Check for updates

OPEN ACCESS

EDITED BY Xiaobo Zhou, Tianjin University, China

REVIEWED BY Shuohan Liu, Qilu University of Technology, China Cheng Tianhao, Air Force Engineering University, China

*CORRESPONDENCE Afzal Badshah I afzalbadshahkhattak@gmail.com Ali Daud alimsdb@gmail.com

RECEIVED 27 January 2025 ACCEPTED 23 May 2025 PUBLISHED 17 June 2025

CITATION

Badshah A, Daud A, Aboulola OI, Bukhari A, Alshemaimri B and Dawood H (2025) Network optimization by regional computing for UAVs' big data. *Front. Comput. Sci.* 7:1565716. doi: 10.3389/fcomp.2025.1565716

COPYRIGHT

© 2025 Badshah, Daud, Aboulola, Bukhari, Alshemaimri and Dawood. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Network optimization by regional computing for UAVs' big data

Afzal Badshah ^{1*}, Ali Daud ^{2*}, Omar Ibrahim Aboulola³, Amal Bukhari³, Bader Alshemaimri⁴ and Hussain Dawood⁵

¹Department of Software Engineering, University of Sargodha, Sargodha, Pakistan, ²Faculty of Resilience, Rabdan Academy, Abu Dhabi, United Arab Emirates, ³Department of Information Systems and Technology, College of Computer Science and Engineering, University of Jeddah, Jeddah, Saudi Arabia, ⁴Software Engineering Department, College of Computing and Information Sciences, King Saud University, Riyadh, Saudi Arabia, ⁵School of Computing, Skyline University College, Sharjah, United Arab Emirates

Unmanned Aerial Vehicles (UAVs) are increasingly used in sectors such as surveillance, agriculture, and disaster response, generating massive volumes of real-time big data. Traditional cloud computing introduces high latency, while edge computing suffers from limited scalability. This paper proposes a novel three-layer computing framework incorporating a Regional Computing (RC) layer between UAVs and the cloud. A dynamic offloading strategy is designed to select the optimal computing tier based on network conditions and resource availability. To validate the proposal, we used EdgeCloudSim. Simulation results demonstrate that the RC layer reduces end-to-end processing delays by approximately 80%, lowers operational costs by up to $5\times$ compared to cloud computing, and achieves lower task failure rates relative to edge computing. These findings establish Regional Computing as an efficient and scalable solution bridging the gap between edge and cloud paradigms for UAV big data management.

KEYWORDS

cloud computing, edge computing, offloading, UAV (unmanned aerial vehicle), computer network

1 Introduction

Unmanned Aerial Vehicles (UAVs)s are increasingly integrated into various sectors, including aerial surveillance, agricultural monitoring, disaster response, and delivery services (Koubaa et al., 2023). The integration of UAVs has significantly increased data generation due to the growing sophistication of sensors, cameras, onboard computers, and interconnected UAVs technologies. This surge in data, often referred to as UAVs' Big Data (illustrated in Figure 1), presents exciting opportunities to gather valuable insights and improve various aspects of these applications.

UAVs operations heavily depend on a vast array of sensors and high-resolution cameras to perceive and react to their surroundings, consequently generating significant volumes of big data. An average UAVs can produce several gigabytes of data per hour, depending on its sensors and cameras (La Salandra et al., 2024). While a significant portion of this data is processed onboard in real-time, specific information must be communicated with external servers. UAVs need high-resolution maps, environmental conditions, infrastructure details, and data from other UAVs for optimal operation (Wakode, 2021). However, offloading this data to faraway servers can lead to potential delays, hindering the real-time capabilities crucial for many UAVs' applications (Sharma and Mehra, 2023).



UAVs depends on real-time transmission with the cloud for various applications, including downloading high-resolution maps, environmental metadata, and operational reports, while uploading sensor-generated data for analysis. The current network infrastructure constrains the efficient migration, processing and storing of the vast volumes of data generated by UAVs, raising concerns regarding network congestion and latency (Badshah et al., 2022; Alshemaimri et al., 2025). Offloading data to the cloud via public networks exacerbates these issues. Drones, particularly when operating in swarms, generate a very high volume of data in the form of video, images, thermal images, and sensors data during surveillance and search operations, which needs to be processed in real-time.

Existing solutions typically rely on cloud computing for data processing (Koubâa et al., 2020; Kim et al., 2023). However, offloading such large volumes of data to the cloud introduces significant delays and causes network congestion. To address these issues, some studies have proposed joint optimization of cloud and edge computing or processing the data solely on the edge (Yang et al., 2020). However, edge computing has its limitations, particularly in terms of scalability, and struggles to handle highdefinition videos and other large data generated by UAVs. Other studies suggest utilizing drone-based computing power (Ye et al., 2023; Valentino et al., 2018), but this approach significantly drains UAVs batteries, reducing their flight time. Therefore, there is a need to develop an efficient middle layer (as shown in Figure 2) for UAVs big data offloading that considers factors such as delay, network congestion, computation time, and energy consumption. This problem can be explained with one scenario;

For example, during surveillance or search operations, a swarm of UAVs may be deployed to scan an assigned area. The highdefinition video captured by these drones requires high-performance computing to analyze each frame. The results of this analysis may be used as input for further queries to optimize the surveillance or operations. In such cases, this data cannot be efficiently processed on the drone edge or ground base edge due to their limited computing capabilities. Offloading to the cloud is also unsuitable because of high delays and network congestion. Therefore, there is a need for a technique that can handle these issues and optimally process the data.

The widespread adoption of UAVss across numerous industries necessitates further investigation, especially regarding the management of UAVss' Big Data. This article aims to achieve the following key objectives:

- To develop an efficient framework for managing the large volumes of data generated by UAVs on regional servers, ensuring timely processing and storage.
- To design a data transfer strategy that offloads UAVs Big Data to the cloud servers during off-peak hours, enabling comprehensive analysis and support for other UAVs without causing network congestion.
- To ensure minimal response times for queries regarding environmental conditions and other relevant information, despite the substantial data output from UAVs.

This article builds on our previous works (Badshah and Ali, 2024a,b) and presents the following proposals to address the challenges associated with UAVs Big Data:

- Implementing Regional Computing (RC) to minimize delays between servers and UAVs. These servers, strategically placed within specific regions, will process data locally, reducing the need to offload high-volume data to the cloud.
- UAVs sensor data will be temporarily stored on regional servers for initial processing before being transferred to the cloud for in-depth analysis and support during off-peak hours.
- The Regional Computing framework integrates the enforcement of operational regulations and rules. This centralized server would store collective data and analysis, promoting compliance with operational standards and fostering efficient UAVs operations.

