BILINGUAL PUBLISHING GROUP

Journal of Environmental & Earth Sciences

https://journals.bilpubgroup.com/index.php/jees

REVIEW

Design and Implementation of a Lightweight AI-Driven Real-Time Preprocessing System for Remote Sensing Imagery

Hongyu Yu [1]***, Fangyuan Chen*** [2]***, Lianxiu Hao*** [3*]

[1] *The Second Geomatics Cartography Institute, Ministry of Natural Resource, Harbin 150081, China*

[2] *Survey and Monitoring Division, The Second Geomatics Cartography Institute, Ministry of Natural Resource, Harbin 150081, China*

[3] *Remote Sensing Information Division, The Second Geomatics Cartography Institute, Ministry of Natural Resource, Harbin 150081, China*

ABSTRACT

Thanks to the intensive development of remote sensing systems, Earth observation has never been accessed at such a scale, and thus, the urgency has been revealed to develop an effective and timely preprocessing response. Conventional preprocessing techniques, mostly operating on physical models, are frequently computationally expensive, fixed, and cannot be implemented in resource-constrained systems, such as satellites, unmanned aerial vehicles and edge devices. Recent technological progress in artificial intelligence, and especially lightweight deep learning models, presents a potential alternative since it provides adaptive, data-driven preprocessing at the cost of less computation. This review provides an in-depth discussion of the design and implementation of lightweight AI-based real-time preprocessing platforms for remote sensing imagery. It analyzes the basic properties of remote sensing data, important preprocessing activities, and how model-driven and data-driven approaches can be changed to each other. The paper goes on to discuss lightweight AI methods, such as model compression and efficient architecture, and system-level design considerations, such as hardware platforms, edge and cloud integration, and pipeline optimization. Practical challenges and opportunities are presented with references to the real-life situations in disaster monitoring, precision agriculture, and urban analytics. This work is a novel

*CORRESPONDING AUTHOR:

Lianxiu Hao, Remote Sensing Information Division, The Second Geomatics Cartography Institute, Ministry of Natural Resource, Harbin 150081, China; Email: 364742111@qq.com

ARTICLE INFO

Received: 9 April 2026 | Revised: 10 June 2026 | Accepted: 15 June 2026 | Published Online: 11 August 2026

DOI: https://doi.org/10.30564/jees.v8i8.13387

CITATION

Yu, H., Chen, F., Hao, L., 2026. Design and Implementation of a Lightweight AI-Driven Real-Time Preprocessing System for Remote Sensing Imagery. Journal of Environmental & Earth Sciences. 8(8): 100–126. DOI: https://doi.org/10.30564/jees.v8i8.13387

one due to its holistic approach, which focuses on the introduction of the innovation of algorithms and the deployment of systems. Lastly, future directions are mentioned to help formulate scalable, efficient, and robust preprocessing structures.
Keywords: Remote Sensing; Real-Time Preprocessing; Lightweight AI; Edge Computing; Image Enhancement

1. Introduction

Remote sensing images have become an inseparable part of data to comprehend and deal with the surface and atmosphere of the Earth [1]. As satellite constellations, unmanned aerial vehicles (UAVs), and high-resolution sensors keep improving in speed, the amount, speed, and types of remotely sensed data have increased exponentially [2]. Such data can be applied in vast fields of critical use, such as monitoring the environment, precision agriculture, urban planning, disaster response, and national security. Nonetheless, although the quality of sensors is increasingly available, raw images of remote sensing are usually marred by many degradations, including atmospheric interference, sensor noise, geometric distortion, and cloud contamination [3]. Therefore, preprocessing is a key functionality that can be used to make downstream analysis processes (classification, object detection, and change detection) accurate and reliable.

Conventionally, remote sensing imagery preprocessing has been based on physically meaningful models and rule-based algorithms. Radiometric calibration, atmospheric correction, geometric correction, and filtering are some of the techniques that have been widely studied and standardized [4]. Although these methods are good, they are usually computationally costly, sensor-based, and cannot be used to generalize over a wide range of environmental conditions. More to the point, the traditional pipelines of preprocessing usually assume offline processing in centralized computing systems and hence lead to severe latency and restrict their usability in real-time or near-real-time [5].

Artificial intelligence (AI) and especially deep learning have become a potent paradigm in recent years and are used to solve complex tasks in image processing [6]. Notable studies are showing that data-driven models have achieved significant performance in denoising, super-resolution, cloud removal, and image enhancement, and have been shown to be more accurate and robust than conventional approaches. These AI-based models are capable of learning implicit representations of noise distributions, atmospheric effects, and spatial-spectral correlations directly using data, eliminating the need to manually design models [7]. Consequently, the remote sensing workflow is starting to be enhanced with AI-based preprocessing.

Nevertheless, the majority of the available AI models in remote sensing preprocessing are computationally intensive and resource-consuming [8]. State-of-the-art deep neural networks typically contain millions of parameters and need high-performance GPUs to perform inference, which cannot be deployed on resource-limited systems like onboard satellite processors, edge devices, and UAV systems. Meanwhile, the need to have real-time processing features is increasing. The use of disaster tracking, border patrol, and precision farming, among others, needs real-time data interpretation so that the decision-making process can be prompt [9]. The fact that current AI models have computational requirements that this world cannot fairly meet is a major challenge.

Lightweight AI is an idea that has been becoming more and more popular in order to fill this gap. Lightweight models are also created to reduce the complexity of computers in terms of computational, memory, and energy costs without compromising on acceptable performance. Pruning of models, quantization, distillation of knowledge, and the creation of efficient network structures (e.g., MobileNet, ShuffleNet, and EfficientNet neuroarchitectures) have allowed the use of AI models on edge devices [10]. Used in remote sensing processing, lightweight AI provides the possibility of various complicated enhancement and correction operations at the point of origin, thus minimizing transmission costs and providing real-time analytics [11].

Regardless of these breakthroughs, the design and realization of a lightweight AI-based real-time preprocessing system for remote sensing imagery are under-researched in the literature. The literature tends to look at specific preprocessing functions (e.g., denoising or super-resolution) or specific model terms, but not the overall problem of the end-to-end system [12,13]. Moreover, there are no systematic reviews that combine algorithmic design with system-level aspects, including hardware limitations, pipeline architecture, optimization of pipelines, and real-time capabilities of performance. Such fragmentation prevents the implementation of operational, practical solutions that can be implemented in the remote sensing systems that will be operational.

The present review will serve to address this gap by offering a systematic and extensive discussion of the design and implementation of lightweight AI-driven preprocessing systems for real-time remote sensing applications. In contrast to other previous reviews, which focus mainly on the algorithmic performance, this work assumes a more holistic view that involves the model-level innovations and system-level integration. In particular, it analyzes the methods through which lightweight AI methods can be efficiently integrated with edge computing platforms, optimized software systems, and scalable data streams in an effort to attain real-time performance with limited resources [14].

This new article has a novelty in many aspects. One, it provides a single approach to comprehend remote sensing preprocessing both algorithmically and system design-wise to overcome the dichotomy between theory and implementation [15]. Second, it gives a comprehensive taxonomy of lightweight AI methods applied to the preprocessing of remote sensing, with their opportunities, limitations, and trade-offs [16]. Third, it highlights real-time processing needs, such as latency, throughput, and energy efficiency, which have been generally neglected in the literature on the subject [17]. Fourth, it explains practical implementation plans on various hardware platforms such as embedded GPUs, FPGAs, and onboard satellite processors with a discussion on the practical challenges of implementation in the real-world situation [18]. Lastly, it indicates the developmental trends, including TinyML, self-supervised learning, and federated learning, and discusses how the three have the potential to enrich lightweight, real-time preprocessing systems [19].

This review intends to pinpoint all the new developments in recent AI, remote, and edge computing to act as a solid reference point for researchers and practitioners in creating the next generation of preprocessing systems. It not only describes the state of the art presently but also gives the open issues of challenges and a research avenue for the future to enable smarter, more efficient, and scalable remote sensing processes. **Figure 1** shows the overall conceptual framework of the paradigm of lightweight AI-driven real-time preprocessing reviewed in the current article.

