the rise of satellite edge computing and inspired by the great benefits of band selection in data size reduction, it seems natural and promising to move this *previously ground-based* pre-processing operations to the satellite, so that informative bands can first be selected and more hyperspectral images can be transmitted to the ground within the same communication time.

However, simply applying traditional band selection methods in satellites is not desirable for two main reasons. First, the traditional band selection methods still adopt a simple band selection strategy and only select a *fixed number* of bands to *preserve fidelity* as much as possible. However, our analysis of the real-world HSI dataset in the following section reveals that different categories of samples have different band selection patterns to reach the optimal *analytic* performance. Choosing an appropriate band size needs tremendous manual effort and easily leads to sub-optimal analytic performance. Second, traditional ground-based band selection methods take no consideration of the satellite edge computing components, like energy, communication, etc. With fluctuating network conditions and limited satellite-terrestrial contact windows, transmitting too many bands under bad network conditions can cause high packet loss; transmitting too few bands under good network conditions easily results in low network utilization. *Therefore, to fully utilize the valuable in-orbit computing power, a more adaptive and satellite-native terrestrial-satellite collaboration framework has to be devised for hyperspectral image processing.*

In this paper, we present the first work for two-tier hyperspectral image processing based on satellite edge computing and terrestrial-satellite collaboration. We first conduct thorough data analysis to reveal the non-trivial relationship between the band selection in hyperspectral image processing and the eventual analytic performance and identify the characteristics of the complex satellite-terrestrial communication networks. We then systematically model the critical energy and communication factors in satellite edge computing. Based on the novel insights and models, we present our two-tier hyperspectral image processing framework where a subset of informative bands is first selected using satellite edge computing; the resulting bands are then transmitted to the ground station for eventual analysis. The core problem of our framework, the band selection problem, is formally formulated as a utility maximization problem. A novel multi-agent reinforcement learning approach, named MaHSI, is designed to judiciously

incorporate the dependence among bands via interpreting it as a multi-agent collaboration. It leverages a deep Q-learning architecture to learn the optimal state-action pairs, and adaptively make band selection decisions considering the dynamic orbital situations. Extensive experiments based on eight real-world HSI datasets validate the performance of our design.

To better identify the research gaps between the current research efforts and our orbital hyperspectral image processing requirements, as well as our contributions, we further summarize the related work as follows:

## 1.1 Satellite Edge Computing

Denby et al. [11] present novel parallel pipelines for image edge processing on nanosatellites, which dramatically reduces the latency compared to traditional bent-pipe architecture. To adapt to the limited bandwidth and unstable connectivity in space information networks, researchers in [19] presents a distributed machine learning system that applies dynamic model compression techniques. In [20], the suitability of three organization paradigms for applications and unique characteristics for satellite edge are discussed. Researchers in [21] compare the performance between CPU and GPU on in-orbit advanced algorithms for heterogeneous system architecture. Researchers in [16] propose a hybrid model of non-orthogonal multiple access (NOMA) and frequency division multiple access to facilitate the multi-user dual computation offloading. While several studies have started to examine the potential of satellite edge computing, they mostly focus on leveraging computing power on satellites to serve ground applications [19] [20] [21], targeting general computation tasks [11] [16] [22], or directly running ground-based algorithms on board, which leads to sub-optimal performances [14]. Different from them, our work focuses on applying satellite edge computing for a satellite-originated hyperspectral imaging processing task, with unique characteristics. We establish an energy model and a communication model and explore the potential opportunity for energy and bandwidth saving using satellite edge computing with HSI band selection. A multi-agent reinforcement learning approach is further proposed to specifically handle the complex band correlation and large exploration space.

## 1.2 HSI Band Selection

Band selection algorithms can be categorized into unsupervised and supervised methods. In unsupervised methods, clustering-based, searching-based, and ranking-based methods are the mainstream. Clustering-based methods like [17] cluster the spectral bands and select the cluster centers as band selection. Searching-based methods [18] usually apply heuristic methods to search the best band subset. Ranking-based methods [23] try to assign each band a score by estimating their significance. Different from unsupervised methods, supervised band selection methods utilize prior knowledge for band selection. Researchers in [24] present a method for band selection based on the local spatial information and classification label of HSI. Arab et al. in [25] apply reinforcement learning for band selection in a general context. However, their proposed method trains

dedicated RL models for each dataset, which is not practical when new HSIs are continually collected and transmitted in the satellite setting. In addition, traditional band selection methods purely aim at preserving fidelity of HSIs without directly considering downstream analytic tasks. Our proposed method can well handle the combination of analytic performance, network, and energy condition, which is more practical for a satellite edge computing environment. Overall, we also contribute to the band selection problem in the remote sensing field by proposing a novel multi-agent reinforcement learning-based approach that achieves the best end-to-end analytic performance in the space environment.

In summary, our main contributions are summarized as follows:

- We present a novel two-tier collaborative computing framework for the important satellite-originated hyperspectral image processing task.
- In contrast to the classical fidelity-driven band selection studies, we reveal the non-trivial relationship between band selections and eventual analytic performances from a novel analytic perspective. Specifically, different categories of HSIs require different bands; more bands do not necessarily lead to better classification performance.
- We formulate the band selection problem in our framework as a utility maximization problem, that incorporates practical satellite edge system requirements and is proved to be NP-Complete.
- We propose a novel multi-agent reinforcement learning approach that judiciously embeds the complex band correlation relationship and successfully learns wise policy in the large searching space and the dynamic environment. This is also the first work that applies multi-agent reinforcement learning for HSI band selection.
- Extensive experiments on real-world satellite communication and the HSI dataset demonstrate that our MaHSI not only outperforms the existing SOTA band selection algorithms and the full-band strategy in classification accuracy by 20% on average but also achieves the highest HSI utility in the dynamic environment.

The rest of this paper is organized as follows. We first introduce the related work in section II. In section III, we present the data analysis and motivation for this work. Section IV describes the system framework and the formulation of our utility maximization problem. We then introduce our MaHSI method for adaptive HSI processing at the satellite edge in section V. The evaluation for MaHSI is presented in section VI. We conclude our work in section VII.

## 2 DATA ANALYSIS AND MOTIVATION

In this section, we reveal the non-trivial relationship between band selections and analytical results, and the unique satellite-terrestrial downlink characteristics via extensive data analysis.

### 2.1 Analysis of HSIs Band Selection and Classification

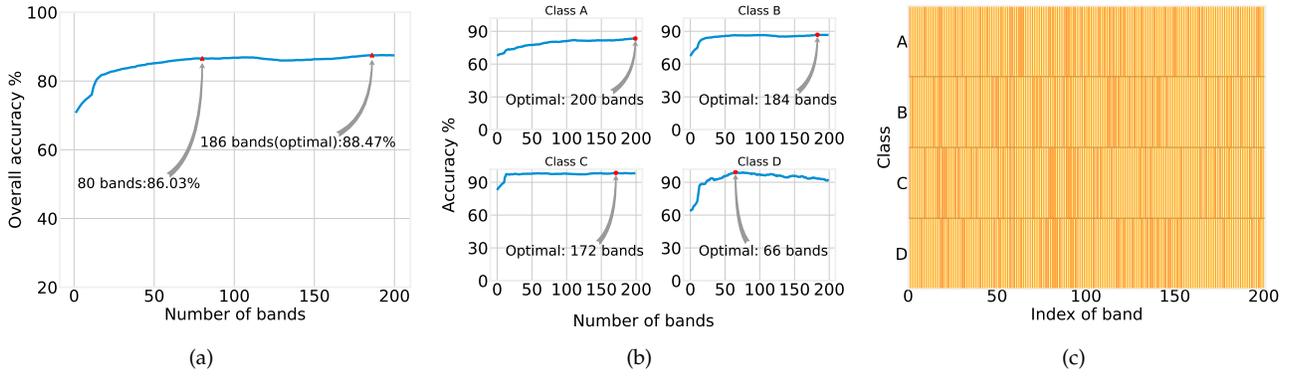
While previous band selection algorithms are developed to preserve the information fidelity as much as possible, in this

part, we examine the end-to-end relationship between band selection and the eventual classification results from *an analytic perspective*. Different from ordinary image classification, hyperspectral image classification assigns individual pixels to a set of classes; it is the fundamental task for a majority of hyperspectral image applications. We choose the classical clustering-based ISSC approach [17] as the band selection algorithm, SVM as our classifier, and five commonly used HSI datasets, Indian Pines, Kennedy Space Center, Pavia University, Salinas Scene [26], and HyRANK Loukia [27] as our dataset. *They are all widely used approaches and datasets in the remote sensing community for hyperspectral image processing.* We use Indian Pines to illustrate our observations due to space limits. Results of other datasets can also be found in the supplementary material, all of which can support our observations here. Indian Pines dataset is an agricultural hyperspectral image dataset captured by the AVIRIS sensor over the Indian Pines test site in Northwest Tippecanoe County, IN, USA. This dataset has 16 classes and 200 bands. Overall accuracy, which is a common metric for HSI classification tasks [28], measures the portion of the number of hyperspectral pixels that are classified correctly out of the number of test samples. We summarize our observations as follows.

**The overall accuracy gain diminishes in general as more bands are selected.** Fig. 15(a) shows the relationship between the overall accuracy and the number of selected bands on the Indian Pines dataset. Generally speaking, the classification performance increases gradually as more informative bands are selected. However, the increasing speed drops quickly after we select a certain number of bands. It takes 186 bands to reach the optimal performance of 88.47 % as annotated in Fig. 15(a), while we can already achieve a close to optimal performance, 85.05%, with about 40 bands. It indicates that we can reduce nearly 80 % of the transmission amount at the cost of only about 3% accuracy loss. This discovery provides us with a valuable chance to make a trade-off between classification performance and cost-saving when running in-orbit HSI processing.