Unlike our prior RC approaches in healthcare, education, and social media domains, the present work addresses the unique challenges of UAVs generated big data. In dynamic UAVs environments, data is produced by moving aerial nodes with strict latency requirements and intermittent connectivity. Our proposed architecture introduces a UAVs specific Regional Computing layer that accounts for these factors, offering adaptive offloading and scalability not explored in the earlier domain-specific studies.

The remaining sections are organized as follows: Section 2 reviews related studies. Section 3 explains the methodology used to address the challenges identified in the literature. Section 4 details the experiments conducted to evaluate the effectiveness of the proposed framework. Section 5 discuss in detail the potential and limitations of UAVs, edge, regional and cloud computing paradigm. Finally, Section 6 concludes the study.



Edge computing solutions for UAVs have been explored to address delay-sensitive applications. Authors in Damigos et al. (2023) highlight the limitations of traditional network technologies and propose using a ground-edge server with 5G connectivity as a control station. The research focuses on analyzing real-world latency requirements and practical challenges associated with this approach. The study in Gao et al. (2019) introduces a multi-tier edge-cloud computing framework, utilizing Integer Linear Programming to minimize service time for UAVs. Simulations show this approach significantly reduces service time compared to traditional edge or cloud offloading and scales well with many UAVs. Another article proposes an energyefficient resource allocation scheme for air-ground mobile cloud computing, emphasizing Quality of Experience (QoE) and energy use. It employs a multi-queueing architecture to prioritize high-QoE data and a Lyapunov optimization technique for managing offloading rates and resource allocation, enhancing UAVs satisfaction and energy efficiency (Liu et al., 2020). Additionally, a three-layer network for big data processing in IoT using MEC and UAVs is proposed, featuring an online scheduling algorithm to manage energy and bandwidth at the edge, and a deep reinforcement learning-based path planning algorithm for UAVs to optimize service coverage (Wan et al., 2019).

Although edge computing reduces latency and improves response times, it often faces processing power and resource allocation limitations (Huda and Moh, 2022; Badshah et al., 2021). Therefore, further research is needed to address these issues.

2.3 UAV computing

Research in UAVs computing has focused on various frameworks to leverage the computational power of UAVs. Authors in Yang et al. (2020) propose a hierarchical deep-learning offloading framework for UAVs target tracking, where lower layers of a CNN model run on the UAVs for faster processing and higher layers are handled by a mobile edge server for better accuracy. Another study introduces a UAVs Swarm Edge Cloud Computing model, using the computational power of UAVs swarms (Chen et al., 2019). Similarly, in disaster areas, UAVs swarms have been used to establish networks, utilizing their computing power effectively (Alsamhi et al., 2022).

In another approach, UAVs infrastructure is used as MEC computing (Ye et al., 2023). However, the short battery life of UAVs poses a challenge. To address this, tasks are offloaded from UAVs with weaker batteries to those with freshly charged ones. A similar contribution is made in Valentino et al. (2018), where a swarm of drones processes their data and uses neighboring drones' resources when needed. Furthermore, a mobile cloud computing system using a moving UAVs is introduced, optimizing data transmission and UAVs flight paths to minimize the total energy consumption while meeting application needs (Jeong et al., 2017). Lastly, a unified design approach for onboard processing is proposed, showcasing a three-layer reference model and a working



2 Background

Network resource optimization is crucial for enabling efficient processing of big data generated by UAVss. This section reviews existing research and is categorized into four key areas: Cloud Computing, Edge Computing, UAVs Computing, and Joint Optimization for UAVs big data.

2.1 Cloud computing

Significant attention has been given to cloud computing to process UAVss' big data. Initially, the cloud computing infrastructure was massively used for high-performance computing and storage capabilities (Sulaj et al., 2018). Authors in Luo et al. (2017) propose a cloud-based UAVs system focusing on two primary aspects: how the cloud can efficiently handle data acquisition with minimal impact on UAVs control, and deriving stability conditions considering the data acquisition rate and system stability. Another study in Koubâa et al. (2020) addresses real-time image processing for UAVs within the Internet of Things (IoT) framework. It proposes a system architecture for offloading deep learning tasks to the cloud, demonstrating that cloud offloading achieves superior throughput compared to edge computing, despite larger communication delays.

While cloud computing provides significant processing power, the inherent delays and potential network congestion pose challenges for real-time applications (Namani and Gonen, 2020; Badshah et al., 2020). This calls for new investigations into hybrid approaches that combine cloud computing with other methods to optimize performance and reduce latency. prototype to demonstrate the feasibility of the solution (Lu et al., 2019).

UAVs computing offers flexibility and localized processing capabilities, but battery life and resource limitations remain significant challenges (Ahmed and Jenihhin, 2022). Therefore, further investigations are needed to address the UAVss' big data issues.

2.4 Joint optimization

Joint optimization techniques have been explored to enhance the efficiency of UAVs big data processing. One study proposes a UAVs-Edge-Cloud model with a joint optimization approach to minimize cost and latency, using an online algorithm based on Markov approximation for dynamic environments (Liu et al., 2020). Another paper presents a game-theory-based approach for offloading computation tasks in drones, considering energy, delay, and cost to determine the optimal offloading strategy, outperforming traditional methods by minimizing the cost function comprising energy overhead and delay (Messous et al., 2017).

Additionally, a study proposes scalable aerial computing for handling tasks with varying quality levels and UAVideo for video processing tasks that exceed a UAVs's computing power, streaming video data to ground servers for processing while considering UAVs trajectory, video characteristics, and communication resources (Liu et al., 2021). Lastly, a joint optimization algorithm is used to optimize energy consumption, cost, and delay for UAVs big data (Zhang et al., 2018).

Joint optimization provides a comprehensive approach to addressing multiple performance metrics simultaneously (Song et al., 2022; Badshah et al., 2024). However, the complexity of these models and their scalability need further investigation.

3 Proposed framework

UAVs are widely used in various sectors, with swarms playing a significant role in surveillance, research, and rescue operations. However, UAVs swarms generate massive amounts of data, posing challenges for processing at the UAVs or edge level due to certain limitations. Offloading this data to the cloud leads to longer delays, higher costs, and network congestion. To address these issues, we propose a Regional Computing approach, as illustrated in Figure 3, to efficiently process this data, especially during peak hours. Regional Computing processes and temporarily stores data at the regional level to minimize delay and cost while preventing congestion in the mainstream network.

The proposed structure operates in three layers. The first layer comprises the UAVs, where sensors and cameras generate data. The communication system in this layer offloads the data to regional servers. The second layer, the regional service layer, processes and stores the data, particularly during peak hours. In the case of extremely important data or during off-peak hours when the mainstream network is less congested, this data is offloaded to the cloud. The following subsections provide a detailed discussion of this methodology.