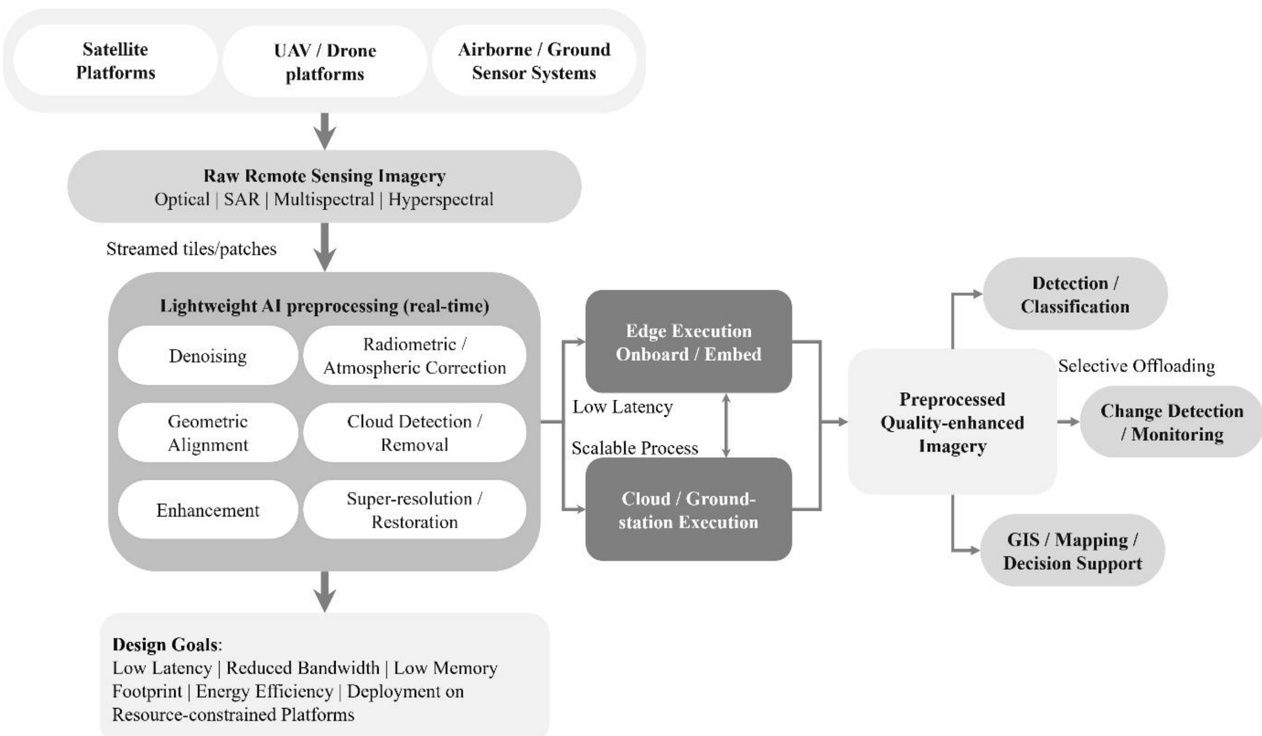

Figure 1. Overall framework of a lightweight AI-driven real-time preprocessing system for remote sensing imagery, showing the relationship among data acquisition platforms, preprocessing modules, edge/cloud execution, and downstream applications.

2. Fundamentals of Remote Sensing Preprocessing

2.1. Overview of Remote Sensing Data Characteristics

Remote sensing imagery is also intrinsically complex because of the variety of sensing modalities, acquisition geometries, and conditions of the environment in which the data is being sampled. Remote sensing data are often characterized by high spectral dimensionality, large spatial extent, and change over time, unlike traditional natural images [20]. These are the features that present distinct challenges to preprocessing since the data is not only large-scale, but also heterogeneous spatially and spectrally.

Optical images, such as multispectral and hyperspectral data, are images that reflect the solar radiation of various wavelengths, which allows the study of land cover and properties of materials [21]. In particular, hyperspectral imagery has hundreds of spectral bands, one right after another, which contain a lot of information but also expose it to noise and redundancy. Synthetic Aperture Radar (SAR) imagery, on the other hand, works in the microwave band and can cut through clouds and record images in all weather conditions [22]. Nevertheless, the use of SAR images is associated with speckle noise and the complicated backscatter processes, making preprocessing tasks more difficult. Third, there is also the introduction of thermal infrared imagery (and LiDAR data), which has a different noise model and needs to be calibrated differently.

The heterogeneity of sensor types and acquisition conditions makes the data inconsistent in radiometric, geometric, and spectral properties. Atmospheric scattering, changes in illumination, sensor drift, and platform motion are some of the factors that cause distortions and should be fixed before analysis [23]. Furthermore, the proliferation of data due to the growing use of small-scale satellites and UAV-based sensors with different resolutions and quality levels has only highlighted the importance of bringing to the fore strong and flexible preprocessing methods. These basic properties are key to the development of preprocessing systems that can be generalized in terms of their datasets and application environments. **Table 1** summarizes the major remote sensing modalities, data characteristics, and the preprocessing challenges associated with these modalities.

Table 1. Major remote sensing data types, representative distortions, preprocessing requirements, and associated challenges.

Data Type	Characteristics	Common Distortions/Noise	Preprocessing Requirements	Challenges
Optical (RGB/Multispectral)	High spatial resolution, limited spectral bands	Atmospheric scattering, illumination variation	Radiometric & atmospheric correction	Sensitive to clouds and lighting
Hyperspectral	High spectral resolution (hundreds of bands)	Sensor noise, redundancy, spectral distortion	Denoising, dimensionality reduction	High computational complexity
SAR	All-weather, day-night capability	Speckle noise (multiplicative)	Speckle filtering, geometric correction	Complex noise distribution
Thermal Infrared	Temperature-sensitive measurements	Sensor drift, calibration errors	Radiometric calibration	Low spatial resolution
LiDAR	3D point cloud data	Missing points, measurement noise	Registration, interpolation	Data sparsity and irregularity

2.2. Core Preprocessing Tasks and Their Physical Foundations

The classical approach to preprocessing in remote sensing has been based on physical models that characterize what happens when electromagnetic radiation interacts with the surface and atmosphere of the Earth [24]. One of the major steps is radiometric correction, which attempts to convert raw digital numbers into physically meaningful values, which might be reflectance or radiance [25]. This process considers sensor calibration, the state of illumination, and sensor degradation with time. Although radiometric correction is firmly established, its quality requires the calibration parameters of sensors and the assumption concerning the atmospheric conditions.

Further atmospheric correction is used to correct the data to remove scattering and absorption by atmospheric constituents (aerosols, water vapor, gases, etc.) [26]. Radiative transfer equations are frequently employed as physical models, but they need a suitable estimation of atmospheric parameters, which are not always readily available, or sometimes hard to measure in real-time. Consequently, atmospheric correction continues to be a major source of ambiguity with regard to preprocessing pipelines.

Geometric correction and image registration refer to the spatial distortions that are due to sensor motion, Earth curvature, and terrain variation [27]. These operations include matching images to a shared coordinate system, usually with ground control points and digital elevation models. Although the geometric correction is essential to multi-temporal and multi-sensor analysis, it is computationally expensive and vulnerable to control point selection errors. Another crucial preprocessing activity, especially that of SAR and hyperspectral data, is noise reduction. More classic methods of filtering, including Gaussian smoothing and median filtering, are common but give up on fine-scale detail and spatial resolution. More complex methods, such as adaptive and non-local filters, also aim at maintaining structural details whilst removing noise; they have complex parameter tuning.

Spotting and elimination of clouds is also an especially difficult issue in optical remote sensing. The clouds cover the surface of the Earth and create a big uncertainty in the analysis downstream. The classical techniques are based on thresholding and spectral indices, which are usually not reliable under different light conditions and in the atmosphere [28]. On the same note, super-resolution and image enhancement methods seek to enhance spatial and spectral quality, although they are normally limited by the physical and computing resources of sensors.

Although these classical preprocessing tasks cannot be ignored, they are commonly considered so that they are entirely autonomous and result in fragmented pipelines that are challenging to optimize as a whole. Further, they are constrained by explicit physical models, which restrict their flexibility to a wide range of changing environments.

2.3. Transition from Model-Driven to Data-Driven Approaches

The drawbacks of conventional model-driven preprocessing have promoted the use of data-driven methods, especially machine learning and deep learning methods [29]. In contrast to the physically-grounded methods, data-driven techniques are trained to learn mappings directly on observed data, and can therefore learn complex and nonlinear relationships that are hard to express in an explicit model.

Denoising, cloud removal, atmospheric correction, and super-resolution are some of the tasks in remote sensing preprocessing that have been successfully addressed using deep learning models [30]. CNNs can take advantage of the spatial context and hierarchical representations of features; they are very appropriate in image restoration operations [31]. Most recently, transformer-based designs have appeared to represent long-range dependencies and spectral correlations, especially in hyperspectral images.

The major benefit of data-driven methods is that they can be generalized to new sensors and new environmental conditions, in case there is enough training data [32]. Or, as an example, learning-based atmospheric correction models are able to implicitly capture the effects of complex scattering without necessarily estimating the parameters. Equally, deep denoising models are able to change to other noise distributions, such as the multiplicative noise of an SAR imagery.

Nevertheless, there are obstacles to the change towards data-driven preprocessing. Deep learning models can take substantial quantities of labeled data, which can be difficult and costly to find in remote sensing [33]. Moreover, these models may be domain sensitive, i.e., alterations in sensor properties or geographic ranges, which will cause poor performance in actual applications [34]. Lack of interpretability is another vital concern because data-driven models are usually black boxes; it is not easy to determine their reliability in safety-critical applications.