**The optimal number of bands varies across different HSI categories.** Though we have presented the general impact of the band sizes on the overall accuracy, an HSI dataset commonly contains imbalanced classes, which makes the overall classification result skew towards the result of a few dominant classes. To comprehensively examine the effect of band sizes on different HSI categories, we then perform band selection on all 16 classes of Indian Pines separately. The results of all classes are listed in Appendix B in the supplementary material. Here we select four representative classes, Grass Pasture, Grass Tree, Grass Pasture Mowed, and Hay Windrowed, for illustration purposes. As shown in Fig. 15(b), the optimal band number for different classes, are quite different. The classification performance of classes like Hay Windrowed reaches the maximum at 66 bands and begins to drop after that. On the contrary, the classification performance of Grass Pasture keeps increasing until the full bands are selected.

**Different classes have different optimal band selections.** We further examine the exact band selected by the band selection algorithm given a target band size. In Fig. 15(d), we visualize the optimal 40 bands for the above-



**Fig. 1:** Anatomy of the classification result on Indian Pines. (a) The relationship between the number of selected bands and the overall accuracy. (b) The relationship between the number of selected bands and the accuracy of four classes in Indian Pines. (c) 40 optimal bands in shaded color for 4 classes in Indian Pines. Class A: Grass Pasture, B: Grass Trees, C: Grass Pasture Mowed, D: Hay Windrowed.

mentioned 4 classes. The band being selected is shaded. We can see that the optimal band subset varies across the selected 4 classes. Though classes may have some shared optimal bands, for example, the algorithm selects bands 1, 10, 18, and 58 for both Grass Trees and Hay Windrowed, a universal band subset that suits well for all the classes hardly exists.

From these results, we can observe the non-trivial relationship between the band selection and the eventual classification performance, which can hardly be summarized into fixed rules. This motivates a more intelligent band selection algorithm with intrinsic HSI content in consideration.

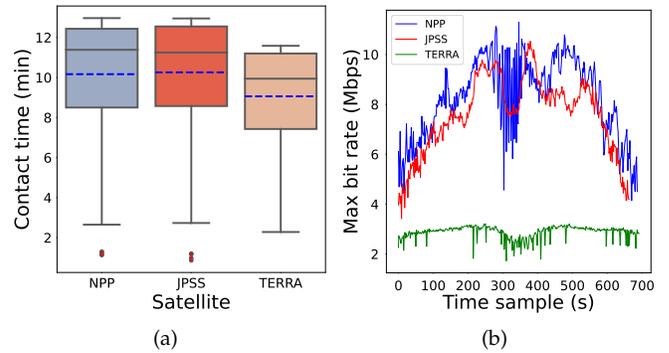
## 2.2 Analysis of Satellite-Terrestrial Downlink Dynamics

In this part, we analyze the real-world satellite-terrestrial network from three X-band LEO satellites (JPSS, TERRA, and NPP) for earth observations and reveal the challenging network conditions. The dataset contains link measurement data for a month, starting from September 2020 [10] [29]. Our major findings are as follows.

### The satellite-terrestrial communication is intermittent.

An LEO satellite typically follows a polar orbit and takes about 90 to 120 minutes to complete one full orbit [30]. The position of the satellite in its orbit from the view of its ground-station pair is defined by azimuth angle  $\phi$  and elevation angle  $\theta$  as illustrated in Fig. 3. Based on our dataset, one contact window between an LEO satellite and ground station pair mostly lasts from 5 to 15 minutes and it happens 3 to 5 times daily. The variation of the contact window for each satellite pass is presented in Fig. 2(a). All three satellites show great variations in the contact windows' length with a standard deviation of around 3 minutes.

**The satellite-terrestrial network is fluctuating and with limited data rate.** We further analyze the satellite-terrestrial network situation. Our analysis reveals that the data transmission rate is fluctuating for every contact window. As Fig. 2(b) shows, for one contact window, different satellites can have different data rate patterns. While JPSS and NPP maintain a relatively high average rate of around 8 Mbps but with high fluctuation, TERRA has a relatively low and



**Fig. 2:** (a) Variations of the contact window for one satellite pass of three satellite ground-station pairs. (b) Fluctuating bit rates for one contact window for three different satellites.

stable data rate with an average of 3 Mbps and a standard deviation of 0.25 Mbps.

In summary, from these results, we can conclude that scarce satellite-terrestrial network resources have to be carefully used to maximize the LEO satellites' value.

## 3 SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we first formally introduce our two-tier HSI processing framework with satellite edge computing. We systematically model the key components in our system and formulate our band selection problem as a combinatorial utility maximization problem.

### 3.1 HSI Satellite Edge Computing Framework

Fig. 3 shows our two-tier hyperspectral image processing framework over the terrestrial-satellite networks. In our framework, the satellite is equipped with a hyperspectral imager to capture HSIs, an onboard edge computing system for band selection, and a communication system for HSI transmission. Ground station is in charge of the final classification. A *band selector* is first trained on the ground and deployed on the onboard computation system beforehand.

Parameter	Description	Unit
$F$	Computation capability of LEO satellite.	CPU cycles/s
$p_i$	Required CPU cycles for processing HSI $d_i$ .	CPU cycles
$t_p$	A time period.	s
$T_i$	Sum of processing and transmission time for HSI image $d_i$ .	s
$b$	Size of a single band with fixed shape.	MB
$s_i$	Size of HSI $d_i$ .	MB
$r_i$	Average transmitting data rate of LEO satellite when transmitting HSI $d_i$ .	MB/s.
$R_s$	Symbol rate for a specific satellite.	MB/s
$P_h$	Power of energy harvesting system.	W
$P_d$	Downlink transmission power of LEO.	W
$P_b$	Power budget for running all other functions on the satellite.	W
$C$	Energy capacity of LEO satellite.	J
$E_{gen}$	Energy generated during a time period.	J
$E_{com}^i$	Energy cost for transmitting HSI $d_i$ .	J
$E_{cmp}^i$	Energy cost for processing HSI $d_i$ .	J
$\epsilon$	Energy factor based on the chip architecture.	
$d_i$	$i_{th}$ HSI in the processing queue of satellite.	
$c_i$	Number of bands in HSI $d_i$ .	

TABLE 1: Important Notations and Symbols.

After being launched, the satellite performs hyperspectral imaging and stores the captured HSIs locally. When a contact window comes, all stored HSIs are organized in a queue for processing and transmission. For each HSI, the satellite selects a subset of bands based on the HSI's content, energy, and network conditions. The selected bands are then transmitted through the downlink to the ground station for further analysis, e.g., classification or object detection. The transmission process terminates when the contact window ends, all HSIs are transmitted, or energy drops to a certain degree that cannot afford extra communication. After a contact window, the satellite stops the transmission and continues to perform hyperspectral imaging. The HSIs left in queue will be processed when a new contact window arrives.

### 3.2 Energy Harvesting and Consumption Model

We next present the energy harvesting and consumption models for our systems. All the important notations we use in this paper are further summarized in Table 1 together with its meaning and unit, if applicable. The energy of nanosatellites mainly comes from the onboard energy harvesting system, e.g., solar panels.

Let  $P_h$  denote the power of the energy harvesting system, given a time period  $t_p$ , we model the energy generated  $E_{gen}$  as follows:

$$E_{gen} = P_h t_p. \quad (1)$$

The energy consumption in our framework mainly consists of two parts: the computation cost when running band selection and other computing functions and the communication cost when transmitting HSIs to the ground station. Similar to [31], we define the computation cost  $E_{cmp}^i$  for processing an HSI image  $d_i$  as follows:

$$E_{cmp}^i = \epsilon (F)^2 p_i, \quad (2)$$

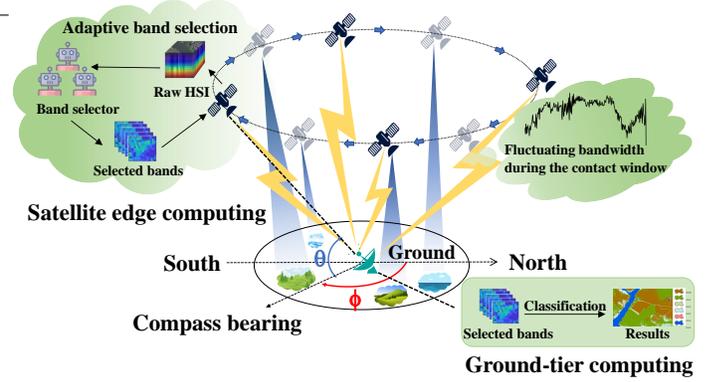


Fig. 3: Two-tier HSI processing framework.  $\theta$  is the elevation angle and  $\phi$  is the azimuth angle.

where  $F$  is the computation capability,  $p_i$  is the required CPU cycles for processing HSI  $d_i$ ,  $\epsilon$  is the energy factor based on the chip architecture [32].

After local processing, the selected bands are transmitted to the ground station through the downlink. Let  $s_i$  denote the size of the HSI to be transmitted,  $P_d$  denotes the downlink transmission power, and  $r_i$  denotes the average downlink data rate when transmitting  $d_i$ , we can model the energy cost  $E_{com}^i$  of transmitting  $d_i$  as follows:

$$E_{com}^i = P_d \frac{s_i}{r_i}. \quad (3)$$

The data size  $s_i$  is mainly determined by the height, width, and selected band number of the HSI. For simplicity, we assume that all HSIs captured by the same satellite have equal width and height, and their size is proportional to their band number. Let  $c_i$  denote the number of bands in  $d_i$ ,  $b$  denote the size (MB) of a single band with fixed shape  $w \times h$ , the energy cost of transmitting  $d_i$  thus can be further modeled as:

$$E_{com}^i = P_d \frac{c_i b}{r_i}. \quad (4)$$

### 3.3 Communication Model

In the current system, since the communication link is mostly used for downlinking pre-processed HSIs to the ground, and uplinking is infrequently used for control purposes, we mainly model the downlinking mode for earth observation satellites in this paper. Nowadays, most digital satellite transmission has adopted the second generation of digital video broadcasting (DVB-S2) standard to realize adaptive coding and modulation (ACM) so that a higher capacity gain and a wider range of code rates can be supported [33]. Based on this standard, spectral efficiency  $\rho_i$  in unit bits per symbol can be found from channel Signal to Noise Ratio (SNR) by looking up the error performance table provided by official documents [33]. Therefore, given the SNR for every second, the resulting bit rate  $r_i$  when transmitting HSI  $d_i$  can be calculated by

$$r_i = \rho_i \times R_s \quad (5)$$

where  $R_s$  represents the symbol rate for a specific satellite. The symbol rate is used to describe the transmission rate of signals for a particular connection. It is determined by

modulation factor, FEC rate, and FEC coding method and is regarded as the constant given a specific type of satellite. For example, for a satellite following a configuration of [QPSK, 21/44, Turbo], the symbol rate is 0.131 MB/s [34].