3.1 Communication model

UAVs swarms ($v = \{v_1, v_2, v_3, \dots, v_n\}$) generate significant amounts of data ($l = \{l_1, l_2, l_3, \dots, l_m\}$) during operations. Efficient transmission and processing are crucial to avoid delays and network congestion. The total delay for transmitting data from UAVs to the processing channel is calculated as:

$$D_{total} = D_{tran} + D_{prop} + D_{pr} + D_{sig} + D_{hv}$$
(1)

Where D_{total} represents the total delay, D_{tran} represents the transmission delay, D_{prop} represents the propagation delay, D_{pr} represents the processing delay, D_{sig} represents the delay due to signal strength, and D_{hv} represents the base station handover delay.

The transmission delay from UAVs to Base Station (D_{tran}^{bs}) can be calculated as:

$$D_{tran}^{\rm bs} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (\nu_i, l_j)}{S_{tran}^{\rm bs}}$$
(2)

Here, S_{tran}^{bs} represents the speed of wireless transmission. Where the v_i shows the UAVs and l_j the data generated from these UAVs. The propagation delay for data sent through 5G base stations (D_{prop}^{bs}) is determined by:

$$D_{prop}^{bs} = \frac{Dis_{i,BS}}{c}$$
(3)

where $Dis_{i,BS}$ is the distance between UAV *i* and the base station, and *c* is the speed of light.

The transmission delay base station to Regional Computing (D_{tran}^{rc}) can be calculated as:

$$D_{tran}^{\rm rc} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (v_i, l_j)}{S_{tran}^{\rm rc}}$$
(4)

Here, S_{tran}^{rc} represents the speed of wired transmission.

The propagation delay for data sent from the base station to the regional computing server (D_{prop}^{rc}) is determined by:

$$D_{prop}^{\rm rc} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (v_i, l_j)}{S_{prop}^{\rm rc}}$$
(5)

Where *S*^{rc}_{*prop*} denotes the wired propagation speed. This speed is calculated as:

$$S_{prop}^{\rm rc} = \frac{Dis_{\rm bs,rc}}{B_{\rm rc}} \tag{6}$$

Here, $Dis_{bs,rc}$ is the distance between the base station and the regional computing server, and B_{rc} is the cable bandwidth.

The total time for processing this data (D_{pr}) is calculated as:

$$D_{pr} = \left(\frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (\nu_i, l_j)}{S_{pr}}\right)$$
(7)



Here, S_{pr} indicates processing speed. The processing speed is calculated as:

$$S_{pr} = \frac{L}{R} \tag{8}$$

Where *L* is the workload and *R* is the processing rate. The delay due to signal strength is given by:

$$D_{sig} = \frac{D_{rate}}{R_{eff}} \times D_{tran} \tag{9}$$

Where D_{rate} is the nominal data rate, R_{eff} is the effective data rate considering Bit Error Rate (BER), and D_{tran} is the nominal transmission delay.

Signal strength (SNR) is calculated as:

$$SNR (dB) = 10 \log_{10} \left(\frac{P_{signal}}{P_{noise}} \right)$$
(10)

The delay due to base station handover is given by:

$$D_{h\nu} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (\nu_i, l_j)}{S_{h\nu}}$$
(11)

Where $D_{h\nu}$ is the handover delay and $S_{h\nu}$ is the speed of handover.

Therefore, the total delay for *n* UAVs and *m* workloads, as per the Equations 2-5, 7, 9, 11 is given by:

$$D_{total} = \left(\frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (v_i, l_j)}{S_{tran}^{bs}}\right) + \left(\frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (v_i, l_j)}{S_{prop}^{bs}}\right) + \left(\frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (v_i, l_j)}{S_{tran}^{rc}}\right) + \left(\frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (v_i, l_j)}{S_{prop}^{rc}}\right) + \left(\frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (v_i, l_j)}{S_{pr}}\right) + \left(\frac{D_{rate}}{R_{eff}} \times D_{tran}\right) + \left(\frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (v_i, l_j)}{S_{hv}}\right)$$

$$(12)$$

The congestion probability is given by:

$$Con_{Prob} \approx \frac{L_{total}}{B_{avail}}$$
 (13)

Where Con_{Prob} refers to congestion probability, L_{total} is the total traffic on the network, and B_{avail} represents available bandwidth. Network congestion likelihood rises with total traffic and declines with available bandwidth. The proliferation of cloud services and increasing UAVs usage will escalate traffic, leading to network congestion.

Since regional computing servers are deployed geographically closer to the UAVs compared to distant cloud servers, the

propagation distance, and thus propagation delay, is substantially reduced as per the Equations 1, 3, 5, 12. This reduction stems from regional computing servers being situated within the region. Furthermore, reliance on cloud-based data or devices decreases with the use of regional servers, as shown by the congestion probability analysis, Equation 13. Minimal workload routed to the main network ensures bandwidth availability for real-time application data, enhancing overall system performance and responsiveness. The flow of data in the regional computing is shown in as shown in Figure 4.

3.2 Energy model

Likewise, to delay, energy consumption is directly proportional to distance. The energy consumption in the UAVs big data communication setup can be computed as follows:

$$\mathcal{E}_{\text{tran}} = \sum_{i=1}^{n} \left(\mathcal{E}_{\text{tran}}^{\text{bs}}(i) + \mathcal{E}_{\text{tran}}^{\text{rc}}(i) \right)$$
(14)

Here, $\mathcal{E}_{tran}^{bs}(i)$ denotes the energy utilization for wireless transmission from UAVs *i* to the base station, and $\mathcal{E}_{tran}^{rc}(i)$ denotes the energy utilization for wired transmission from the base station to the regional computing server.

Where

$$\mathcal{E}_{\text{tran}}^{\text{bs}}(i) \approx Dis_{i,\text{bs}} \cdot P_i^{\text{bs}} \cdot T_{i,\text{bs}}$$
(15)

and

$$\mathcal{E}_{\text{tran}}^{\text{rc}}(i) \approx Dis_{\text{bs,rc}} \cdot P_{\text{bs}}^{\text{rc}} \cdot T_{\text{bs,rc}}$$
 (16)

Here, $Dis_{i,bs}$ represents the distance between UAVs *i* and the base station, P_i^{bs} represents the power consumption for wireless transmission by UAVs *i*, $T_{i,bs}$ represents the transmission time from UAVs *i* to the base station. Similarly, $Dis_{bs,rc}$ represents the distance between the base station and the regional computing server, P_{bs}^{rc} represents the power consumption for wired transmission by the base station, and $T_{bs,rc}$ represents the transmission time from the base station to the regional computing server.