Nevertheless, methods of data-driven integration into preprocessing pipelines are a change of paradigm, as they make solutions more flexible and adaptive [35]. The most important question is how to reconcile the advantages of model-driven and data-driven methods, which may be achieved by hybrid frameworks that integrate physical knowledge with learnt representations. These structural differences between traditional sequential preprocessing and AI-based integrated preprocessing are compared in **Figure 2**.

2.4. Evaluation Metrics and Performance Trade-Offs

When analyzing preprocessing techniques in the domain of remote sensing, one cannot help but pay close attention to the quality of data as well as the efficiency of computation. Fidelity of reconstructed images is often determined by traditional image quality metrics, including, but not limited to, peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM)[36]. In the case of hyperspectral data, spectral angle mapper (SAM) and similar measurements give a measure of spectral distortion, which is extremely important in matter identification applications.

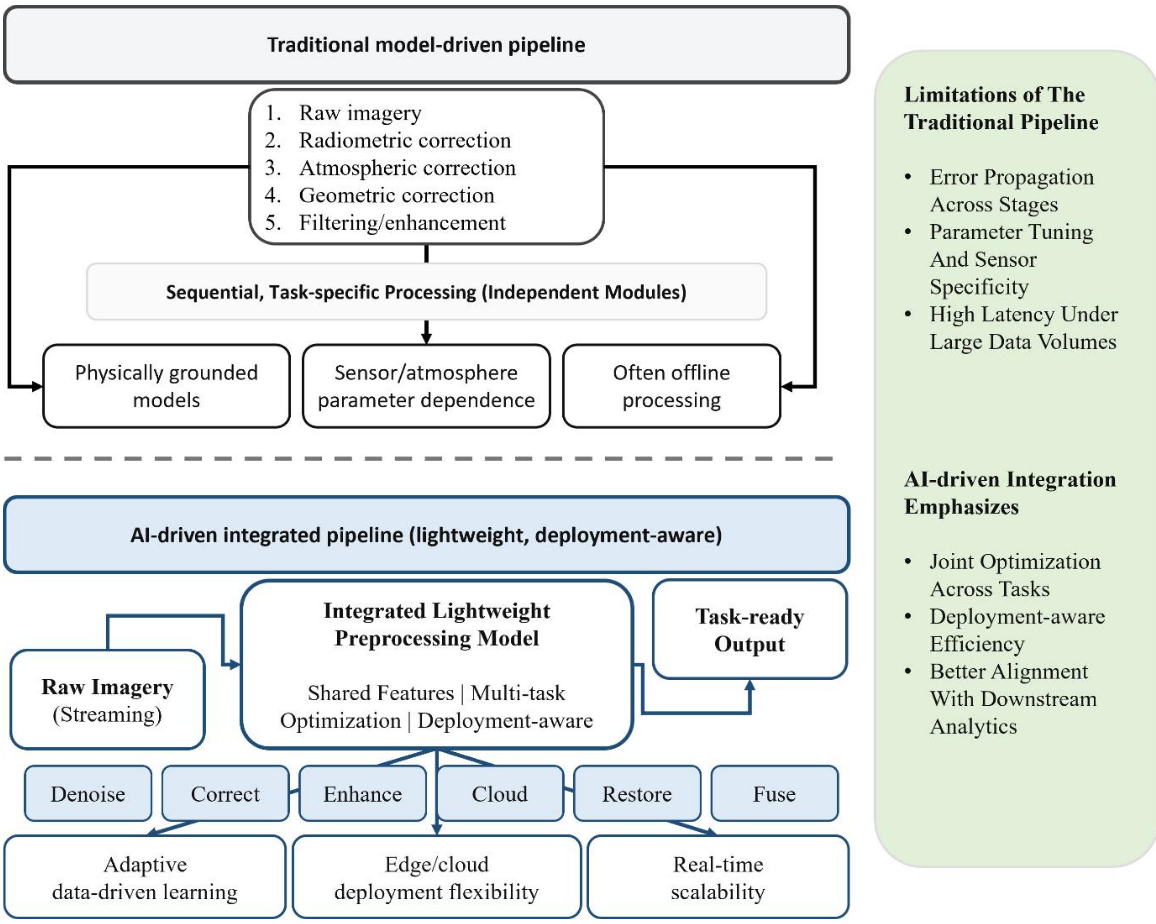

Figure 2. Schematic comparison between traditional model-driven preprocessing pipelines and AI-driven preprocessing frameworks for remote sensing imagery.

Although these measures are effective, they do not provide a complete picture of the effect of preprocessing on downstream activities. Considering the example, a high PSNR image can still have ineffective classification performance when the significant features are distorted[37]. In turn, task-based evaluation, when the performance of the subsequent analysis tasks is measured, is becoming more defined as a more important measure.

Computational efficiency has also become a primary issue to be considered in addition to accuracy, especially in real-time and resource-constrained contexts[18]. Inferences, latency, throughput, memory usage, and energy consumption metrics are important in determining the viability of preprocessing systems. These measures can be said to be dependent, and they give rise to complicated trade-offs. An example is that to achieve higher accuracy in a model, it is common to have a larger and more complex architecture, which increases the cost and latency of computation.

Scalability is another factor to consider. Remote sensing data can be extremely large, and the preprocessing systems should be able to process high volumes of data without major performance deterioration[38]. This is essential, especially when dealing with satellite constellations that produce streams of information on a continuous basis. Preprocessing method evaluation is therefore a multi-dimensional optimization problem that is composed of accuracy, efficiency, robustness, and scalability. Current research tends to concentrate on part of these measures, resulting in partial evaluations. The next generation of preprocessing systems should be developed with the help of a more complex evaluation scheme.

2.5. Limitations of Conventional Preprocessing Pipelines

Nevertheless, traditional preprocessing pipelines have several inherent weaknesses that prevent them from being generalized to contemporary remote sensing systems. One of the major problems is that they are not flexible [39]. The traditional techniques tend to be specific to a particular sensor or environmental condition and thus are hard to generalize to a wide range of datasets. The problem of this limitation is especially acute in monitoring globally, as information is gathered on various platforms with different properties.

The other constraint is the sequential and modular construction of a conventional pipeline [40]. The individual stages of preprocessing are generally done individually, without looking at any relationship between the stages. This dissection can result in less-than-optimal performance, where the errors added at each step can be passed on and increase in other steps. In addition, the end-to-end optimization cannot be achieved, and hence it is hard to generate a consistent performance across the tasks.

Computational complexity is an important consideration as well [41]. Most of the existing traditional preprocessing algorithms require either iterative optimization or complicated physical models, which are computationally prohibitive and cannot be applied in real-time. Consequently, preprocessing can be done offline, creating latencies that cannot be tolerated in time-critical systems like disaster response. Moreover, standard pipelines do not target the current hardware architecture. They do not frequently take advantage of parallel processing of GPUs and other specialized accelerators, so they do not optimally use resources. This is a weakness that is magnified by the fact that remote sensing systems are shifting towards edge and onboard processing.

Lastly, the increasing need for real-time analytics reveals the weakness of the current preprocessing methods [42]. The delay incurred with the traditional approaches can be significant, especially when information needs to be manipulated and acted upon within a number of seconds or minutes. It is this issue that new paradigms are necessary that combine algorithmic efficiency and system-level optimization.

2.6. Toward Integrated and Real-Time Preprocessing Paradigms

There has been an increasing push towards integrated and real-time systems that bring together developments in AI, edge computing, and system design because of the drawbacks of the traditional preprocessing method [14]. Instead of considering preprocessing a set of discrete operations, new models strive to combine several functions into one, end-to-end streamlined pipeline. By doing this, it is possible to jointly optimize various preprocessing goals, which can lead to the enhancement of accuracy and efficiency. Real-time preprocessing needs a change in the design philosophy, where low-latency inference, efficient resource use, and changes in accordance with the dynamic conditions are parameters of focus. In this regard, lightweight AI models are highly important because they allow complex processing operations to be carried out on resource-constrained devices [43]. However, real-time performance is not only a design concern but also requires data flow optimization, memory management, and hardware optimization.

The development of edge computing has seen it as a major facilitator of the real-time preprocessing of data that can be processed nearer to the source [44]. This will minimize the amount of data to be sent to the centralized servers, resulting in low latency and bandwidth requirements. When it comes to remote sensing, edge computing may be deployed on the satellites, UAVs, or ground stations with their respective restrictions and opportunities [45]. Although this has been advanced, there are many challenges. Assuring strong resistance and generalization in dynamic environments is a significant issue, especially in AI-based approaches. Also, the processing modules and heterogeneous data sources require standard interfaces and interoperability frameworks. It is the existence of such challenges that will be important in the development of viable, scalable preprocessing systems.

To conclude, the traditional model-driven remote sensing preprocessing is transforming into significant, integrated, AI-driven models of remote sensing and preprocessing that value real-time performance and system-level efficiency [46]. Such a shift preconditions the creation of lightweight preprocessing systems that will be discussed in the following sections.