We denote the total time cost for a satellite to send out an HSI as  $T_i$ . Since we assume HSIs are processed in a pipeline way, we only consider the processing time and the transmission time that consume the satellite's battery life and contact duration. Formally, the total time cost  $T_i$  of a satellite to process an HSI image  $d_i$  can be represented as

$$T_i = \frac{c_i b}{r_i} + \frac{p_i}{F}. \quad (6)$$

where the computation capability  $F$  is specified by the type of CPU. For example, the computation capability of Jetson TX2 is 1.2 GHz [35].

### 3.4 Problem Formulation

Suppose a satellite has  $I$  HSIs at its local storage. Each HSI is denoted by  $d_i, i \in \{1, \dots, I\}$ . The selected band subset is represented using a multi-hot vector  $\mathbf{x}_i = [x_{i,1}, x_{i,2}, \dots, x_{i,M}], x_{i,m} \in \{0, 1\}, m \in \{1, \dots, M\}$ , where  $M$  is the band number of the raw HSIs  $d_i$ . We use a metric  $A$  to describe the resulting analytic performance of the transmitted HSI with its selected bands. Considering the importance of classification tasks [28], we refer  $A$  to the overall accuracy in the HSI classification throughout this paper. Other quantifiable analytic metrics in satellite applications also apply. We represent the overall accuracy of the HSI  $d_i$  with its selected bands  $\mathbf{x}_i$  as  $A_i$ .  $A_i$  can be written as:

$$A_i = A(f(\mathbf{x}_i, d_i), y_i), \quad (7)$$

where  $y_i$  represents the classification ground truth of  $d_i$  and  $f$  represents an HSI classifier. Common HSI classifiers include HybridSN [36], SpectralNet [37] and EvaluateNet [38].

We assume the satellite is charged to its energy capacity  $C$  at the beginning of a contact window with  $T_{max}$  seconds. Based on the selection decisions  $\mathbf{x}_i$ , we also rewrite the corresponding energy cost  $E_{com}^i$  as:

$$E_{com}^i = P_d \frac{b \sum_{m=1}^{m=M} x_{i,m}}{r_i}, \quad (8)$$

and the time cost  $T_i$  as:

$$T_i = \frac{b \sum_{m=1}^{m=M} x_{i,m}}{r_i} + \frac{p_i}{F}. \quad (9)$$

During the contact window, we try to transmit all the stored HSIs while reducing the energy cost and maximizing the classification performance. The utility for transmitting an HSI  $d_i$  thus is defined as:

$$U_i = \alpha A_i - \beta (E_{cmp}^i + E_{com}^i) - \gamma T_i, \quad (10)$$

where  $\alpha, \beta$ , and  $\gamma$  are the positive weight factors to adjust the preference. For our two-tier hyperspectral image processing framework, we target selecting the right band subset for all the HSIs in the storage so that the sum of utilities can be maximized. Correspondingly, we sum up the utility

defined in equation (10) of all HSIs and present the utility maximization problem as follows:

$$\max \sum_{i \in I} U_i \quad (11)$$

$$s.t. \sum_{i \in I} (E_{cmp}^i + E_{com}^i) \leq C + (P_h - P_b) T_{max} \quad (12)$$

$$\sum_{i \in I} T_i \leq T_{max} \quad (13)$$

$$x_{i,m} \in \{0, 1\}, \quad (14)$$

The decision variable  $x_{i,m}$  means whether we select the band  $m$  of an HSI  $d_i$  for transmitting to the ground or not. It influences the downloading time, energy cost, and classification accuracy directly. Constraint (12) guarantees that the total energy consumption in the task won't exceed the sum of capacity and energy harvested during the contact window. Constraint (13) defines that the time cost for processing and transmitting HSIs should be within the contact window. Selecting too many bands may violate the energy constraints (12) and time constraints (13).

**Theorem 1.** *Our utility maximization problem  $\Omega$  is NP-complete.*

*Proof.* See Appendix A in the supplementary material.  $\square$

From the analysis, we can see that our problem is non-trivial in an offline scenario. The problem becomes even more complicated when we handle it in an online scenario, where network and energy situations change over time. Classical model-based solutions may not well adapt to the dynamics of these factors and make sub-optimal decisions. In addition, satellites hover over different parts of the earth and correspondingly collect multiple HSIs of different scenes and landforms. The relationship between the content and the classification results also makes the typical models hard to capture, as we have demonstrated in Sec. 2.

The recent success in deep reinforcement learning offers us an alternative way to solve this complex problem [39] [40]. Reinforcement learning targets maximizing expected total reward in a decision process. While single-agent deep reinforcement learning cannot perform well when the action space is large, in this paper, we design a novel multi-agent deep reinforcement learning approach to solve our utility maximization problem.

## 4 MULTI-AGENT DEEP REINFORCEMENT LEARNING FOR ADAPTIVE BAND SELECTION

In this section, we present MaHSI, a multi-agent deep reinforcement learning-based solution for two-tier HSI processing. We first illustrate the necessity and advantage of a multi-agent solution in detail. Then, we present how we design the multi-agent deep reinforcement learning to adaptively make decisions at the satellite edge and solve our utility maximization problem.

### 4.1 Necessity of a Multi-Agent Solution

We first illustrate the necessity of a multi-agent solution by revealing the large exploration space in our problem. Suppose we have an HSI containing 100 bands. With the

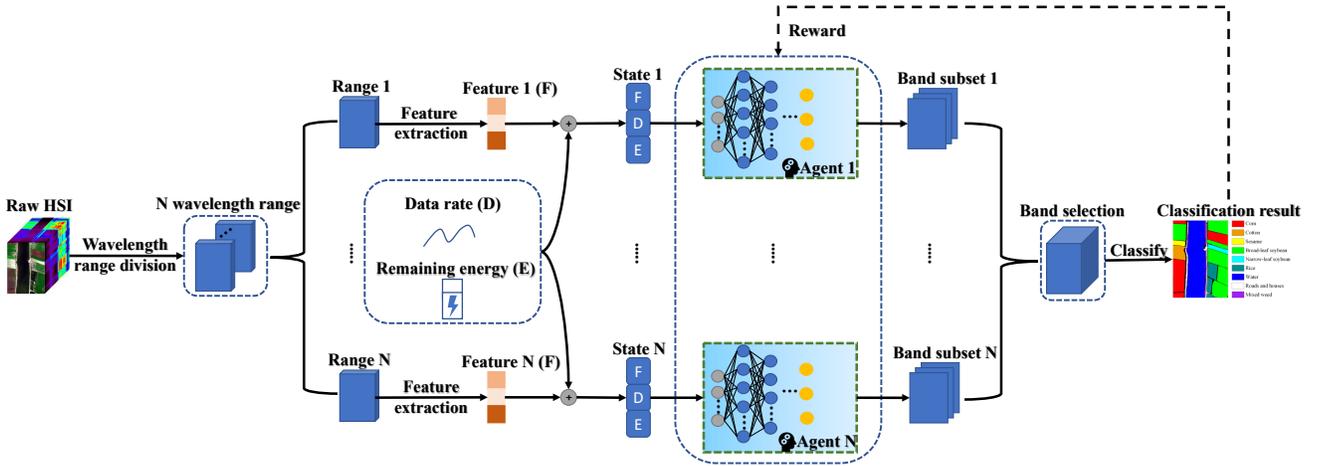


Fig. 4: Model structure of MaHSI.

single-agent reinforcement learning approach such as DQN [41], we need a model with an action space of  $2^{100}$  where each action is a unique combination of bands. The action space grows exponentially as the band number increases. It is unrealistic either to store or run such a huge model, not to mention training.

Multi-agent reinforcement learning (MARL) emerges as a promising solution to reduce action space, enhance information sharing and enable distributed design. Recently, MARL-based methods have achieved SOTA performance in many networked systems scenarios. Researchers in [42] propose a MARL approach to sample several representative video frames for video classification. In [43], RL agents are deployed at different edge servers to make caching strategies individually while learning policy collaboratively. Researchers in [44] design a joint charging and relocation recommendation system for e-taxi drivers with MARL. In [45], each agent separately tries to solve its multiarmed bandit problem independently from the others and makes the data-offloading decision with price and risk awareness. Different works come with different representations of agents and embedding of other key components in MARL. How to apply the MARL framework to the hyperspectral image processing applications has not been studied yet. Compared to existing solutions, our MaHSI framework is carefully designed for adaptive HSI processing on LEO satellites. The proposed framework novelly splits an HSI into multiple band subsets and assigns them to a single agent, which dramatically reduces the action space and enables parallel processing for quick band selection. In addition, it also takes account of the energy and time conditions and perform wise decisions in resource-constrained scenarios. In the following, we present the detailed design for instantiating the multi-agent reinforcement learning framework in our two-tier hyperspectral image processing scenario.

## 4.2 Design

We formulate our utility maximization problem as a Markov Decision Process which involves episode, state, action, and reward. We present our design for each of them as follows.

**Episode:** We model the episode in our MARL algorithm as an HSI transmission process during a contact window. For each HSI  $d_i$  to be processed at time  $t$ , agents select and transmit the selected band subset, then move to the next HSIs  $d_{i+1}$ . The episode terminates when 1) all stored HSIs are transmitted; 2) contact window runs out; 3) the energy of the satellite runs out or is below a certain threshold.

**Agent:** We uniformly divide an original HSI into several small band subsets with  $L$  bands, denoted by  $w_j = [j \times (L - 1) + 1, \dots, j \times L]$ , where each term of  $w_j$  represents the index of a specific band. Each band subset  $w_j$  falls into a specific wavelength range  $W_j = (W_{low}^j, W_{high}^j]$ , where  $W_{low}^j$  and  $W_{high}^j$  is the lower bound and upper bound of wavelength range  $W_j$ , respectively. In this way, the relation information between adjacent bands is preserved and it makes multi-agent design possible. We then assign each band subset  $w_j$  to a single agent  $j$  for band selection. After all of the agents produce band selection in their own band subset  $w_j$ , we merge them as the final band selection for HSI.