These equations illustrate that energy utilization increases with distance, which is also directly proportional to cost.

$$\mathcal{E}_{\text{other}} = \mathcal{E}_{\text{pro}} + \mathcal{E}_{\text{stor}} + \mathcal{E}_{\text{col}} + k \tag{17}$$

Similarly, \mathcal{E}_{other} represents the energy used in various activities, including computing (\mathcal{E}_{pro}), storage (\mathcal{E}_{stor}), and cooling data centers (\mathcal{E}_{col}).

The total energy utilization can be calculated as follows:

$$\mathcal{E}_{\text{total}} = \mathcal{E}_{\text{tran}} + \mathcal{E}_{\text{other}} \tag{18}$$

Here, \mathcal{E}_{total} shows the total energy usage, while \mathcal{E}_{tran} denotes the total energy consumption on data offload.

It is also known that,

$$Cost_{oper} \propto \mathcal{E}$$
 (19)

There is a direct correlation between a UAVs's power consumption (denoted by \mathcal{E} and its operational costs (*Cost*_{oper}). In other words, as a UAVs consumes more power, its operational costs tend to rise.

The Equations 15, 16 presented above affirm that energy consumption escalates with increasing distance. Additionally, a higher workload congests the network and processors, resulting in additional heat generation, which in turn consumes more energy (as shown in Equation 19). Consequently, based on the aforementioned energy model, it can be concluded that cloud computing (CC) demands greater energy consumption. Conversely, when the workload is processed by regional computing (RC), the energy requirements decrease significantly.

3.3 Cost model

In previous sections, we have observed that delay, congestion, and energy consumption increase as the distance between UAVs and processing units expands. This section focuses on the cost behavior with respect to these parameters for UAVss' big data. The total cost can be computed using the following formula:

$$Cost_t = Cost_{tr} + Cost_p + Cost_{pr} + Cost_{col}$$
(20)

where $Cost_t$ represents the total cost, $Cost_{tr}$ denotes the transmission cost, $Cost_p$ is the propagation cost, $Cost_{pr}$ refers to the processing cost, and $Cost_{col}$ accounts for the cooling cost.

The transmission cost, $Cost_{tr}$, includes two components: the cost associated with transmitting data from UAVs to base stations and the cost from base stations to regional computing servers.

For the transmission from UAVs to base stations, the cost can be expressed as:

$$C_{\text{trn, bs}} = \sum_{i=1}^{n} \sum_{j=1}^{m} \left(C_{\text{rate, v}} \cdot L_{i,\text{base}} w_{j} + C_{\text{time, v}} \cdot T_{i,\text{base}} + C_{\text{energy, v}} \cdot E_{i,\text{base}} + C_{\text{bandwidth, v}} \cdot B_{i,\text{base}} + C_{\text{fixed, v}} \right)$$

$$(21)$$

In this formula, *n* represents the number of UAVs and *m* denotes the number of workload tasks generated by these UAVs. $C_{\text{rate, v}}$ denotes the cost per unit of data transmitted, while $L_{i,\text{base}}w_j$ is the amount of data from UAVs *i* for workload *j*. $C_{\text{time, v}}$ represents the cost per unit of time for this transmission, with $T_{i,\text{base}}$ indicating the time required. $C_{\text{energy, v}}$ signifies the cost per unit of energy consumed, and $E_{i,\text{base}}$ denotes the energy used in the transmission from UAVs to base stations. Additionally, $C_{\text{bandwidth, v}}$ is the cost per unit of bandwidth utilized, while $B_{i,\text{base}}$ represents the bandwidth used. $C_{\text{fixed, UAV}}$ is the fixed cost associated with each transmission event from UAVs.



For the transmission from base stations to regional computing servers, the cost is:

$$C_{\text{trn, rc}} = \sum_{i=1}^{n} \sum_{j=1}^{m} \left(C_{\text{rate, bs}} \cdot l_{\text{base,rc}} w_j + C_{\text{time, bs}} \cdot T_{\text{base,rc}} + C_{\text{energy, BS}} \cdot E_{\text{base,rc}} + C_{\text{B, bs}} \cdot B_{\text{bs,rc}} + C_{\text{fixed, bs}} \right)$$
(22)

In this formula, $C_{\text{rate, BS}}$ represents the cost per unit of data transmitted from base stations, and $d_{\text{bs,rc}}l_j$ is the amount of data for workload *j*. The cost per unit of time for this transmission is $C_{\text{time, bs}}$, with $T_{\text{bs,rc}}$ being the time taken. $C_{\text{energy, bs}}$ denotes the cost per unit of energy consumed during this transmission, and $E_{\text{bs,rc}}$ is the energy used in the transmission from base stations to regional computing servers. $C_{\text{B, bs}}$ is the cost per unit of bandwidth used, while $B_{\text{bs,rc}}$ represents the bandwidth utilized. Lastly, $C_{\text{fixed, bs}}$ is the fixed cost associated with each transmission event from base stations.

The total transmission cost can be expressed as:

$$C_{\rm trn} = C_{\rm trn,\,bs} + C_{\rm trn,\,rc} \tag{23}$$

The propagation cost, $Cost_p$, is computed based on the distances involved and is expressed as:

$$C_{\text{prop}} = \sum_{i=1}^{n} \left(C_{\text{dis, UAV}} \cdot Dis_{i,\text{bs}} + C_{\text{dis, bs}} \cdot Dis_{\text{bs,rc}} \right)$$
(24)

In this formula, $C_{\text{dis, UAV}}$ is the cost per unit of distance for UAV-to-base station transmission, while $Dis_{i,\text{bs}}$ is the distance from UAVs *i* to base stations. $C_{\text{dis, bs}}$ denotes the cost per unit of distance for base station-to-regional computing transmission, with $Dis_{\text{bs,rc}}$ representing the distance involved in this stage.

The processing cost *Cost*_{pr} is calculated as:

$$C_{\rm prc} = \sum_{i=1}^{n} \sum_{j=1}^{m} \left(C_{\rm uprc} \cdot T_{\rm prc_{i,j}} \right) \tag{25}$$

Here, $T_{\text{prc}_{i,j}}$ represents the processing time for data from UAVs *i* and workload *j*, and C_{uprc} denotes the processing time cost per unit.

This comprehensive cost model addresses the costs associated with transmitting data from UAVs to base stations, from base stations to regional computing servers and the associated propagation and processing costs.

The cost model equations illustrate the significant impact of distance on transmission and propagation costs, as well as processing time on overall expenses (as shown in Equations 21, 22, 25). Cloud computing tends to incur higher costs due to the increased distances involved in data transmission. In contrast, regional computing provides cost efficiencies by minimizing these distances and optimizing processing tasks, thereby making RC a more cost-effective solution for managing UAVs big data workloads.

The proposed offloading strategy dynamically determines the optimal processing destination for each UAV-generated workload based on real-time evaluation of system parameters, as formalized in Algorithm 1.