2.7. Systematic Taxonomy of Remote Sensing Preprocessing Tasks

We suggest a three-level hierarchical classification of all fundamental preprocessing tasks based on their physical purpose, mathematical formulation, and impact on downstream tasks. We suggest a three-level hierarchical taxonomy of basic preprocessing tasks, organized by physical purpose, mathematical formulation, and impact on subsequent tasks. This taxonomy allows for a common language across various task types and provides a common set of criteria for evaluating tasks.

The first level is radiometric correction, which attempts to transform raw sensor digital numbers into physically meaningful radiometric quantities, such as reflectance or radiance. These can include radiometric calibration to correct for sensor-specific gain and offset variations, atmospheric correction to remove scattering and absorption effects from atmospheric constituents such as water vapor and aerosols, and illumination normalization to account for topographic and solar angle variations. These tasks have in common that they are represented as linear or nonlinear transformations of pixel values, and that the degradation factors are typically sensor- and environment-dependent [47,48].

The second level is geometric correction for spatial distortion caused by sensor motion, Earth curvature, terrain relief, and platform instability. It includes geometric rectification, in which image coordinates are transformed to a standard map projection; orthorectification, which corrects for relief-induced displacements using a digital elevation model; and image registration, which aligns multiple images of the same scene taken at different times or by different sensors. These are essentially spatial transformation problems, and the need for accurate ground control points or tie points is inherent. However, more recent work in AI has considered learning-based registration without explicit feature matching.

The third level is for enhancement and restoration applications that improve visual quality or recover lost information but do not change the geometric or radiometric characteristics of the data. This category encompasses denoising of optical images (where the dominant noise source is additive Gaussian or Poisson noise), speckle filtering of synthetic aperture radar data (where the dominant noise source is multiplicative and depends on the signal), cloud detection and removal (where the noise consists of masking of obscured pixels and reconstruction of the underlying surface image), super-resolution (where the signal is reconstructed from a set of lower-resolution inputs), and deblurring (which involves removal of motion or optical blurring). These tasks involve ill-posed inverse problems with a crucial role for domain knowledge and learned priors.

Different degradation models exist for each task in this taxonomy and directly impact the design of lightweight AI solutions. Typically, the transformations used to make radiometric corrections can be represented as linear, either globally or locally, and can be handled easily by simple convolutional or even fully connected networks. In the context of geometric corrections, efficient alternatives to traditional interpolation methods, such as spatial transformer networks and differentiable warping modules, have been proposed. Residual learning, attention, and generative models have been transferred to efficient architectures like MobileNet and EfficientNet for enhancement tasks.

Importantly, the taxonomy is supported by a consistent evaluation framework used at each of the three levels. Signal fidelity measures such as peak signal-to-noise ratio, structural similarity index, and the spectral angle mapper assess reconstruction quality. Computational metrics, including inference latency, floating-point operations, memory footprint, and energy consumption, assess real-time suitability. Task-specific metrics, such as downstream classification accuracy or object detection precision, assess the usefulness of the outputs from the preprocessing steps. With the aid of this taxonomy and its unified criteria, researchers can systematically evaluate methods across various deployment scenarios and preprocessing tasks [49,50].

3. Lightweight AI Techniques for Real-Time Preprocessing

3.1. Conceptual Foundations of Lightweight AI in Remote Sensing

Lightweight artificial intelligence has become a trend of innovation in light of increasing calls to apply deep learning models in resource-constrained settings [51]. Lightweight, when applied to remote sensing preprocessing, does not just confine itself to the notion of reduced model size, but represents a more general set of design requirements, such as low computational complexity, small memory footprint, low energy use, and low inference latency [52]. These limitations are especially important in edge platforms, including satellites, UAVs, and embedded ground stations, in which the computational resources are restricted, and power efficiency is the key factor.

Lightweight AI focuses on the balance between performance and resource consumption, unlike the traditional models of deep learning, which are based on accuracy without respect to efficiency [53]. This tradeoff is particularly significant when doing preprocessing jobs where the aim of the work is not always to generate semantically rich products, but to improve data quality in a manner that is useful at the end-user level. This means that tolerance to minor accuracy trade-offs can be greater, as long as the preprocessing can be executed on scale, and in real time.

Systems Lightweight AI allows transitioning to decentralized, edge-based systems, compared to the existing, centralized cloud-based systems [54]. This change minimizes the need for transmission of data and enables preprocessing to be closer to the source of data, and hence reduces the latency. Nonetheless, this demands a balance between co-designing algorithms and hardware, as well as a trade-off between the complexity of models and their accuracy and their constraints on deployment. Therefore, lightweight AI cannot be considered merely as a collection of methods, but rather as a design philosophy that transcends the entire preprocessing pipeline.

3.2. Model Compression and Optimization Techniques

One of the basic practices of attaining lightweight AI is model compression, where the goals are to minimize the size and the computational demands of deep neural networks without necessarily compromising performance [55]. Curing, quantization, and knowledge distillation are some of the most popular methods with their unique strengths and drawbacks. Pruning is the process of eliminating unnecessary or unnecessary parameters in a trained network. Pruning Structured pruning is specifically applicable to real-time preprocessing, since it results in more efficient implementations on hardware accelerators. Aggressive pruning may, however, lead to loss of important information, particularly in an activity like hyperspectral denoising that requires fine spectral details to be maintained. In addition, repetition of retraining is frequent due to pruning, and this also complicates the model development process.

Quantization is a type of model parameter and activation quantization that decreases the precision of model parameters and activations, usually decreasing as the number of bits, such as an 8-bit integer [56]. Such a reduction will greatly reduce memory usage and infer more quickly on hardware that implements low-precision arithmetic. Quantization can be very useful in remote sensing preprocessing, especially in image processing and cloud-finding, where the dynamic range of features can be represented with reduced precision. However, the quantization can cause numerical instability, especially in high dynamic range models, like the SAR images.

Knowledge distillation is the transfer of knowledge between a large, high-performing teacher model and a smaller student model [57]. It is especially attractive in the context of preprocessing operations, and in this setting, small-scale models can be used to estimate the performance of more complicated architectures. Nevertheless, the success of distillation is dependent on the teacher model selection and the correspondence between training and deployment factors. This alignment is not trivial in remote sensing when the data distributions may have radically different shapes.

These techniques have shown great potential, but their use in the pre-treatment of remote sensing must be adapted with great caution. **Table 2** gives a comparative overview of the key strategies to optimize lightweight, alongside the advantages, drawbacks, and applicability to real-time implementation of each. Due to the peculiarities of remote sensing data, e.g., the high spectral dimensionality and sensor-specific noise patterns, task-based compression methods are needed [58]. An unsophisticated use of generic compression techniques can cause unacceptable loss in data quality, which is why domain-specific optimization is necessary.

Table 2. Comparison of representative lightweight AI techniques for remote sensing preprocessing, including core principles, advantages, limitations, and real-time suitability.

Technique	Principle	Advantages	Limitations	Suitability for Real-Time
Pruning	Remove redundant parameters	Reduces model size, faster inference	Accuracy degradation if over-pruned	High
Quantization	Lower precision representation	Reduced memory and energy usage	Numerical instability	Very High
Knowledge Distillation	Transfer knowledge to smaller model	Maintains performance with smaller model	Requires teacher model	High
Efficient Architectures (e.g., MobileNet)	Lightweight design by default	Low FLOPs and latency	Limited representation capacity	Very High
Neural Architecture Search (NAS)	Automated model design	Optimized for hardware/task	High search cost	Medium

3.3. Efficient Neural Network Architectures

In addition to post hoc compression, neural network architecture design is also of great importance in lightweight AI. MobileNet, ShuffleNet, and EfficientNet are architectures that have made efficiency a major concern, including depthwise separable convolutions, group convolutions, and compound scaling to minimize computation costs [59].

These architectures also provide a prospective base to construct lightweight models in the remote sensing preprocessing setting. Depthwise separable convolutions are much lower in terms of parameters and operations than regular convolutions, and are applied in tasks like denoising and super-resolution [60]. Yet they are susceptible to the capability to capture spatial and spectral correlations, which can be more complicated in remote sensing data as compared to natural images. Most recent work has also examined the use of a lightweight transformer architecture that tries to model long-range dependencies at low computational cost. Models that integrate convolutional and transformer architectures are known to be promising in applications to the fields of hyperspectral reconstruction and cloud removal. These frameworks combine the advantages of the two paradigms, where convolutional layers are employed to extract local features, and attention is employed to model global features. However, the computational cost of attention mechanisms is an issue, especially with high-resolution images.

The second factor that should be considered is the configuration of architectures for particular preprocessing tasks. As an example, lightweight variants of U-Net have been extensively applied to tasks of restoring images, as they have an encoder-decoder architecture and skip connections [61]. These models can find a balance that exists between efficiency and performance by minimizing the channels and layers. At the same time, their capacity to rebuild finer details, which are important to the downstream analysis, can be restricted due to excessive simplification.