**State:** As shown in our analysis of Fig 15(a) and Fig 15(b), the optimal band selection is quite content-varying. We input an informative feature vector  $f_{j,i}$  to help the agent develop content awareness for adaptive band selection.  $r_i$  is the predicted data rate of the future downlink. Note that this attribute is closely related to our problem formulations, for example, the constraints (12) and (13). With the predicted data rate, agents are aware of the incoming network conditions and thus can learn to make a trade-off between classification performance and transmission and energy cost. We further incorporate the current remaining energy  $e_i$  in our state design so that agents can learn a long-term plan and use the remaining energy wisely to achieve higher utility.

In summary, the state for each agent  $j$  of HSI  $d_i$  is denoted by  $s_{j,i} = \{f_{j,i}, r_i, e_i\}$ , where

- $f_{j,i}$  is the feature vector extracted from the band subset  $w_j$  in the wavelength range  $W_j$ .
- $r_i = \{r_i^t, r_i^{t+1}, \dots, r_i^{t+k-1}\}$  is the predicted incoming data rate from time  $t$  to  $t+k-1$ .
- $e_i$  is the remaining energy at time  $t$  of the satellite.

**Action:** With the observation of state  $s_{j,i}$  and the HSI image  $i$ , the agent  $j$  will take an action  $x_j^i = \{x_{j,1}^i, x_{j,2}^i, \dots, x_{j,L}^i\}$ ,  $x_{j,l}^i \in \{0, 1\}$ ,  $l \in \{1, \dots, L\}$ .  $x_{j,i}^l = 1$  means that  $l$ -th band of  $w_j$  is selected and  $x_{j,l}^i = 0$  otherwise.

**Reward:** According to our problem  $\Omega$ , we define the reward  $R_i$  for processing the HSI  $d_i$  as follows:

$$R_i = \alpha(A_i - A_{full}^i) - \beta(E_{cmp}^i + E_{com}^i) - \gamma S_i. \quad (15)$$

Here,  $\alpha$ ,  $\beta$ , and  $\gamma$  is the weight factor to adjust the agent's preference on classification, energy-saving, and transmission time. We can see there are some slight differences between our utility function in (10) and the reward function in (15). We make such a modification for two main reasons. First, we subtract  $A_{full}^i$  from  $A_i$ , which is the classification result, e.g., overall accuracy, with the selected bands on  $d_i$  given by a well-trained classification model, in (15) to reduce the variance among classification results of different HSIs and makes the training more stable. Second, to satisfy the constraint (13) in reinforcement learning and transmit HSIs as much as possible, we add a penalty denoted by  $S_i$  to our reward function. To compute so, we first define the average time we can assign to each HSI during contact with the ground station as  $T_{avg}$ , where

$$T_{avg} = \frac{T_{max}}{I}. \quad (16)$$

Intuitively, if the time cost of processing and transmission in total for each HSI is within  $T_{avg}$ , we can transmit all HSIs in a contact window. Based on  $T_{avg}$ , we define  $S_i$  to measure the amount of time that the satellite has exceeded the  $T_{avg}$  when transmitting  $d_i$ . It will be set to zero when the actual time cost is within  $T_{avg}$ . Formally, our penalty  $S_i$  is defined as:

$$S_i = \begin{cases} 0 & T_i \leq T_{avg} \\ T_i - T_{avg} & T_i > T_{avg}. \end{cases} \quad (17)$$

**Network architecture** In our MaHSI framework, we use DQN [41] as the RL training algorithm for each agent. DQN is a value-based RL approach that seeks a critic to judge how good a state-action pair is, denoted by  $Q(s, a)$ . The basic idea behind DQN is to estimate the action-value function by using the Bellman equation as an iterative update. The Bellman equation is written as follows:

$$Q(s, a) = \mathbb{E}_{s'}[r + \mu \max_{a'} Q(s', a') | s, a], \quad (18)$$

where  $r$  denotes the reward,  $\mu$  is the discounted factor and  $Q(s', a')$  is the state-action value of next step. DQN uses a neural network named Q-network with parameter  $\theta$  to approximate  $Q(s, a)$ . The Q-network in our work is a multi-layer perceptron (MLP) with stacked fully-connected layers and Tanh as the activation function. The Q-network can be trained to minimize the mean squared error in the Bellman equation with the approximate target value  $y$ :

$$y = r + \mu \max_{a'} Q(s', a'; \theta), \quad (19)$$

where  $\theta$  is the parameter of the Q-network. Specifically, the loss function for training DQN can be calculated as follows:

$$L(\theta) = \mathbb{E}_{s,a,r}[(\mathbb{E}_{s'}[y|s, a] - Q(s, a; \theta))^2], \quad (20)$$

With the loss function, we then apply gradient descent to update the network parameter:

$$\theta' = \theta - \eta \nabla_{\theta} L(\theta), \quad (21)$$

where  $\eta$  is the learning rate.

---

**Algorithm 1** Adaptive HSI Processing with MaHSI in One Contact Window

---

**Input:** A queue of captured HSIs  $Q$ , remained contact window  $T$ , remained satellite energy  $E$ .

**Output:** HSIs with selected bands.

- 1: **while**  $T > 0$  &  $E > 0$  & ! $Q.empty()$  **do**
  - 2:   Pop a new HSI  $d_i$  from  $Q$ .
  - 3:   Initialize an empty band set  $S = \{\}$  for  $d_i$ .
  - 4:   Spilt  $d_i$  into equal band subsets  $\{d_{i,i}, d_{2,i}, \dots, d_{N,i}\}$  according to the number of the agents of MaHSI.
  - 5:   Predict incoming data rate  $r_i$ .
  - 6:   **for** agent  $j = 0$  to  $N$  **do**
  - 7:     Apply PCA on  $d_{j,i}$  to extract features  $f_{j,i}$ .
  - 8:     Concatenate  $f_{j,i}$ ,  $r_i$  and  $e_i$  to form state  $s_i$
  - 9:     Input  $s_{j,i}$  to agent  $j$  to get selected band subset  $b_{j,i}$ .
  - 10:      $S \leftarrow S \cup b_{j,i}$ .
  - 11:   **end for**
  - 12:   Transmit  $S$  to the ground through the terrestrial-satellite network.
  - 13:   Obtain time cost  $t$  and energy cost  $e$ .
  - 14:    $T \leftarrow T - t$
  - 15:    $E \leftarrow E - e$
  - 16: **end while**
- 

**Terrestrial-satellite network prediction** In addition to the core policy network, our decision-making model relies on the prediction of network conditions to handle dynamic environments. For LEO satellites, though the orbit is fixed, the network quality is also vulnerable to harsh weather like rain, snow, and cloud [46]. Classical model-based approaches work for general link behavior and do not capture link fluctuations due to specific satellite-ground station pair design, which directly leads to lower accuracy of prediction. Data-driven approaches have shown their effectiveness as compared to simple heuristics in this sub-problem [47] [48] [10]. Though orthogonal to this study, in this paper, we adopt the Long Short Term Memory (LSTM) recurrent neural network, which is commonly used for time series prediction, to predict the incoming SNR [48] [49]. We do not directly train an LSTM model to predict the datarate since datarate is discrete according to the satellite communication protocol and it leads to a non-differentiable loss function. The input to the LSTM prediction network contains the azimuth angle and elevation angle at each time step and recent bandwidth measurements. We then transform the predicted SNR into datarate according to our communication model in Section III. The effectiveness of the LSTM for the prediction of the terrestrial-satellite communication network is also evaluated in the experiment part. More advanced network prediction approaches can also be used in our framework. In summary, the overall model structure is presented in Fig. 4. After the policy network is trained, the detailed inference process for MaHSI is also summarized in Algorithm 1.

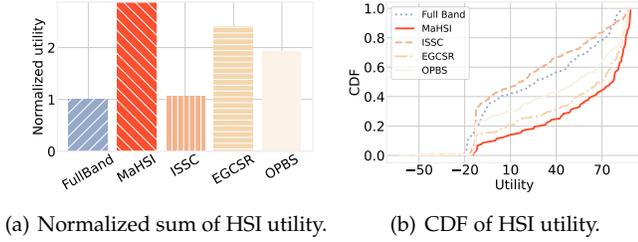


Fig. 5: HSI utility comparison of different approaches

### 4.3 Computation Complexity Analysis

The computation complexity of our MaHSI is two-fold, one is the complexity of training and the other is the complexity of inference. The train complexity simply depends on the number of steps  $S$  we want to train it, denoted as  $O(S)$  [50]. As for inference, we first assume that the inference time of a neural network is proportional to its number of parameters. Since we use the same network architecture for MaHSI with a different number of agents, the only thing we need to consider is the inference time complexity of the last layer, the parameter number of which depends on the agent number and HSI band number. Let  $P$  denote the parameter number of all network layers except for the last one, for MaHSI with  $N$  agents and an HSI with  $M$  bands, the time complexity of processing an HSI can be calculated as  $O(N(P + \frac{M}{N}))$ . Here,  $O(P + \frac{M}{N})$  represents the inference complexity of one agent.

## 5 PERFORMANCE EVALUATION

### 5.1 Evaluation Setup

**HSI Data:** We use eight available real-world HSI classification datasets to simulate the captured HSIs with 200 bands from an LEO satellite. The details for these datasets are shown in Table 2. To resolve the inconsistency in shapes, we first pre-process these datasets by expanding those datasets with less than 200 band sizes with zeros-filled bands to make them equal in total band size. The zeros-filled bands won't be selected. Each dataset is then cut into uniform-sized image tiles with the size of  $100 \times 100 \times 200$  to meet the requirement of the downstream classifier.