For each workload l_j generated by UAV v_i , the system first calculates the total delay D_{total} according to Equation 12, energy consumption E_{total} based on Equation 18, overall cost Cost_t using Equation 20, and congestion probability ConProb following Equation 13. These parameters are then compared against predefined system thresholds: D_{thresh} for acceptable delay and Con_{thresh} for congestion tolerance.

The decision-making process proceeds by first checking whether the calculated total delay D_{total} is within the acceptable threshold and the congestion probability ConProb is higher than the congestion threshold. If both conditions are satisfied, the workload is offloaded to the Regional Computing (RC) layer, taking advantage of its proximity and moderate resource capacity to reduce network congestion and maintain low latency.

If the workload does not satisfy the first condition, the system then examines the temporal state of the network. Specifically, it checks whether the current time falls within off-peak network hours, where traffic is typically lower. If the request occurs during off-peak hours and sufficient processing capacity is available at the RC servers, the workload is offloaded to the RC layer to achieve energy and cost savings.

However, if the RC layer lacks sufficient resources or if the request arrives during peak traffic periods, the workload is directly offloaded to the Cloud Computing (CC) layer to ensure that it is processed without overloading local resources or increasing queuing delays.

1	Input: UAVs $v = \{v_1, v_2, v_3,, v_n\}$ and their			
	respective workloads $l = \{l_1, l_2, l_3, \dots, l_m\}$			
2	Output: Decision on optimal processing tier: UAV,			
	Regional Computing (RC), or Cloud Computing (CC)			
3	foreach workload l_j generated by UAV v_i do			
4	Calculate total delay D _{total} using Equation (12)			
5	Calculate energy consumption E _{total} using			
	Equation (18)			
6	Calculate overall cost Cost _t using			
	Equation (20)			
7	7 Calculate congestion probability ConProb using			
	Equation (13)			
8	s if $D_{total} \leq D_{thresh}$ AND ConProb \geq Con _{thresh} then			
9	\bullet Offload I_i to Regional Computing (RC) to			
	minimize delay and congestion			
10	else			
11	if Current time is in off-peak hours then			
12	if Sufficient processing capacity available			
	at RC then			
13	Offload <i>l_j</i> to Regional Computing (RC)			
	to reduce cost and energy consumption			
14	else			
15	Offload l_j to Cloud Computing (CC)			
16	end			
17	else			
18	Offload <i>l_j</i> directly to Cloud Computing			
	(CC)			
19	end			
20	end			
21	${f if}$ workload l_j is required for cross-regional			
	accessibility or global data aggregation then			
22	Migrate l_j to Cloud Computing (CC)			
23	3 end			
24	24 end			

Algorithm 1. Dynamic Offloading Strategy for UAV Big Data Based on Delay, Energy, Cost, and Congestion

Additionally, regardless of the above conditions, if a workload is identified as requiring global accessibility, such as when the data must be made available across multiple regions or aggregated at a global level, the workload is compulsorily migrated to the CC layer, bypassing local decision policies.

The complete logical sequence of these offloading decisions is detailed in Algorithm 1, while the operational flow is visually summarized in Figure 5.

4 Evaluation

To assess the effectiveness of the proposed framework, we utilized EdgeCloudSim (2023). This advanced simulation tool is tailored for analyzing the performance of edge computing infrastructures and applications. By building on the popular CloudSim (CloudSlab, 2023) framework, EdgeCloudSim addresses the specific demands of edge computing, such as user proximity,



TABLE 1 Simulation parameters and settings.

Parameters	Description
Quantity of cloud servers	1
Quantity of regional servers	1
Quantity of edge servers	1
Network topology	Wireless/LAN/WAN
Minimum UAVs	5
Maximum UAVs	100
Simulation time	30 Min
Delay	Round trip delay
Cost	Communication and processing cost
Simulation repetition for average calculation	10 times

low latency, and distributed processing. It offers a robust platform for modeling diverse edge and cloud scenarios, enabling researchers and developers to simulate and study the performance of their applications across various network conditions, resource management strategies, and workload distributions.

4.1 Experimental setup

The experimental setup outlined in this study includes four distinct scenarios to handle UAVss' big data. First, we evaluate the performance and cost associated with offloading UAVs big data to cloud computing servers. Second, we examine the performance and cost of offloading UAVs big data to edge computing servers. Third, we assess the performance and cost of the proposed regional computing solution. Fourth, we include a UAVs computing scenario to test performance, simulating a network of UAVs collecting and processing large volumes of data in real-time. The simulation parameters are shown in Table 1.

4.1.1 Cloud context

In the initial setup of the cloud computing context, a single cloud data center was established to handle UAVss data. The server was configured to support 2,000 UAVs and manage 300,000 tasks globally. The server setup included 36 cores, 16,000 MIPS, 16,000 MB of RAM, and 200,000 MB of storage capacity. Within this host, eight virtual machines (VMs) were deployed, each equipped with 4 cores, 10,000 MIPS, 4,000 MB of RAM, and 50,000 MB of storage. It was conducted ten times, with each session lasting half an hour, to ensure robustness and reliability. Subsequently, the average delay and cost associated with data transfer and processing were calculated.

4.1.2 Edge context

In the initial setup for edge computing, a single-edge data center was established. The server configuration included 4 cores, 6,000 MIPS, 4,000 MB of RAM, and 100,000 MB of storage capacity. Within this host, four VMs were deployed, each equipped with a single core. The setup can involve a range of UAVss up 2,00. It was conducted ten times, with each session lasting half an hour, to ensure robustness and reliability. Subsequently, the average delay and cost associated with data transfer and processing were calculated.

4.1.3 Regional context

In the setup for regional computing, a single regional data center was established. The server configuration included 8 cores, 10,000 MIPS, 16,000 MB of RAM, and 150,000 MB of storage capacity. Within this host, four VMs were deployed, each equipped with 2 cores. The setup can involve a range of UAVss up to 500. It was conducted ten times, with each session lasting half an hour, to ensure robustness and reliability. Subsequently, the average delay and cost associated with data transfer and processing were calculated.

4.1.4 UAV computing context

In the UAVs computing context, each UAVs was responsible for processing its own data independently. The UAVss were equipped with onboard computing resources to handle data processing tasks locally. This scenario involved a range of UAVss from a minimum of 5 to a maximum of 100. The experiment was conducted ten



times, with each session lasting half an hour, to ensure robustness and reliability. The average delay and cost associated with data processing within each UAVs were then calculated to assess the effectiveness of this decentralized approach.

4.2 Result analysis

The results analysis section is organized into five subsections, which include delay, resource utilization, task failure, performance, and cost comparison.