In general, efficient architecture design is a demanding task that needs a delicate awareness of the task demands and the nature of remote sensing data. Although the general-purpose lightweight models will make a handy first step, the domain-specific adjustments are frequently needed to achieve the best results.

3.4. Task-Specific Lightweight Models for Preprocessing

Various preprocessing tasks have different requirements on the design of the model, and task-specific variations of lightweight AI techniques should be used [62]. In denoising, the models should be effective in reducing noise without losing fine spatial as well as spectral detail. The residual learning and attention mechanisms are frequently used to make lightweight denoising networks efficient at representing features and are typically used to improve overall efficiency. The difficulty is, however, to deal with the various types of noises, like the Gaussian noise in optical imagery and the multiplicative speckle noise in SAR data.

Another group of difficulties is cloud detection and removal, where the complex and variable patterns of clouds are involved [63]. The models used to perform these tasks are often lightweight models that are based on semantic segmentation, but need to be optimized to achieve very strong latency requirements. Less sophisticated encoder-decoder structures and low-resolution processing are often employed, although they can adversely affect accuracy, especially with bright or semi-transparent clouds.

Tasks that involve super-resolution and image enhancement involve models that rebuild high-frequency features using low-resolution images. Lightweight super-resolution networks can employ effective up-sampling algorithms and smaller feature maps in order to reduce the computation cost. But these models should take care of the trade-off between reconstruction of details and the generation of artifacts that may hurt downstream work. The data of hyperspectral preprocessing have a high dimensionality that creates more complexity. Lightweight models need to be able to manipulate both spatial and spectral information and may need special architecture, like 3D convolutions or spectral attention [64]. It is still a challenge to reduce the computational burden of these operations and still perform them at the same level of performance.

One important lesson observed in all these tasks is that it is impossible to decouple lightweight design and task requirements. An optimized model that is good at one preprocessing task might not translate well to a different preprocessing task; thus, the importance of modular and adjustable frameworks. In addition, the ability to combine several preprocessing operations into a single lightweight model is also a field of ongoing study, and much can be done to increase its efficiency and consistency [65].

3.5. Trade-Offs between Accuracy, Efficiency, and Robustness

The creation of lightweight AI models is by definition a tradeoff between conflicting goals [66]. Although simplification of models increases efficiency, they usually reduce accuracy as well. This trade-off should be handled carefully in remote sensing preprocessing because even minor deteriorations in data quality may be transferred to its downstream work and impact the performance of the entire system.

Compression or architectural simplification that is used to gain efficiency can also affect robustness [67]. Lightweight models can also be less adaptable to changes in data distribution, e.g., sensor characteristics or environmental conditions. This is an especially troublesome constraint in remote sensing, where information is observed under a variety of ad hoc and changing conditions. To achieve robustness in lightweight models, it is important to implement methods that include data augmentation, domain adaptation, and uncertainty estimation.

The other significant trade-off is the one between generalization and specialization [68]. The highly specialized models can perform very well on particular tasks or databases, yet they cannot be used in other situations. On the other hand, more general models can trade task-specific performance for generalizability. Regarding real-time preprocessing, these two strategies would depend upon the target deployment environment and the needs of the application.

Another dimension to this trade-off space is added through energy efficiency [69]. Power consumption is a major constraint on edge devices, having an impact on model and hardware design, as well as system architecture. Dynamic model scaling and adaptive inference are techniques that can be used to optimize both the performance and the energy consumption, but introduce additional complexity to the system.

Finally, to design lightweight AI models for preprocessing remote sensing, a multifaceted approach should be used, taking into account multiple objectives at the same time [70]. Instead of maximizing one metric, researchers have to find solutions to a multidimensional design space, where the solutions can provide an acceptable tradeoff between accuracy, efficiency, robustness, and scalability.

3.6. Benchmarking and Standardization Challenges

Although a major advancement has been made in lightweight AI, benchmarking and standardization are some of the key challenges of the discipline. In the literature, various datasets, performance metrics and controlled conditions are usually applied, so it is hard to compare the findings and determine the actual efficiency of the methods suggested. This is especially an issue with remote sensing, where data range and use needs are diverse. Lightweight models should be benchmarked based on not only on the measure of accuracy, but also on the measure of computational efficiency in realistic deployment settings [71]. Latency, throughput, and energy consumption metrics have to be evaluated on target hardware platforms, as opposed to using theoretical estimates only. But these are usually left out because of the complexity of experimental arrangement and the non-existence of standard protocols.

The other problem is the reproducibility of the results [72]. Most of the studies lack adequate information regarding model architecture, training processes, or deployment infrastructure, which makes it difficult to duplicate and apply the work of other researchers. This is a further problem in the context of an AI with a lightweight, where even minor implementation details can make a big difference in performance. Additionally, benchmark data corresponding to real-life scenarios, e.g., sensor type differences, noise, and environmental changes, are required. Recent datasets tend to be narrower, concentrating on a particular task or location, which does not reflect the complexity of an operational situation.

It is necessary to solve these issues through the concerted efforts of the research community to create standard benchmarks, protocols to evaluate them, and open-source implementations. This would allow closer comparisons to be made and enable the creation of efficient, lightweight preprocessing solutions.

3.7. Emerging Directions in Lightweight AI for Remote Sensing

Lightweight AI is a developing field, and a number of directions are currently emerging that have the potential to support remote sensing preprocessing. Integration of neural architecture search (NAS) is one of those directions, and it automates the creation of efficient models depending on the tasks and hardware limitations [73]. NAS can potentially find new architectures that are better than those of the manually designed models, but its cost is a limitation. The other potential field of use is in the creation of TinyML, which aims at executing machine learning models on low-power systems [74]. Even though TinyML is still in its infancy, it may allow even the smallest of systems, including nanosatellites and sensor nodes, to perform preprocessing. In order to meaningfully perform on such constraints, however, this demands considerable innovation both in algorithms and hardware.

Unsupervised and self-supervised methods of learning are also becoming popular when it comes to minimizing the use of labeled data [75]. These techniques have the ability to utilize large amounts of unlabeled remote sensing data to train strong feature representations that can be reused for preprocessing tasks. The method is specifically applicable to remote sensing, where the labeled data may be limited. Lastly, the intersection of lightweight AI and edge computing and distributed learning systems, including federated learning, creates new prospects for scalable and privacy-aware preprocessing systems. These methods have the potential to enhance the performance of the models and address constraints of data in the face of distributed learning across the nodes.

Conclusively, lightweight AI methods play the central role of facilitating real-time preprocessing in remote sensing. Although a lot of progress has already been achieved, more research is still required to overcome all the issues of efficiency, strength, and standardization to enable a practical and scalable implementation.

4. System Design and Implementation for Real-Time Processing

4.1. End-to-End Architecture of Real-Time Preprocessing Systems

The actual implementation of a lightweight AI-based preprocessing system for remote sensing imagery demands the careful coordination of an end-to-end architecture for data acquisition, processing, and transmission under a tight latency requirement [76]. Real-time systems are continuously required to run on streaming data, unlike traditional offline pipelines, which often have dynamic environments and operational conditions. This requires a transition from an inert, batch-based design to a pipe-based design, capable of processing data in small segments effectively.

An example of a real-time preprocessing system starts at the data acquisition point, where sensor platforms on other satellites, UAVs, or on the ground acquire raw imagery [77]. This information is further subjected to a preprocessing module where tasks like denoising, correction, and enhancement are done before it is sent to the downstream analytics or storage systems. The main issue is to reduce the interval between acquisition and output, and still have a reasonable data quality for further work. In order to do this, the new architecture of systems is embracing the streaming and parallel processors paradigm. Data are broken into smaller units, or tiles, or patches that can be processed individually and simultaneously. Not only does this strategy minimize memory needs, but it also provides an opportunity to enjoy hardware resources. It, however, comes with a difficulty regarding boundary artifacts and the consistency between tiles, especially when working on tasks that need global context, such as atmospheric correction or cloud removal.

The other important issue in system design is the preprocessing and downstream tasks integration [78]. Preprocessing is no longer a separate step but a component of an end-to-end pipeline in many applications, which also includes detection, classification, or change analysis. These stages can be optimized together to achieve greater total performance and result in a more complex system. It is thus necessary to design modular and, at the same time, interoperable components so as to ensure a degree of flexibility and scalability.

4.2. Edge Computing versus Cloud-Centric Architectures

One of the main design choices that should be made in real-time preprocessing systems is the decision between edge computing and cloud-based processing [79]. Conventionally, remote sensing data have been sent to centralized servers to be processed, which has taken advantage of the computing capabilities and storage capacity of the cloud infrastructures. Although this method is very accurate and flexible, it has great latency and bandwidth needs, which are becoming unsustainable with the size of the present-day remote sensing data.