Dataset	Size	Number of Classes
IndiaPines [26]	$145 \times 145 \times 200$	16
SalinasScene [26]	$512 \times 217 \times 200$	16
PaviaUniversity [26]	$610 \times 340 \times 103$	9
KennedySpaceCenter [26]	$512 \times 614 \times 176$	13
Botswana [26]	$1476 \times 256 \times 145$	14
Houston [51]	$349 \times 1905 \times 144$	17
HyRANKDioni [27]	$250 \times 1376 \times 176$	14
HyRANKLoukia [27]	$249 \times 945 \times 176$	14

TABLE 2: Details of our used HSI datasets.

**Network data:** We use the link data from a real-world satellite, JPSS, to trace drive the network [10] [29]. JPSS operates in the X band, which is a very common band for earth observation. Link data contains the elevation and azimuth angles of the satellite with respect to the ground station, signal strength, and SNR (Signal to Noise Ratio) of the received signal from the satellite. We add the attenuation factor to the derived ideal capacity so that the data rate

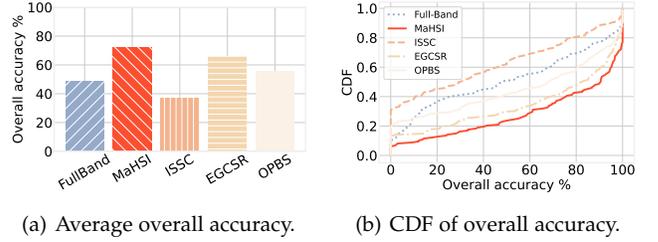


Fig. 6: Accuracy comparison of different approaches.

is within the range of 1~10 Mbps, which matches the real-world downlink situation for hyperspectral imaging nanosatellite [12].

**HSI classifier:** We use EvaluateNet [38] as our classifier. EvaluateNet is a state-of-the-art classifier specially designed for evaluating band selection algorithms. We follow the same setting for the size of the input regional data cubes, learning rate, and episodes as in the original paper.

**Experiment environment and parameter settings** We implemented MaHSI using TensorFlow 2.2. We select 50% of the dataset as the policy training set, and the left 50% is used for policy testing. We train our MaHSI on a Ubuntu 20.04 machine with RTX 2070super GPU cards, Intel i7-10700 CPU 2.90GHz, and 64GB memory. The trained MaHSI is deployed on a Jetson TX2, which is a common testbed for examining satellite edge computing [11] [52], for inference. Several types of LEO satellites [53] [54] have also been equipped with it to perform machine learning inference. With a power of only 7.5W, Jetson TX2 is appropriate to conduct deep learning tasks on LEO satellites with limited energy. The original HSI is split into 50 small band subsets, where each wavelength range uniformly covers 4 bands of an HSI. We normalize each metric in the utility function to  $(0, 1]$  for fair aggregation. Both HSI utility and RL reward are set with weight factors  $\alpha = 1, \beta = 0.2, \gamma = 0.2$ . The learning rate for agents in MaHSI is set to 0.0005.

For SNR prediction, we train different prediction models offline for different satellites respectively. The model takes azimuth and elevation angle and recent 5 SNR measurements as input features and predicts next 5-second SNR values. Learning rate is set to 0.001. We select 80% of the network data as the training set, and the left 20% as the testing set.

**Benchmarks:** We use the following four benchmarks to evaluate our MaHSI framework. In addition to the full band baseline, we also select three representative ground-based band selection algorithms under different categories. We fixed the number of selected bands of the 3 traditional band selection algorithms as the same as the average number of bands selected by MaHSI.

- 1) **FullBand:** We transmit all bands for every captured HSI. This is the default operating mode of traditional hyperspectral imaging satellites.
- 2) **ISSC:** ISSC [17] is a traditional *clustering-based* band selection method. It applies spectral clustering on HSI and selects the cluster centers as the selected bands.
- 3) **EGCSR-Ranking (EGCSR):** EGCSR [23] is a traditional *ranking-based band* selection method. It

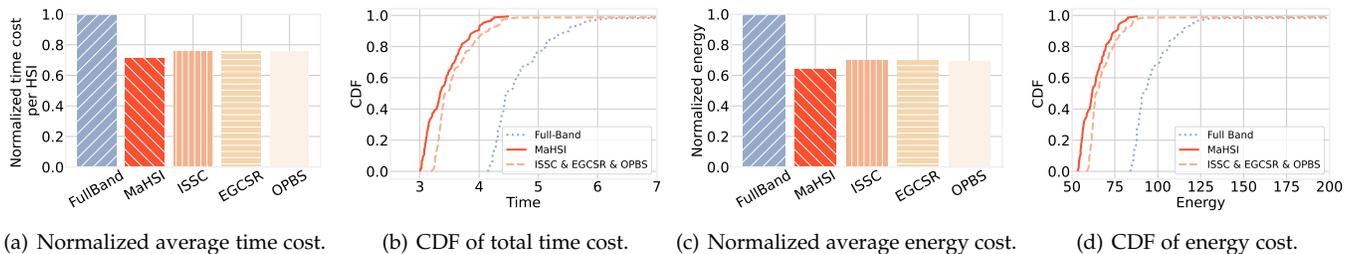


Fig. 7: Energy and time cost comparison of different results.

incorporates graph convolution into the self-representation model for band selection and selects the top  $k$  bands according to the defined ranking score.

- 4) **OPBS**: OPBS [18] is a traditional *searching-based* band selection method. It combines the maximum ellipsoid volume and sequential forward search for band selection.

## 5.2 Evaluation Results

**Overall utility comparison** Fig. 5(a) and Fig. 5(b) present the average normalized sum of HSI utility and the CDF of HSI utility. The sum of utility calculates the achieved utility of all HSIs throughout an entire contact window. As shown in Fig. 5(a), our MaHSI outperforms all other 4 methods with the highest HSI utility. Specifically, MaHSI achieves 2.87, 2.73, 1.20, and 1.50 times higher utility than that of FullBand, ISSC, EGCSR, and OPBS, respectively. In addition, Fig. 5(b) shows that nearly 60% of the processed HSIs achieve higher HSI utility, over 70, with MaHSI. In contrast, only 20% of the HSIs achieve this value in full band strategy. The best-performed classical band selection algorithm, EGCSR, still has 7% fewer HSIs with utility over 70 compared to MaHSI. Recall that our utility definition contains two negative terms for time and energy consumption. Thus, some HSIs with low classification scores might result in negative utility as shown in Fig. 5(b). Note again that, all the classical band selection algorithms already use the average band size summarized from the MaHSI’s learned policy as their band selection sizes. In practice, identifying the appropriate band size requires tremendous hand-tuning efforts or domain knowledge.

**Classification performance** We next decompose the overall utility into three considered components to examine the performance of our approach. We first compare the classification performance between MaHSI and four other methods. From Fig. 6(a) we can see that MaHSI outperforms FullBand, ISSC, EGCSR, and OPBS by 24%, 35%, 7%, and 17% in overall accuracy, respectively. In addition, as shown in Fig. 6(b), 30% HSIs processed by MaHSI achieve nearly 100% overall accuracy. In contrast, only 9%, 5%, 23%, and 10% of the HSIs have 100% overall accuracy in FullBand, ISSC, EGCSR, and OPBS, respectively. With an adaptive band selection strategy and content awareness, MaHSI can find the optimal band selection for HSI while traditional methods fail with a fixed band number. Note that the other three band selection methods perform much worse compared to the results reported in their papers. Because

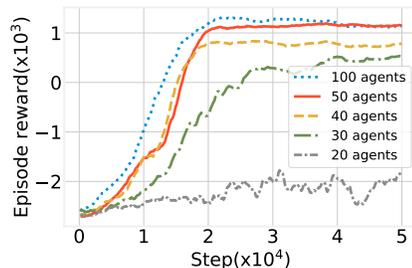


Fig. 8: Learning curve of MaHSI with different number of agents.

their classification results come from the classifier trained on samples of selected bands and then tested on the remaining samples. We get the classification result by simply feeding all the samples into a pre-trained classifier. We argue that our evaluation strategy is more practical for real-world scenarios since we have no labeled training data for newly collected HSIs.

**Time and network cost** We next examine the average total time cost and the average energy cost of processing one HSI in Fig. 7. Total time cost measures the sum of transmission time and the computation time as we modeled before. In Fig. 7(a) we can see that MaHSI has the lowest average time cost for each HSI. From the CDF plot of time cost in Fig. 7(b), we find that the time cost of the FullBand strategy varies widely in the range of [4.2, 7]. As a comparison, our MaHSI can keep the transmission time within a smaller range in [3, 4.5] no matter how the network condition changes. In addition, though we set the average band size of 3 traditional band selection methods the same as MaHSI, MaHSI still achieves less time cost for 100% HSIs than those methods. Because these fixed methods still try to transmit the same number of bands when the network condition is bad. It shows that our MaHSI can adjust its band selection strategy according to the network situation. This capability helps the satellite better utilize the contact window with good network conditions and avoid packet loss when bandwidth is limited. Similar results can be discovered in the energy cost plot in Fig. 7(c) and Fig. 7(d) because the energy cost is also related to network conditions. In general, MaHSI reduces energy cost by 8% compared to the classical fixed band approaches, and by 36% compared to the full band strategy. In the CDF of energy cost, 90% HSIs processed by MaHSI consume less than 75 energy. While only 80% and 0% of the HSIs pro-

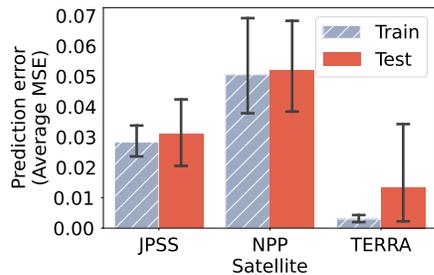


Fig. 9: Network prediction performance of our LSTM model.

cessed by 3 traditional methods and FullBand consume less than 75 energy, respectively. Since the capacitor on satellites has limited capacity, sudden high energy consumption may run out of the stored energy and make the satellite fail to maintain communication with the ground station. Our MaHSI can well handle this situation because it performs band selection based on current network conditions and remaining energy.