4.2.1 Resources utilization

Figure 6 illustrates the resources utilization comparison of UAVs computing, edge computing, cloud computing, and regional computing.

UAVs onboard computing shows a progressive increase in utilization, starting from 16.6% and reaching up to 166% as the number of tasks increases. While UAVs onboard computing has almost no network delay, however, the resources are quickly overutilized, making it infeasible to process large volumes of UAVss' big data. Additionally, UAVs have limited battery life, and overutilization will further reduce their flight time, compounding the challenge.

Edge computing utilization ranges from 2.29% to 94%, with a rapid increase as task demands grow. This high utilization rate results from edge computing's limited resource capacity, leading to potential overutilization and performance degradation. The initial low utilization rates rise sharply, highlighting the strain on edge resources under increasing task demands. Cloud computing, on the other hand, demonstrates more stable and lower utilization rates, ranging from 0.16% to 1%. This lower utilization is attributed to the vast and scalable resources available in cloud data centers. Regional computing exhibits utilization rates similar to edge computing,



ranging from 0.56% to 1.4%. This moderate utilization rate indicates that regional computing can effectively manage resources to avoid the high overutilization observed in edge computing, while still offering lower latency compared to cloud computing.

The comparison indicates that regional computing strikes a balance between resource utilization and delay, offering lower latency compared to cloud computing while maintaining utilization rates comparable to edge and UAVs's computing. This positions regional computing as superior in scenarios where both efficient resource utilization and reduced latency are crucial.

4.2.2 Task failure

Figure 7 illustrates the task failure comparison among UAVs computing, edge computing, cloud computing, and regional computing. As observed in the delay and server utilization sections, edge computing's limited resources increase its susceptibility to overutilization, potentially leading to task failures under heavy loads.

UAVs computing exhibits a progressive increase in task failures, starting from 5% and reaching up to 50% as the number of tasks increases. While UAVs onboard computing has almost no network delay, the overutilization of resources leads to a high number of task failures. Additionally, the limited battery life of UAVs further exacerbates this issue, making UAVs onboard computing unsuitable for processing large volumes of UAVs big data.

Task failure rates on edge computing range from 0.47% to 14%. The escalating failure rates align with the rising server utilization, highlighting a significant challenge in edge computing where resource constraints can compromise reliability during periods of high demand. In contrast, task failure rates on cloud computing remain consistently low, ranging from 0% to 0.27%. The robust infrastructure and scalability of cloud data centers ensure minimal task failures even under significant workloads. However, this reliability is achieved at the expense of higher latency compared to edge and regional computing solutions.



Regional computing emerges as a viable compromise among UAVs, edge and cloud computing in terms of task failure rates. Failure rates on regional computing fall between 0.01% and 0.29%, indicating a moderate level of reliability even under increasing workloads and server utilization. This suggests that regional computing can effectively manage task failures while offering lower latency than cloud computing and mitigating the over-utilization issues encountered in edge computing.

4.2.3 Processing time

Figure 8 illustrates the processing time comparison among UAVs computing, edge computing, cloud computing, and regional computing. As observed in the previous sections, UAVs and edge computing's tendency to become overutilized can lead to an increase in processing time.

UAVs computing shows a progressive increase in processing time, starting from 7.34 ms and reaching up to 73.4 ms as the number of tasks increases. While UAVs onboard computing experiences almost no network delay, the processing time increases significantly due to overutilization of resources. This highlights the limitations of UAVs in handling large volumes of big data, exacerbated by their limited battery life.

Edge computing processing time ranges from 1.01 ms to 5.9 ms. The processing time increases as server utilization rises, indicating the impact of overutilization on processing efficiency. This underscores a significant drawback of edge computing, where resource constraints can degrade processing performance under heavy workloads. In contrast, cloud computing demonstrates consistently low processing times, ranging from 0.05 ms to 0.07 ms. The vast resources and scalability of cloud data centers ensure efficient processing even under significant workloads, although this comes at the cost of higher latency compared to the edge and regional computing solutions.

Regional computing emerges as the optimal choice among, UAVs, edge and cloud computing in terms of processing time.



Processing times on regional computing range from 1 ms to 2 ms, offering a balance between the low latency of edge computing and the efficient resource utilization of cloud computing. By effectively managing processing tasks without leading to overutilization, regional computing provides reliable and consistent processing performance, making it a suitable solution for a wide range of applications.

4.2.4 Cost

Figure 9 illustrates the cost comparison for processing data from 200 to 2,000 UAVss. The cost model used in this section accounts for transmission, processing, propagation, and cooling expenses. Since UAVs and edge computing utilize minimal resources, its cost is negligible for this comparison. Therefore, the focus is on comparing the costs of cloud and regional computing.

As shown in the figure, the cost of cloud computing ranges from 0.52 \$ to 5.36 \$. This higher cost is attributed to transmission expenses and the significant electricity required for running and cooling large data centers. In contrast, regional computing costs range from 0.08 to 1.14, making it a more cost-effective option.

4.2.5 Delay

Figure 10 illustrates the delay comparison among UAVs edge, edge computing, regional computing, and cloud computing.

The UAVs edge shows no delay, maintaining a consistent value of 0 ms across all measurements. This is because there is no distance or hop to pass data, resulting in minimal delay. However, UAVss cannot process large volumes of data, making them unsuitable for big data applications despite the minimal delay.

Edge computing delay starts at 0.062 ms and increases to 0.21 ms as the number of tasks grows. This gradual increase highlights how server utilization affects delay. While edge computing maintains minimal delay, it faces scalability issues, limiting its effectiveness under heavy workloads.



Regional computing experiences delay values close to those of edge computing, starting from 0.95 ms and reaching up to 1.5 ms. Although the delay is slightly higher than edge computing, regional computing can offer scalability similar to cloud computing. This makes regional computing the best choice, balancing low latency and scalability.

Cloud computing exhibits the highest delay, starting at 8.95 ms and increasing to 8.21 ms. Despite the extensive resources available in cloud data centers, the significant network delay makes cloud computing less suitable for applications requiring real-time processing.

In summary, UAVs edge computing demonstrates no delay but is unsuitable for big data processing. Edge computing offers minimal delay but struggles with scalability. Regional computing provides a good balance with low delay and scalability akin to cloud computing, making it the optimal choice. Cloud computing, despite its high processing power, faces challenges due to substantial network delay.

5 Discussion

The analysis of UAVs, edge, cloud, and regional computing paradigms reveals distinct potential and limitations for each, with regional computing emerging as the most balanced and effective solution, as we can see in Table 2.

5.1 UAV computing

UAVs computing, characterized by its on-board processing capabilities, initially benefits from minimal network delay, making it well-suited for real-time data analysis and immediate decisionmaking during operations. However, as task demands increase, UAVs computing faces significant challenges. The utilization of resources on UAVs rises dramatically from 16.6% to 166%, which leads to severe overutilization. This overutilization compromises

TABLE 2 Comparison of computing paradigms.