Edge computing can help to overcome these shortcomings by preprocessing the data nearer to the data source (on a satellite, UAVs, or ground station) [80]. This paradigm minimizes the transmission data required to be sent, and response time is quicker, and communication resources are used more efficiently. As an example, onboard preprocessing can be used to filter irrelevant or low-quality data before transmission, thus reducing the bandwidth used by a great margin.

But edge computing puts stringent requirements on processing units, power, and thermal control [81]. Lightweight artificial intelligence models will be required here, but their implementation should be optimized to correspond with the capabilities of the target hardware. Moreover, edge devices may be used in harsh and changeable conditions that may influence the reliability and performance of the system.

On the other hand, cloud-based architectures may still be useful in those activities that demand high computational capabilities or access to big datasets [82]. The combination of these two types of processing through hybrid techniques is thus on the rise. In these systems, the first stage of preprocessing is done at the edge to minimize the volume of data and latency, and more complex or computationally intensive tasks are deployed to the cloud. This autonomy of labor demands efficient data synchronization and communication schemes, along with mechanisms for the dynamic assignment of tasks under the circumstances of the system.

The decision on edge, cloud, or hybrid architecture is therefore not binary but rather is based on the needs of the application, available infrastructure, and operational limitations. One of the most important research problems is the development of adaptive systems that can easily switch between these modes in order to maximize their performance.

4.3. Hardware Platforms and Deployment Constraints

The implementation of real-time preprocessing systems is essentially limited by the performance of the supporting hardware. Hardware platforms in remote sensing use can be high-performance servers and GPUs, embedded systems, and accelerators with unique attributes and constraints [18].

Embedded GPUs include those used in edge devices and UAVs, and provide a trade-off between computing power and energy efficiency [83]. They are friendly to parallel processing and well-suited to executing deep learning models, especially low-latency optimized models. Their memory and power efficiency are, however, lower in comparison to data centre GPUs, requiring very careful model design and optimization. Another beneficial platform for real-time preprocessing is field-programmable gate arrays (FPGAs). This is due to the reconfigurable architecture that enables the implementation of specific algorithms in new ways and thus makes them highly efficient and low-latency. FPGAs are especially useful in fixed-function tasks (e.g., filtering and geometric transformations). FPGAs are, however, difficult to program and specialized knowledge is needed, which may restrict their availability.

The most efficient but least flexible is the application-specific integrated circuits (ASICs). ASICs can be engineered to handle particular workloads in order to perform better and consume less energy than any other device [84]. ASICs are also starting to be considered in satellite payloads as onboard processors in the field of remote sensing. They are, however, inadequate in terms of flexibility and, as such, not so suited to quickly changing AI models and algorithms. Hardware reliability and robustness are the other significant factors that should be considered in the operational environment. The extreme conditions that satellites and UAVs can endure are temperature variations, radiation exposure, and mechanical strains. All these can influence the performance of the hardware as well as the accuracy of the models, and this requires a strong system design and verification.

Finally, the hardware platform selection has a two-fold effect on the algorithm and software design as well as the viability of real-time preprocessing. Comparison of key hardware platforms deployed in real-time remote sensing preprocessing; **Table 3** lists the main comparisons of the hardware. To achieve optimal performance, co-design approaches, i.e., the simultaneous optimization of hardware and software, are therefore necessary.

Table 3. Comparison of major hardware platforms for real-time remote sensing preprocessing in terms of computational capability, energy efficiency, flexibility, use cases, and constraints.

Platform	Computational Capability	Energy Efficiency	Flexibility	Typical Use Case	Limitations
CPU	Moderate	Moderate	High	Ground stations	Limited parallelism
GPU (Embedded)	High	Moderate	High	UAVs, edge devices	Power consumption
FPGA	High (customizable)	High	Medium	Real-time pipelines	Complex programming
ASIC	Very High	Very High	Low	Satellite onboard processing	Lack of flexibility
Edge AI Chips (e.g., NPUs)	High	Very High	Medium	Mobile/edge AI	Limited compatibility

4.4. Software Frameworks and Model Deployment Pipelines

The practical implementation of lightweight AI models used to perform real-time preprocessing is strongly dependent on software frameworks to optimize, execute, and integrate the model[85]. The tools to convert and optimize models and run them on different hardware platforms include frameworks such as TensorRT, ONNX Runtime, TensorFlow Lite, and PyTorch Mobile. These models facilitate various optimization methods such as graph simplification, operator fusion, and hardware-based acceleration. They allow executing high-level model representations more quickly and consuming fewer resources by converting them into efficient execution graphs. The success of such optimizations, however, is determined by the compatibility of the model architecture with the target hardware.

Model deployment pipelines usually have different stages, such as model training, model compression, model conversion, and validation[86]. The stages present possible causes of error and performance degradation. As an example, the transformation of a model between different frameworks can result in differences in numerical accuracy or an inability to perform certain operations. The consistency of these stages is important in the reliability of the models. The other crucial thing is the runtime management that is scheduling the model execution, managing the allocation of memory, and input/output operations. In real-time systems, these activities should be undertaken at low overhead to prevent bottlenecks. Asynchronous execution and pipeline parallelism are some of the techniques that are usually employed to enhance efficiency.

In spite of having advanced frameworks, the implementation of AI models in real-world remote sensing systems is still not easy[70]. Problems like compatibility, scalability, and maintainability have to be addressed, especially when it is deployed on a large scale or long-term. The building of common deployment workflows and tools is hence a significant research field.

4.5. Pipeline Optimization and Data Flow Management

To have real-time performance in preprocessing systems, it is not just the efficient models that should be used but also the optimized data pipelines. Data flow management is a process of coordinating the flow and transformation of data within the system in order to ensure that every process is both efficient and free of unnecessary delays[87].

Parallel processing is one of the most important optimization techniques of the pipeline since several steps in the pipeline are performed simultaneously[88]. As an example, when a denoising model is being trained to process a single image tile, another image tile can be processed to perform geometric correction. This calculation overlap minimizes idleness and enhances throughput. Nonetheless, it should be handled with close concurrence so as to maintain the consistency of data and prevent race conditions. Streaming architectures additionally improve efficiency since they do not have to wait till the entire datasets are available, but they can work on the incoming data. This method is especially suitable when dealing with continuous streams of data from satellites or UAVs. Nevertheless, streaming presents buffering, latency management, and error correction issues that need to be resolved with sound system design.

Another important factor is memory management, especially where resources are limited [89]. Effective utilization of memory may play a vital role in the performance of the system since the complex data transfer between the memory and the processing units may turn out to be a bottleneck. Data compression, tiling, and in-place processing are the common techniques that are employed to alleviate this problem. Distributed systems are also important in load balancing, whereby the processing tasks are spread to several devices or nodes. Maintaining the aspect of ensuring that every node is at its optimum capacity and not a bottleneck is a complicated issue, especially in dynamic environments where data rates and system conditions might be fluctuating.

In general, an optimization of pipelines has to be a whole ecosystem where both data flows and computations are taken into account [17]. Failure to focus on one may result in an inefficient performance, although there might be high efficiency of individual components.

4.6. Real-Time Constraints: Latency, Throughput, and Energy Efficiency

The property of real-time preprocessing systems is the possibility to fulfill the high-performance requirements, such as latency, throughput, and energy consumption [90]. Latency is defined as the amount of time it takes to process one unit of data, whereas throughput is a metric that indicates how many units are processed per unit. These metrics usually go hand in hand, and one may improve the other. Latency is the most important issue in most remote sensing applications, especially when time is an issue, as in disaster monitoring or surveillance. To obtain a low latency, it is necessary to reduce the computational and communication delay, which in turn must be achieved through effective model design, hardware acceleration, and efficient data pipelines.

In high-volume data cases, throughput becomes a critical factor, e.g., the constant satellite imaging [91]. Processing of large volumes of data should not be constrained by systems, and this necessitates scalable architectures and resource-efficient operations. There are more effective techniques, like batching and parallel processing, which can enhance the throughput, but they can also cause extra latency, which is why a thorough analysis of trade-offs is paramount. Another important factor to be taken into account is energy efficiency, especially in edge devices and onboard systems. The constraints of power restrict the models and continuous execution time. The concept of energy-efficient design not only tries to optimize the models and hardware, but it also deals with system-wide aspects of the cooling and power distribution.

To provide a balance between these constraints, a clear picture of system dynamics and the requirements of the application is necessary [92]. Practically, optimum performance can be attained with repetitive tuning and improvement, and through the application of monitoring and feedback mechanisms to control system behavior in real time.

4.7. Integration with Downstream Remote Sensing Applications

The end product of preprocessing is to optimize the performance of the downstream applications, namely object detection, classification, and change detection [93]. Consequently, the design of preprocessing systems should take into account how they affect these operations, as opposed to how they affect intermediate image quality measures.