**Sensitivity analysis of agent numbers** We then investigate the influence of agents' number on MaHSI. Fig. 8 shows the training process of MaHSI with different number of agents from 20 to 100. As the number of agents increases, the convergence speed, and episode reward improve gradually since the action space reduces with more agents involved. It becomes easier for each agent to learn the global optimal strategy. The fluctuating performance of MaHSI with 20 agents after fixed training steps demonstrates the difficulty in learning the optimal policy in a large action space of  $20 \times 2^{10}$ . The achieved reward improves by 55% when we increase the agent size from 30 to 40. However, a large action space may prohibit MaHSI with 30 and 40 agents to learn a global optimal policy, compared to the performance of MaHSI-50. On the other hand, we can also notice the diminishing return for introducing more agents. MaHSI with 50 agents and 100 agents almost ends with the same final performance. A possible explanation is that the benefit of reducing the action space is neutralized by the spectral information loss for each agent brought by introducing more agents. In addition, another shortcoming of involving more agents is that it takes more space to store the models on resource-constrained nanosatellites.

**Terrestrial-satellite network prediction results** We now analyze the effectiveness of the LSTM-based network prediction model. We compute the prediction error for three satellites with different network characteristics as their snapshots also being indicated in Fig. 2(b). Fig. 9 shows the average mean squared error (MSE) for both the training set and testing set. The average MSE is below 0.052 for all three satellites. The better performance in TERRA can be boiled down to two reasons. First, it results from the low variance of the bit rate for Terra and relatively stable trend compared to the other two satellites, which can also be seen in Fig. 2(b). Second, it is due to environmental characteristics (obstacles, reflections) of the setup. For instance, JPSS and NPP have antennas closer to solar panels, which makes it harder to predict the link quality [10]. These render the prediction easier and more accurate. These results, along with the previously demonstrated performance in accuracy

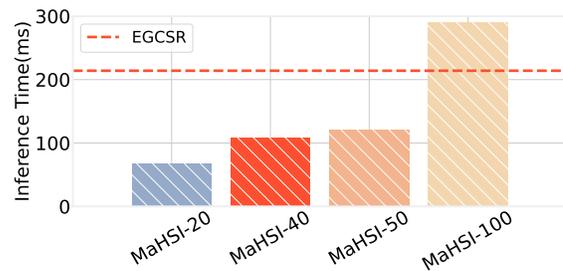


Fig. 10: Average inference time of MaHSI on Jetson TX2 with different number of agents.

and other factors indicate that the LSTM-based network prediction model can serve our purpose well. More advanced data-driven network prediction can also be designed and applied in our general framework, which is not the focus of this work. When this module is incorporated into our framework, we select the minimum of the next five-second predicted SNR as the next-second SNR to avoid over-estimation of the SNR.

**Inference Time Measurement** We measure the inference time of MaHSI with the different number of agents on Jetson TX2. Fig. 10 shows our results. The dashed line represents the inference time of EGCSR, which is the fastest among all three band selection methods in our benchmarks. As we can see, MaHSI with 50 agents, which performs the best in our experiment, takes only about 100ms to process an HSI with 200 bands, faster than all other three benchmarks. The main reason is that our MaHSI can take advantage of the GPU on Jetson TX2 while the other three methods can only use CPU for inference. In addition, methods like clustering-based ISCC and searching-based OPBS take multiple iterations/steps to find the solution. In addition to the real execution time, we also record the actual storage size of our model. MaHSI with agent numbers 20, 40, 50, and 100 only occupies 27.7, 14.6, 17.4, and 33.7 MB, respectively. Thus, our MaHSI is light-weighted enough to run onboard for real-time HSI processing.

## 6 CONCLUSION

In this paper, we present the first work for two-tier hyperspectral image processing using satellite edge computing and terrestrial-satellite collaboration. We first reveal the non-trivial relationship between the band selection decisions and the eventual classification performance. Our data analysis also validates that satellite-terrestrial communication is intermittent and fluctuating in data bandwidth. We present the two-tier collaborative hyperspectral image processing framework and formulate the core in-orbit band selection problem as a utility maximization problem. Given the NP-Completeness of our problem and the dynamics of the satellite environment, we present a novel multi-agent reinforcement learning-based approach for adaptive band selection. Our approach judiciously embeds the correlation among neighboring wavelengths by allowing multiple agents to take charge of the individual band subset. It successfully makes adaptive decisions with the observation of the HSI content feature, downlink network condition, and

the remaining energy of the satellite. Experiments on real-world communication and HSI datasets demonstrate that our method outperforms both the full-band strategy and the traditional SOTA band selection methods.

## 7 ACKNOWLEDGEMENT

The authors would like to thank the Hyperspectral Image Analysis group and the NSF Funded Center for Airborne Laser Mapping (NCALM) at the University of Houston for providing the data sets used in this study. This work is supported by "the Fundamental Research Funds for the Central Universities".

## REFERENCES

- [1] M. Swartwout. Cubesat database. [Online]. Available: <https://sites.google.com/a/slu.edu/swartwout/home/cubesat-database>
- [2] U. of Concerned Scientists. Ucs satellite database. [Online]. Available: <https://www.ucsus.org/resources/satellite-database>
- [3] V. Bhosale, K. Bhardwaj, and A. Gavrilovska, "Toward loosely coupled orchestration for the LEO satellite edge," in *Proc. USENIX HotEdge*, Jun. 2020.
- [4] L. Liang, L. Di, L. Zhang, M. Deng, Z. Qin, S. Zhao, and H. Lin, "Estimation of crop lai using hyperspectral vegetation indices and a hybrid inversion method," *Remote Sens. Environ.*, vol. 165, pp. 123–134, Apr. 2015.
- [5] N. Yokoya, J. C.-W. Chan, and K. Segl, "Potential of resolution-enhanced hyperspectral data for mineral mapping using simulated enmap and sentinel-2 images," *Remote Sens.*, vol. 8, no. 3, p. 172, Feb. 2016.
- [6] A. C. Karaca, D. Çeşmeci, A. Ertürk, M. K. Güllü, and S. Ertürk, "Hyperspectral change detection with stereo disparity information enhancement," in *Workshop Hyperspectral Image and Signal Process.: Evolution in Remote Sens. (WHISPERS)*, Oct. 2014, pp. 1–4.
- [7] R. S. Allison, J. M. Johnston, G. Craig, and S. Jennings, "Airborne optical and thermal remote sensing for wildfire detection and monitoring," *Sensors*, vol. 16, no. 8, p. 1310, Nov. 2016.
- [8] C. Floridaia, F. C. Salgado, J. B. Rosolem, F. R. Bassan, J. P. V. Fracarolli, R. S. Penze, and L. M. Pereira, "Methane leak detection and spectral analysis by using only optical time domain reflectometry in semidistributed remote optical sensors," in *Proc. IEEE SENSORS*, Oct. 2016, pp. 1–3.
- [9] K. Devaraj, R. Kingsbury, M. Ligon, J. Breu, V. Vittaldev, B. Klofas, P. Yeon, and K. Colton, "Dove high speed downlink system," in *Proc. AIAA/USU Conf. Small Satell.*, 2017.
- [10] D. Vasisht, J. Shenoy, and R. Chandra, "L2D2: low latency distributed downlink for LEO satellites," in *ACM SIGCOMM, Virtual Event, USA*, Aug. 2021, pp. 151–164.
- [11] B. Denby and B. Lucia, "Orbital edge computing: Nanosatellite constellations as a new class of computer system," in *Proc. ACM Int. Conf. Architectural Support for Programming Languages and Operating Syst. (ASPLOS)*, Lausanne, Switzerland, Mar. 2020, pp. 939–954.
- [12] Gomx-4. [Online]. Available: <https://directory.eoportal.org/web/eoportal/satellite-missions/g/gomx-4>
- [13] D. L. Cotten, N. Neel, D. Mishra, M. Madden, and F. Beyette, "The spectral ocean color imager (spoc) - an adjustable multispectral imager," in *Proc. AIAA/USU Conf. Small Satell.*, Logan Utah, Aug. 2019.
- [14] The new space age: Ibm develops a unique, custom edge computing solution in space. [Online]. Available: <https://www.ibm.com/cloud/blog/ibm-develops-a-unique-custom-edge-computing-solution-in-space>
- [15] Huawei's cloud integration making progress. [Online]. Available: <https://www.globaltimes.cn/page/202112/1241203.shtml?id=11>
- [16] Y. Li, Y. Wu, M. Dai, B. Lin, W. Jia, and X. Shen, "Hybrid nomadma assisted dual computation offloading: A latency minimization approach," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 5, pp. 3345–3360, 2022.
- [17] W. Sun, L. Zhang, B. Du, W. Li, and Y. Mark Lai, "Band selection using improved sparse subspace clustering for hyperspectral imagery classification," *IEEE J. Sel. Topics in Applied Earth Observations and Remote Sens.*, vol. 8, no. 6, pp. 2784–2797, Apr. 2015.
- [18] W. Zhang, X. Li, Y. Dou, and L. Zhao, "A geometry-based band selection approach for hyperspectral image analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4318–4333, Mar. 2018.
- [19] H. Guo, Q. Yang, H. Wang, Y. Hua, T. Song, R. Ma, and H. Guan, "Spacedml: Enabling distributed machine learning in space information networks," *IEEE Netw.*, vol. 35, no. 4, pp. 82–87, Aug. 2021.
- [20] T. Pfandzelter, J. Hasenburg, and D. Bermbach, "Towards a computing platform for the LEO edge," in *Proc. ACM Int. Workshop on Edge Syst., Analytics and Netw.*, Online Event, United Kingdom, Jan. 2021, pp. 43–48.
- [21] N. Tsog, M. Behnam, M. Sjödin, and F. Bruhn, "Intelligent data processing using in-orbit advanced algorithms on heterogeneous system architecture," in *Proc. IEEE Aerosp. Conf.*, MT, USA, Mar. 2018, pp. 1–8.
- [22] Y. Zhang, H. Zhang, H. Zhou, K. Long, and G. K. Karagiannidis, "Resource allocation in terrestrial-satellite-based next generation multiple access networks with interference cooperation," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 4, pp. 1210–1221, Apr. 2022.
- [23] Y. Cai, Z. Zhang, X. Liu, and Z. Cai, "Efficient graph convolutional self-representation for band selection of hyperspectral image," *IEEE J. Sel. Topics in Applied Earth Observations and Remote Sens.*, vol. 13, pp. 4869–4880, Aug. 2020.
- [24] X. Cao, T. Xiong, and L. Jiao, "Supervised band selection using local spatial information for hyperspectral image," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 3, pp. 329–333, Jan. 2016.
- [25] M. A. Arab, K. Calagari, and M. Hefeeda, "Band and quality selection for efficient transmission of hyperspectral images," in *Proc. ACM MM*, Nice, France, Oct. 2019, pp. 2423–2430.
- [26] "Hyperspectral remote sensing scenes," [http://www.ehu.eu/ccwintco/index.php?title=Hyperspectral\\_Remote\\_Sensing\\_Scenes](http://www.ehu.eu/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes), Grupo de Inteligencia Computacional.
- [27] K. Karantzalos, C. Karakizi, Z. Kandykakis, and G. Antoniou, "HyRank hyperspectral satellite dataset I," Apr. 2018.
- [28] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Deep learning for hyperspectral image classification: An overview," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6690–6709, Apr. 2019.
- [29] L2d2 satellite datasets. [Online]. Available: <https://github.com/ConnectedSystemsLab/L2D2>
- [30] Y. Borthomieu, "Satellite lithium-ion batteries," in *Lithium-ion batteries*. Elsevier, 2014, pp. 311–344.
- [31] Q. Tang, Z. Fei, B. Li, and Z. Han, "Computation offloading in LEO satellite networks with hybrid cloud and edge computing," *IEEE Internet Things J.*, vol. 8, no. 11, pp. 9164–9176, Feb. 2021.
- [32] A. P. Miettinen and J. K. Nurminen, "Energy efficiency of mobile clients in cloud computing," in *Proc. USENIX Workshop on Hot Topics in Cloud Comput.*, 2010.
- [33] D. ETSI, "Document etsi en 302 307 v1. 2.1 (2009-08)," *Digital Video Broadcasting (DVB)*.
- [34] Symbol rate. [Online]. Available: <https://www.satsig.net/symbol01.htm>
- [35] Nvidia jetson tx2 module. [Online]. Available: <https://www.nvidia.cn/autonomous-machines/embedded-systems/jetson-tx2/>
- [36] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, "Hybridsn: Exploring 3-d-2-d cnn feature hierarchy for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 277–281, Jun. 2020.
- [37] T. Chakraborty and U. Trehan, "Spectralnet: Exploring spatial-spectral waveletcnn for hyperspectral image classification," *CoRR*, vol. abs/2104.00341, Apr. 2021.
- [38] J. Feng, D. Li, J. Gu, X. Cao, R. Shang, X. Zhang, and L. Jiao, "Deep reinforcement learning for semisupervised hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–19, Dec. 2022.
- [39] M. Afrin, J. Jin, A. Rahman, A. Rahman, J. Wan, and E. Hosain, "Resource allocation and service provisioning in multi-agent cloud robotics: A comprehensive survey," *IEEE Commun. Surv. Tut.*, vol. 23, no. 2, pp. 842–870, Feb. 2021.
- [40] C. Wang, X. Yao, W. Wang, and J. M. Jornet, "Multi-hop deflection routing algorithm based on reinforcement learning for energy-harvesting nanonetworks," *IEEE Trans. Mob. Comput.*, vol. 21, no. 1, pp. 211–225, Jul. 2022.
- [41] V. Mnih, K. Kavukcuoglu, D. Silver, and A. A. Rusu, "Human-level control through deep reinforcement learning," *Nat.*, vol. 518, no. 7540, p. 529, Jun. 2015.