Paradigm	Potentials	Limitations
UAV computing	Low network delay	Resource overutilization, high task failure rate
Edge computing	Reduced latency	High utilization, high task failure rate, limited scalability
Cloud computing	Scalability, low processing time, minimal task failures	High latency, high costs
Regional computing	Balanced utilization and latency, moderate task failure rates, cost-effective	Limited scalability compared to cloud

the UAVs's ability to handle large volumes of data and further shortens its battery life, limiting its effectiveness for extensive data processing tasks. Additionally, the task failure rate escalates considerably, reaching up to 50% as the number of tasks grows. This high failure rate, combined with the limited battery life and constrained resources, renders UAVs computing unsuitable for processing large data efficiently.

5.2 Edge computing

Edge computing provides reduced latency by processing data closer to its source, which is advantageous for applications requiring real-time responses. Despite this, edge computing faces notable limitations. Resource constraints at the edge lead to utilization rates that range from 2.29% to 94%, indicating that the edge infrastructure often becomes overutilized under heavy workloads. This overutilization results in increased task failure rates, which range from 0.47% to 14%. The limited scalability of edge computing further exacerbates these issues, as the fixed resources can become a bottleneck when scaling up to handle larger volumes of tasks. Consequently, while edge computing offers low latency, its resource limitations and high task failure rates pose significant challenges.

5.3 Cloud computing

Cloud computing excels in scalability and efficiency, with utilization rates consistently low, ranging from 0.16% to 1%. The extensive and scalable resources available in cloud data centers ensure that processing times remain minimal, between 0.05 ms and 0.07 ms, even under substantial workloads. Furthermore, the cloud infrastructure provides minimal task failure rates, from 0% to 0.27%, due to its scalable capabilities. However, cloud computing comes with its drawbacks. The latency associated with data being processed in remote data centers is higher compared to the edge and regional computing. Additionally, the cost of cloud computing is considerably high, ranging from 0.52 to 5.36, driven by expenses related to the transmission, processing, and cooling of large data centers. Thus, while cloud computing is highly efficient and reliable, its higher latency and cost are significant factors to consider.

5.4 Regional computing

Regional computing emerges as the most balanced and effective solution among the paradigms analyzed. It offers a middle ground by providing moderate resource utilization rates, ranging from 0.56% to 1.4%, and processing times that fall between 1 ms and 2 ms. This balance ensures that regional computing manages resources effectively, avoiding the overutilization issues seen in edge computing while delivering lower latency compared to cloud computing. Task failure rates in regional computing are also moderate, ranging from 0.01% to 0.29%, indicating reliable performance even with increasing workloads. Additionally, regional computing is more cost-effective, with costs ranging from 0.08 to 1.14, making it a more economical option compared to cloud computing. Although regional computing may not scale as extensively as cloud solutions, it provides a well-rounded solution that balances efficiency, performance, and cost-effectiveness.

In summary, while each computing paradigm has its specific strengths and limitations, regional computing stands out as the optimal choice for many applications. It offers an effective compromise between resource utilization, latency, reliability, and cost, making it a versatile and practical solution for diverse computational needs.

6 Conclusion

In this study, we addressed the growing challenge of realtime data processing and storage for UAVss' big data. The everincreasing adoption of UAVss' leads to a surge in data generation, exceeding the capabilities of onboard computing and introducing latency issues with cloud storage due to network congestion. We proposed a regional computing framework as a solution. This framework leverages regional servers to process UAVs data during peak hours, alleviating onboard processing limitations and network congestion. Subsequently, the processed data is offloaded to the cloud for later use or in-depth analysis during off-peak hours, optimizing resource utilization. The results demonstrate the effectiveness of regional computing in minimizing processing delays and costs while maximizing network utilization. This approach offers a promising solution for managing the evergrowing volume of UAVs data, paving the way for more efficient and reliable UAVs operations.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding authors.

Author contributions

ABa: Conceptualization, Methodology, Visualization, Writing – original draft. AD: Project administration, Writing – review & editing. OA: Formal analysis, Resources, Software, Writing – review

& editing. ABu: Data curation, Investigation, Validation, Writing – review & editing. BA: Data curation, Validation, Visualization, Writing – review & editing. HD: Validation, Writing – review & editing.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This research was supported by the Ongoing Research Funding program, (ORF-2025-1124), King Saud University, Riyadh, Saudi Arabia.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships

References

Ahmed, F., and Jenihhin, M. (2022). A survey on UAV computing platforms: a hardware reliability perspective. *Sensors* 22:6286. doi: 10.3390/s22166286

Alsamhi, S. H., Shvetsov, A. V., Kumar, S., Shvetsova, S. V., Alhartomi, M. A., Hawbani, A., et al. (2022). UAV computing-assisted search and rescue mission framework for disaster and harsh environment mitigation. *Drones* 6:154. doi: 10.3390/drones6070154

Alshemaimri, B., Badshah, A., Daud, A., Bukhari, A., Alsini, R., and Alghushairy, O. (2025). Regional computing approach for educational big data. *Sci. Rep.* 15:7619. doi: 10.1038/s41598-025-92120-7

Badshah, A., and Ali, D. (2024a). Network resources optimization through regional computing for vehicular big data. *IEEE Access* 2024, 1–13.

Badshah, A., and Ali, D. (2024b). Smart cities' big data: performance and cost optimization with regional computing. *IEEE Access* 12, 128896–128908. doi: 10.1109/ACCESS.2024.3457269

Badshah, A., Daud, A., Khan, H. U., Alghushairy, O., and Bukhari, A. (2024). Optimizing the over and underutilization of network resources during peak and off-peak hours. *IEEE Access* 12, 82549–82559. doi: 10.1109/ACCESS.2024.3402396

Badshah, A., Ghani, A., Irshad, A., Naqvi, H., and Kumari, S. (2021). Smart workload migration on external cloud service providers to minimize delay, running time, and transfer cost. *Int. J. Commun. Syst.* 34:e4686. doi: 10.1002/dac.4686

Badshah, A., Ghani, A., Shamshirband, S., Aceto, G., and Pescapé, A. (2020). Performance-based service-level agreement in cloud computing to optimise penalties and revenue. *IET Commun.* 14, 1102–1112. doi: 10.1049/iet-com.2019.0855

Badshah, A., Iwendi, C., Jalal, A., Hasan, S. S. U., Said, G., Band, S. S., et al. (2022). Use of regional computing to minimize the social big data effects. *Comput. Ind. Eng.* 171:108433. doi: 10.1016/j.cie.2022.108433

Chen, W., Liu, B., Huang, H., Guo, S., and Zheng, Z. (2019). When UAV swarm meets edge-cloud computing: the QoS perspective. *IEEE Netw.* 33, 36–43. doi: 10.1109/MNET.2019.1800222

Cloudslab (2023). Cloudsim.