The current tendencies are directed at the unification of preprocessing and analysis into common systems, at unified optimization of both stages [94]. As an example, end-to-end learning may include preprocessing modules as part of a larger network that permits gradients to pass throughout the pipeline. This integration is possible to provide better performance of the tasks; however, it also complicates models and makes them harder to train. Interoperability with current geospatial information systems (GIS) and data standards is another factor to be considered. Downstream tools and workflows need to be compatible with preprocessed data, and may enforce a particular format or coordinate system. This compatibility is another complexity added to system design. In addition, the dynamic character of the real-life applications demands that the preprocessing systems be flexible and receptive. As an example, the system might need to give priority to some areas or activities in evolving disaster situations that are evolving. Such dynamic configuration and control mechanisms are needed to ensure that such adaptability is incorporated.

Altogether, the creation of real-time preprocessing systems and their design is a complicated mixture of algorithmic, hardware, and system-level considerations [70]. With the combination of light AI methods and optimization of architectures and deployment solutions, it can result in efficient and scalable solutions satisfying the demands of the modern remote sensing applications. **Figure 3** shows a unified edge cloud collaborative architecture of real-time remote sensing preprocessing and downstream analytics.

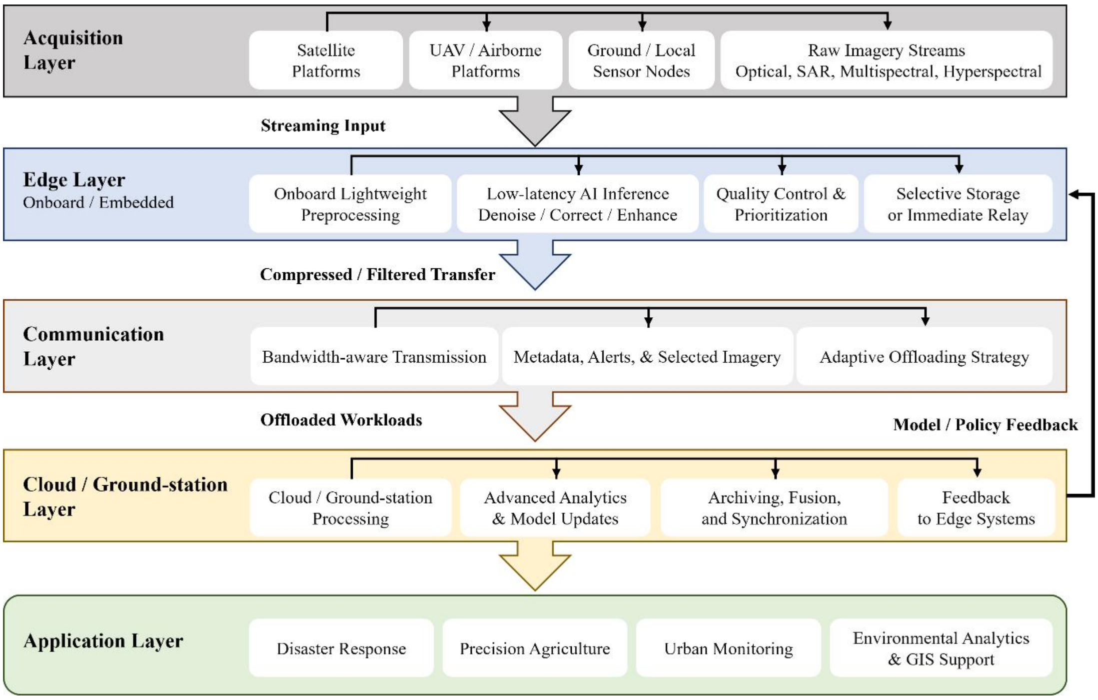

Figure 3. Edge–cloud collaborative architecture for lightweight AI-driven real-time preprocessing, showing data flow from acquisition platforms to edge inference, communication layers, cloud processing, and application outputs.

5. Applications and Case Studies

5.1. Role of Real-Time Preprocessing in Operational Remote Sensing

The growing need to obtain timely and actionable information has transformed preprocessing into a mission-driven provision of remote sensing systems rather than a preparation phase for operational systems. Preprocessing in traditional workflows can be done offline, which tends to cause delays that do not support real-time decision-making. But with a higher shift in the remote sensing applications to continuous monitoring and rapid response, real-time preprocessing of imagery becomes imperative [5].

Real-time preprocessing also provides the opportunity to improve the quality of raw data immediately and then use more reliable inputs in downstream analytical models [95]. It is especially crucial in those situations when the decisions have to be made within strict time limits, e.g., disaster response or surveillance. Preprocessing systems can have a direct and direct impact on the quality and strength of the validity and reliability of the following analysis by reducing noise and distortions, and by countering atmospheric effects in-situ.

Besides, real-time preprocessing enables the effective management of data through filtering and compression of data before transmission [5]. Since the bandwidth of communicating with the satellite is limited, or UAVs are limited in bandwidth, sending only meaningful and preprocessed information may greatly enhance the efficiency of the system. This is of particular importance in a large-scale monitoring system where the data volumes may easily overwhelm storage and transmission capabilities.

Although it is important, there are serious challenges associated with the integration of real-time preprocessing into working systems. These involve the ability to assure consistent performance under varied conditions, reliability of the system, as well as balancing between the constraints of computing and the quality of data. To solve these challenges, a high level of algorithms is needed, and the system should be designed and validated.

5.2. Disaster Monitoring and Emergency Response

Disaster monitoring and emergency response is another field in which real-time preprocessing has proven to be significantly effective[96]. Floods, wildfires, earthquakes, and hurricanes are some of the events that need prompt evaluation in order to inform the rescue efforts and allocation of resources. The remote sensing offers a distinct perspective on this kind of monitoring, yet the value of the collected information relies on the speed and the quality of the preprocessing.

SAR imagery can also be applied in other fields, such as flood monitoring, because it can cut through cloud cover and work in any weather condition[97]. Nevertheless, the speckle noise and geometric distortions influence SAR data, and it is necessary to deal with them prior to the completion of flood mapping. Denoising and correction models of lightweight AI-based algorithms can work with SAR images nearly in real time, enabling the responders to promptly locate the affected regions. Likewise, during wildfire detection, optical images need to be processed in order to help increase contrast and eliminate atmospheric interference. Online cloud-cutting and image-enhancement systems can be used to enhance the visibility of fire lines and smoke columns, facilitating better identification as well as tracking. Onboard pre-processing of data on UAVs or satellites makes the latency even lower, and timely information is delivered to ground teams.

One important lesson that we learnt during these applications is that the quality of preprocessing directly influences the results of decision-making[5]. False or slow preprocessing may cause the wrong interpretation of data, which may result in unproductive or even destructive acts. Thus, the importance of these systems is not the speed but rather their robustness and reliability.

Nevertheless, the disaster situations also shed some light on the weaknesses of the existing methods[98]. The issues of data heterogeneity, highly dynamic conditions, and the absence of ground truth data are major challenges to AI-based preprocessing models. It is an open research question to make lightweight models capable of generalizing between such conditions.

5.3. Precision Agriculture and Environmental Monitoring

Another field of application of real-time preprocessing is precision agriculture. Crop health, soil conditions, and irrigation requirements are broadly monitored using remote sensing images[99,100]. Nevertheless, the usefulness of such applications relies on the quality of the input data, which is usually impacted by noise, weather conditions, and sensor variations. Real-time preprocessing can help the farmer monitor agricultural fields continuously and make decisions concerning fertilization, irrigation, and pests on time. As an illustration, the lightweight AI models can be used to conduct an atmospheric correction and noise removal on-the-fly so that vegetation indices like NDVI are calculated correctly[101]. This is especially useful when using UAV systems where data are gathered at a high rate and must be processed in real time.

Real-time preprocessing is also applicable in environmental monitoring applications such as the tracking of deforestation, water quality, and air pollution[102]. Time-series data used in such situations are forced to identify long-term trends, and, therefore, a consistent and reliable preprocessing is necessary on two or more observations. The preprocessing can be standardized with the help of lightweight AI models, which will enhance the comparability of outcomes. Nevertheless, there are also difficulties in scalability and generalization that are identified with these applications. The agricultural and environmental conditions are diverse by region and season and few models can be designed to work in the same way. Further complicating preprocessing, there is the integration of multisource data, e.g., integration of optical and SAR images.

One of the main findings is that, besides the improved quality of the data, real-time preprocessing also allows making adaptive decisions[103]. These systems have the potential to assist in efficient and sustainable management of the natural resources by ensuring they have the right information at the right time.

5.4. Urban Monitoring and Smart City Applications

Cities offer specific solutions and opportunities that present real-time remote sensing preprocessors. Some of the applications of high-resolution images include traffic control, infrastructure evaluation, and town planning [104]. The information in these applications needs to be accurate and time-limited, which might be on fine spatial and temporal scales.