- [42] W. Wu, D. He, X. Tan, S. Chen, and S. Wen, "Multi-agent reinforcement learning based frame sampling for effective untrimmed video recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vision (ICCV)*, Oct. 2019, pp. 6221–6230.
- [43] F. Wang, F. Wang, J. Liu, R. Shea, and L. Sun, "Intelligent video caching at network edge: A multi-agent deep reinforcement learning approach," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Toronto, ON, Canada, Aug. 2020, pp. 2499–2508.
- [44] E. Wang, R. Ding, Z. Yang, H. Jin, C. Miao, L. Su, F. Zhang, C. Qiao, and X. Wang, "Joint charging and relocation recommendation for e-taxi drivers via multi-agent mean field hierarchical reinforcement learning," *IEEE Trans. Mob. Comput.*, vol. 21, no. 4, pp. 1274–1290, Sept. 2022.
- [45] G. Mitsis, E. E. Tsiropoulou, and S. Papavassiliou, "Price and risk awareness for data offloading decision-making in edge computing systems," *IEEE Systems Journal*, pp. 1–12, 2022.
- [46] G. Ezech, N. Chukwunke, N. Ogujiofor, and U. Diala, "Effects of rain attenuation on satellite communication link," *Advances in Sci. Technol. Res. J.*, vol. 8, no. 22, pp. 1–11, 2014.
- [47] Y. Zuo, F. Zhang, and C. Jia, "Prediction for satellite communication based on data mining," in *2016 15th International Conference on Optical Communications and Networks (ICOON)*. IEEE, 2016, pp. 1–3.
- [48] L. Mei, R. Hu, H. Cao, Y. Liu, Z. Han, F. Li, and J. Li, "Realtime mobile bandwidth prediction using lstm neural network and bayesian fusion," *Compt. Netw.*, vol. 182, p. 107515, 2020.
- [49] F. Altché and A. de La Fortelle, "An lstm network for highway trajectory prediction," in *IEEE Int. Conf. Intell. Transp. Syst. (ITSC)*. IEEE, 2017, pp. 353–359.
- [50] C. Jin, Z. Allen-Zhu, S. Bubeck, and M. I. Jordan, "Is q-learning provably efficient?" *Advances in neural information processing systems*, vol. 31, 2018.
- [51] 2013 ieeegrss data fusion contest. [Online]. Available: <http://www.grss-ieee.org/community/technical-committees/data-fusion/>
- [52] B. Denby and B. Lucia, "Orbital edge computing: Machine inference in space," *IEEE Computer Architecture Letters*, vol. 18, no. 1, pp. 59–62, 2019.
- [53] spiralblue. [Online]. Available: <https://www.spiralblue.space/space-edge-computing>
- [54] Ai supercomputing from aitech for next generation space apps. [Online]. Available: <https://news.satnews.com/2021/08/04/ai-supercomputing-from-aitech-for-next-generation-space-apps/>



**Botao Zhu** received his B.Eng. degree from Software College, Northeastern University, China in 2022. He is currently a Ph.D. candidate in UM-SJTU Joint Institute, Shanghai Jiao Tong University. His research interests include reinforcement learning, cloud and edge computing.



**Siyuan Lin** is currently a senior student in UM-SJTU Joint Institute, Shanghai Jiao Tong University, China. He majors in electrical computer engineering. His current research interests include edge computing and federated learning.



**Yifei Zhu** (Member, IEEE) is currently an Assistant Professor at Shanghai Jiao Tong University, China. He received the B.E. degree from Xi'an Jiaotong University, China, in 2012, and the M.Phil. degree from The Hong Kong University of Science and Technology, China, in 2015, and the Ph.D degree in Computer Science from the Simon Fraser University, Canada, in 2020. His current research interests include edge computing, multimedia networking, distributed machine learning systems, where he published in ACM SIGCOMM, IEEE INFOCOM, ACM Multimedia, and many other venues.



**Xudong Wang** (Fellow, IEEE) received the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology in 2003. He is currently the John Wu and Jane Sun Chair Professor of engineering at the UM-SJTU Joint Institute, Shanghai Jiao Tong University. He is also an affiliate Professor with the Department of Electrical and Computer Engineering, University of Washington. He was a Senior Research Engineer, a Senior Network Architect, and a Research and Development Manager with several companies. He has been actively involved in research and development, technology transfer, and commercialization of various wireless networking technologies. He holds a number of patents on wireless networking technologies and most of his inventions have been successfully transferred to products. His research interests include wireless communication networks, joint communications and sensing, edge computing, and federated learning. He was a voting member of IEEE 802.11 and 802.15 Standard Committees. He was the General Chair of 2017 IEEE 5G Summit in Shanghai and the TPC Co-Chair of the 32nd International Conference on Information Networking. He was the Demo Co-Chair of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM MOBIHOC 2006), the Technical Program Co-Chair of the Wireless Internet Conference (WICON) 2007, and the General Co-Chair of WICON 2008. He was an Editor for IEEE TRANSACTIONS ON MOBILE COMPUTING and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, *Ad Hoc Networks* (Elsevier) and *China Communications*. He was also a Guest Editor of several international journals.

## APPENDIX A PROOF OF THEOREM 1

Our problem  $\Omega$  tries to find a band subset for each HSI in a task to achieve satisfying accuracy and reduce computation and communication costs. Now we consider a simplified offline scenario with only one HSI  $d_i$ , the power constraint, and binary variable in considered. We further make the following assumption: The selection of a band increases  $A_i$  by a fixed number but may vary among different bands. We have the following simplified problem for each HSI image  $d_i$ .