Damigos, G., Lindgren, T., and Nikolakopoulos, G. (2023). Toward 5G edge computing for enabling autonomous aerial vehicles. *IEEE Access* 11, 3926–3941. doi: 10.1109/ACCESS.2023.3235067

EdgeCloudSim (2023). *Edgecloudsim*. Available online at: https://github.com/CagataySonmez/EdgeCloudSim (accessed August 2, 2024).

Gao, A., Hu, Y., Liang, W., Lin, Y., Li, L., and Li, X. (2019). A QoE-oriented scheduling scheme for energy-efficient computation offloading in UAV cloud system. *IEEE Access* 7, 68656–68668. doi: 10.1109/ACCESS.2019.2919290

Huda, S. A., and Moh, S. (2022). Survey on computation offloading in UAV-enabled mobile edge computing. J. Netw. Comput. Applic. 201:103341. doi: 10.1016/j.jnca.2022.103341

Jeong, S., Simeone, O., and Kang, J. (2017). Mobile edge computing via a UAVmounted cloudlet: optimization of bit allocation and path planning. *IEEE Trans. Vehic. Technol.* 67, 2049–2063. doi: 10.1109/TVT.2017.2706308 that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that no Gen AI was used in the creation of this manuscript.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Kim, B., Jang, J., Jung, J., Han, J., Heo, J., and Min, H. (2023). A computation offloading scheme for UAV-edge cloud computing environments considering energy consumption fairness. *Drones* 7:139. doi: 10.3390/drones7020139

Koubaa, A., Ammar, A., Abdelkader, M., Alhabashi, Y., and Ghouti, L. (2023). Aero: AI-enabled remote sensing observation with onboard edge computing in UAVs. *Remote Sens.* 15:1873. doi: 10.3390/rs15071873

Koubâa, A., Ammar, A., Alahdab, M., Kanhouch, A., and Azar, A. T. (2020). Deepbrain: experimental evaluation of cloud-based computation offloading and edge computing in the internet-of-drones for deep learning applications. *Sensors* 20:5240. doi: 10.3390/s20185240

La Salandra, M., Nicotri, S., Donvito, G., Italiano, A., Colacicco, R., Miniello, G., et al. (2024). A paradigm shift in processing large UAV image datasets for emergency management of natural hazards. *Int. J. Appl. Earth Observ. Geoinform.* 132:103996. doi: 10.1016/j.jag.2024.103996

Liu, B., Zhang, W., Chen, W., Huang, H., and Guo, S. (2020). Online computation offloading and traffic routing for UAV swarms in edge-cloud computing. *IEEE Trans. Vehic. Technol.* 69, 8777–8791. doi: 10.1109/TVT.2020.2994541

Liu, Z., Zhan, C., Cui, Y., Wu, C., and Hu, H. (2021). Robust edge computing in UAV systems via scalable computing and cooperative computing. *IEEE Wirel. Commun.* 28, 36–42. doi: 10.1109/MWC.121.2100041

Lu, K., Xie, J., Wan, Y., and Fu, S. (2019). Toward UAV-based airborne computing. *IEEE Wirel. Commun.* 26, 172–179. doi: 10.1109/MWC.2019.1900025

Luo, F., Jiang, C., Yu, S., Wang, J., Li, Y., and Ren, Y. (2017). Stability of cloudbased UAV systems supporting big data acquisition and processing. *IEEE Trans. Cloud Comput.* 7, 866–877. doi: 10.1109/TCC.2017.2696529

Messous, M.-A., Sedjelmaci, H., Houari, N., and Senouci, S.-M. (2017). "Computation offloading game for an UAV network in mobile edge computing," in 2017 IEEE International Conference on Communications (ICC) (IEEE), 1–6. doi: 10.1109/ICC.2017.7996483

Namani, S., and Gonen, B. (2020). "Smart agriculture based on IoT and cloud computing," in 2020 3rd International Conference on Information and Computer Technologies (ICICT), 553–556. doi: 10.1109/ICICT50521.2020.00094

Sharma, J., and Mehra, P. S. (2023). Secure communication in IoTbased UAV networks: a systematic survey. *Internet Things* 23:100883. doi: 10.1016/j.iot.2023.100883

Song, Z., Qin, X., Hao, Y., Hou, T., Wang, J., and Sun, X. (2022). A comprehensive survey on aerial mobile edge computing: challenges, state-of-the-art, and future directions. *Comput. Commun.* 191, 233–256. doi: 10.1016/j.comcom.2022.05.004

Sulaj, P., Haluska, R., Ovsenik, L., Marchevsky, S., and Kramar, V. (2018). "Examples of real-time UAV data processing with cloud computing," in *Proceedings of the 23rd Conference of Open Innovations Association FRUCT*, 543–548.

Valentino, R., Jung, W.-S., and Ko, Y.-B. (2018). "Opportunistic computational offloading system for clusters of drones," in 2018 20th International Conference on Advanced Communication Technology (ICACT) (IEEE), 303–306. doi: 10.23919/ICACT.2018.8323734

Wakode, M. S. (2021). "Role of AI and big data analytics in UAV-enabled IoT applications for smart cities," in Unmanned Aerial Vehicles for Internet of Things (IoT) Concepts, Techniques, and Applications, 193–205. doi: 10.1002/9781119769170.ch11

Wan, S., Lu, J., Fan, P., and Letaief, K. B. (2019). Toward big data processing in IoT: Path planning and resource management of UAV base stations in mobile-edge computing system. *IEEE Internet Things J.* 7, 5995–6009. doi: 10.1109/JIOT.2019.2954825

Yang, B., Cao, X., Yuen, C., and Qian, L. (2020). Offloading optimization in edge computing for deep-learning-enabled target tracking by internet

of UAVS. IEEE Internet of Things J. 8, 9878–9893. doi: 10.1109/JIOT.2020. 3016694

Ye, Z., Ji, P. N., and Wang, T. (2023). "Seamless service handover in UAV-based mobile edge computing," in *GLOBECOM 2023–2023 IEEE Global Communications Conference* (IEEE), 1113–1118. doi: 10.1109/GLOBECOM54140.2023.10437843

Zhang, J., Zhou, L., Tang, Q., Ngai, E. C.-H., Hu, X., Zhao, H., et al. (2018). Stochastic computation offloading and trajectory scheduling for UAV-assisted mobile edge computing. *IEEE Internet Things J.* 6, 3688–3699. doi: 10.1109/JIOT.2018. 2890133