$$\max \quad \Omega' = A_i \quad (22)$$

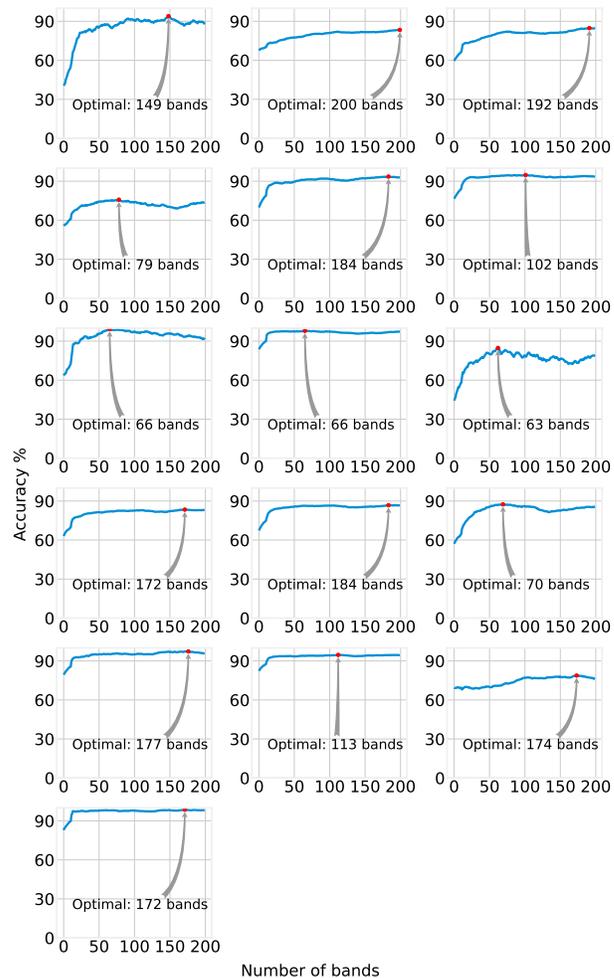
$$s.t. \quad E_{cmp}^i + E_{com}^i \leq C \quad (23)$$

$$x_{i,m} \in \{0, 1\} \quad (24)$$

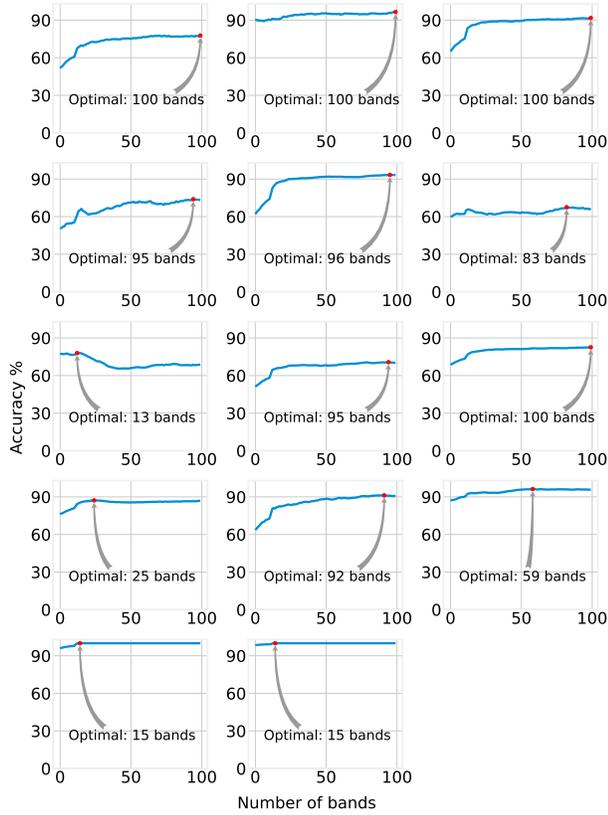
Our simplified version  $\Omega'$  is a 0-1 Knapsack Problem where each band is an "item" with fixed data size as "weight" and classification improvement as "value". The Knapsack problem has been proved to be NP-complete by reducing it to the sub-set sum problem in polynomial time. Since  $\Omega'$  is a simplified version of  $\Omega$ , we can also prove that our original problem is NP-complete.

## APPENDIX B COMPLETE BAND-ACCURACY RESULTS IN INDIAN PINES AND OTHER DATASETS

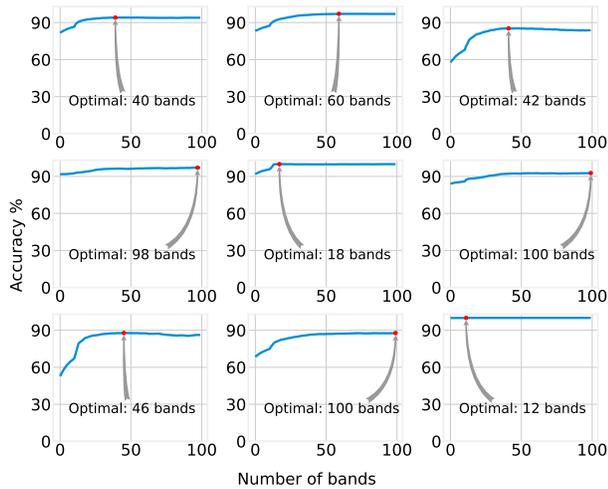
In this part, we provide the complete band-accuracy results of all 16 classes in Indian Pines and the other three datasets to supplement the data used in Section II. It can be seen that our observations in Section II still hold. The sequence of the presented class results here follows the class sequence used in the original dataset. A basic description of these datasets can be found in Table 2.



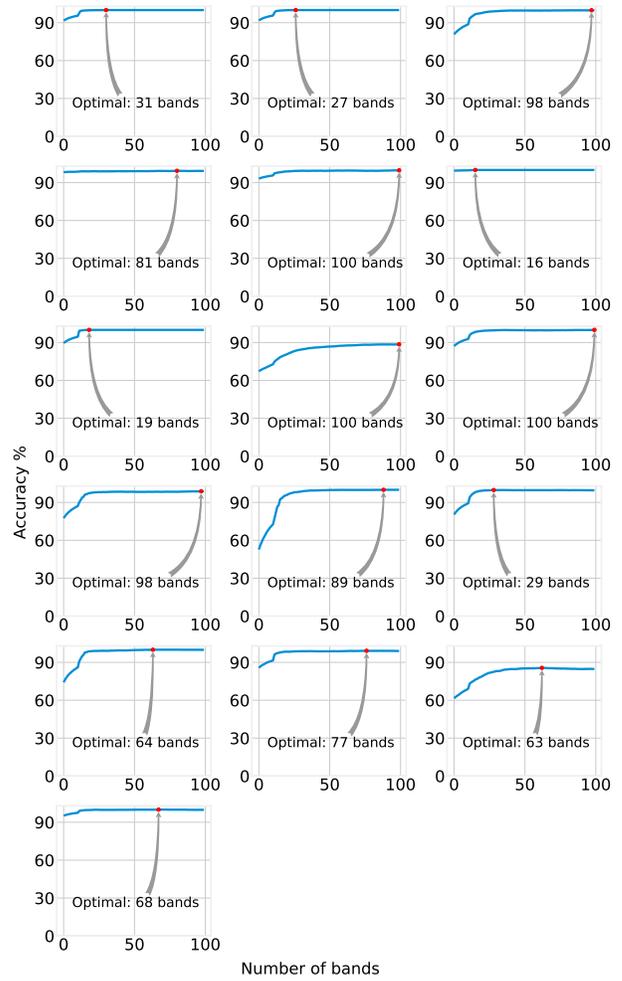
**Fig. 11:** Relationship between the number of selected bands and the accuracy of 16 classes in Indian Pines



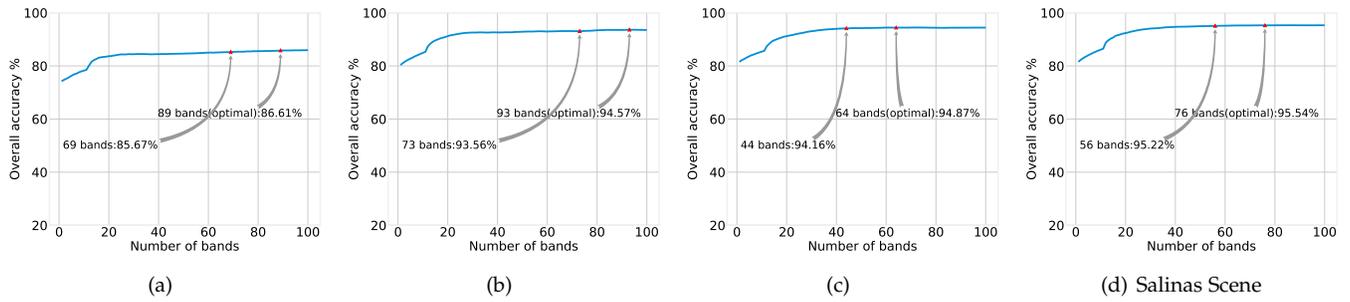
**Fig. 12:** Relationship between the number of selected bands and the accuracy of 14 classes in HyRANK Loukia



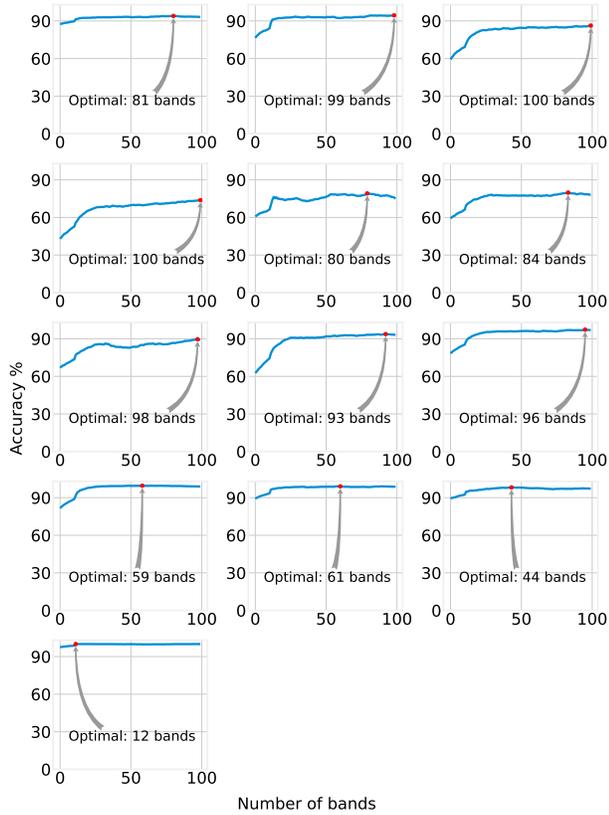
**Fig. 13:** Relationship between the number of selected bands and the accuracy of 9 classes in Pavia University



**Fig. 14:** Relationship between the number of selected bands and the accuracy of 16 classes in Salinas Scene



**Fig. 15:** Anatomy of the classification result on (a)HyRANK Loukia, (b)Kennedy Space Center, (c)Pivia University, (d)Salinas Scene



**Fig. 16:** Relationship between the number of selected bands and the accuracy of 13 classes in Kennedy Space Center