The large data volumes that are related to urban monitoring require real-time preprocessing. Indicatively, UAVs used in traffic surveillance produce continuous streams of high-resolution images that need to be computed on the fly to identify vehicles and interpret traffic patterns [105]. Lightweight AI models make it possible to perform efficient denoising, geometric correction, and enhancement, as well as to make downstream detection algorithms efficient. Remote sensing data are typically combined with other data in smart city applications, including IoT sensors and geographic information systems. Preprocessing is an important process in guaranteeing consistency and compatibility of data in these sources. As an example, to track remote sensing imaging, it needs an accurate geometric correction in order to reconcile the remote sensing imaging data with the pre-existing maps and infrastructure data [105].

The other factor is that of real-time responsiveness. Other applications, like emergency traffic management or infrastructure monitoring, need real-time information, and this can only be realized with effective preprocessing pipelines [106]. Nevertheless, preprocessing models face a lot of difficulties due to the complexity of the urban setting, which comprises occlusions, shadows, and dynamic variations.

At the system level, city applications tend to use distributed architecture in which data are computed on both ends of a graph, at edge devices and cloud computing server nodes [107]. The need to ensure a smooth integration and synchronization of these components is important in promoting system performance.

5.5. UAV-Based and Onboard Satellite Preprocessing Systems

The usage of preprocessing systems on UAVs and satellites is a breakthrough in the use of remote sensing technology. These systems can radically decrease latency and bandwidth requirements by doing preprocessing on the platform where the data is being collected. The UAV-based systems are especially suitable in real-time systems, since the systems have a relatively low altitude, and can offer high-resolution imagery with adaptable coverage. UAVs can have lightweight AI models deployed on-board embedded processors to assist in tasks like denoising, enhancement, and preliminary analysis. This can be used in search and rescue, precision agriculture, and infrastructure inspection by allowing quick decisions to be made in these fields [108].

There is a further drawback in onboard satellite preprocessing because of the high power, memory, and computational limitations [109]. Nevertheless, with current innovations in equipment and lightweight artificial intelligence, preprocessing in orbit has become more possible. Onboard systems may, for instance, eliminate cloud-covered pictures or do prior correction before sending information to ground stations and save on the amount of data transmitted. The onboard preprocessing has a critical benefit in the form of relevancy-based prioritization of data transmission [110]. These systems can use limited bandwidth more effectively by transmitting only high-quality or high-priority data over a limited communication bandwidth. Nonetheless, this demands effective preprocessing models that can be used in various and unforeseeable conditions.

The use of onboard systems also presents significant issues of reliability, maintainability, and upgradability despite these advantages. Hardware and software upgrades are hard or impossible after deployment, which requires well-tested and robust designs. This limitation explains why lightweight and efficient models capable of working over long distances are important.

5.6. Comparative Analysis of Traditional and AI-Driven Preprocessing

Replacement of traditional with AI-based preprocessing is a paradigm change in remote sensing processes. Physically-modeled, handwritten algorithms provide interpretability and familiar behavior but tend to be less flexible and efficient. On the other hand, AI-based strategies are more flexible and perform better, but they come with issues of interpretability and generalization [111,112].

Comparative analysis has indicated that AI-based techniques can usually beat conventional ones in activities like denoising and removing clouds, especially in harsh or noisy conditions. But these improvements are not always equally applicable[113]. In other situations, old methods can also offer more stable and predictable outcomes, particularly where physical models are well-characterized. Computationally, AI-based approaches can be more efficient to run on a particular hardware, in particular, with lightweight architectures. The initial training and optimization of such models, however, can be resource-intensive, and their implementation must be sensitive to the hardware and software limitations.

The other distinction that is important is that of integrating preprocessing and downstream tasks[114]. The AI-inspired methods are more susceptible to end-to-end optimization, which allows the joint preprocessing and analysis learning. This integration may help to achieve better overall performance; however, it also makes the systems more complex.

Finally, the decision between the conventional and AI-based preprocessing is not a black-and-white one. Integration The integration of both paradigms is also taking a promising path as hybrid approaches that involve the strong elements of both paradigms. An example is the ability to make AI models more interpretable and robust through physical constraints, and to make traditional methods more flexible and efficient with the help of AI.

5.7. Deployment Challenges and Practical Considerations

Although the promise of lightweight AI-based preprocessing systems can be seen, their implementation is associated with many challenges[70]. Data heterogeneity is one of the main problems because data provided by the remote sensing are collected by different sensors with diverse features. It is necessary to have models that are able to generalize in order to ensure that the results of the preprocessing are comparable across these datasets.

The second problem is that there is no labeled data to train and validate[112]. Ground truth data is very limited or non-existent in most instances, especially when it comes to preprocessing (atmospheric correction). This restriction requires the application of unsupervised or self-supervised methods of learning, which are in a state of active development. Reliability in the system is also one of the critical issues, especially in mission-critical systems. Preprocessing systems have to work with different conditions, and under varying conditions, they should be able to work continuously and implement the necessary error-handling and fault-tolerance mechanisms. Moreover, the preprocessing may be complicated to integrate with a current workflow and infrastructure, and needs thorough planning and standardization.

Last but not least, the ethical and regulatory issues have to be considered, especially when using such applications with surveillance or sensitive data[115]. The privacy of data and the ability to adhere to regulations are other challenges to system design and deployment. To sum up, the use of lightweight AI-based real-time preprocessing systems has a very broad field of use, and each area has its own requirements and challenges. These application context differences, differences in preprocessing priority, and system differences are listed in **Table 4**. Although a lot has been achieved, more research and development are required in order to overcome the still existing challenges and fully utilize the potential of these systems in the operational environment.

Table 4. Representative application scenarios of lightweight AI-driven real-time preprocessing systems, including data types, preprocessing priorities, real-time requirements, and operational constraints.

Application	Data Type	Preprocessing Tasks	Real-Time Requirement	System Constraints	Impact
Disaster Monitoring	SAR, Optical	Denoising, cloud removal	Very High	Low latency, high reliability	Rapid response
Precision Agriculture	Multispectral	Atmospheric correction, enhancement	High	Edge deployment	Crop optimization
Urban Monitoring	High-res optical	Geometric correction, denoising	High	High throughput	Smart city analytics
Environmental Monitoring	Multisource	Standardization, enhancement	Medium	Long-term consistency	Sustainability insights
UAV Surveillance	Optical	Real-time enhancement	Very High	Power & memory limits	Immediate decision-making

6. Conclusion and Future Perspectives

6.1. Assessment of Insights

This review has exploited the changing nature of remote sensing preprocessing in terms of lightweight artificial intelligence and real-time system design. An analysis indicates the obvious shift from classical pipelines, which are model-oriented, to integrated, data-driven models that focus on efficiency, adaptability, and deployment feasibility. Preprocessing, which used to be a pre-processing task, has been realized to be a key enabling factor for sound remote sensing analytics.

One of the key lessons is that the efficiency of preprocessing cannot be considered as an independent variable. It possesses value in the fact that it can improve downstream tasks and still be functional within the stringent system parameters. Lightweight AI systems have become a major solution to this predicament, with complex preprocessing functions being able to be implemented on resource-limited platforms. Nonetheless, the success of these methods is determined by a thin line between accuracy, efficiency, and robustness, and is to be maintained carefully by means of algorithm design and system-level optimization.

Another point which is made in the review is the necessity to think of the whole pipeline of processing, which starts with the acquisition of data and ends with the ultimate use of data. They can hardly satisfy the needs of real-time applications using fragmented methodologies that maximize the efforts of individual components but ignore their integration into systems. This needs to be replaced by comprehensive systems incorporating both preprocessing and downstream analytics and hardware requirements, to enable scalable and realistic solutions.

6.2. Current Limitations and Open Challenges

Various constraints in the testing of lightweight AI-driven preprocessing systems still exist despite the major improvements in the field. Generalization over a wide range of sensors, environments, and conditions of acquisition is one of the most pressing issues. The data from remote sensing are always heterogeneous, and models trained on certain sets are usually not able to work consistently in different situations. This weakness is especially crucial when working with real-time systems where dynamism with regard to shifting circumstances is necessary. The other big problem is that labeled data of high quality is in short supply to be used during preprocessing activities. In contrast to classification or detection, preprocessing does not always have a well-defined ground truth, so a supervised learning method can be challenging to use. Although unsupervised and self-supervised approaches seem to be promising alternatives, they are still at their initial stages of development and need improvements in order to reach reliable performance.

The trade-off between efficiency and accuracy is also one of the basic limitations. Although lightweight models can be used for real-time processing, performance can be compromised in complicated conditions, e.g., dense cloud cover or noisy SAR images. It is a continuous challenge to make sure that these models are faithful enough to undertake downstream tasks. Another barrier is deployment complexity, which comes up at the system level. Combining AI models with software frameworks, hardware platforms, and data pipelines demands a lot of expertise and integration. There are issues like compatibility, scalability, and maintainability that need to be solved so that the system can be reliable in the long term.

Lastly, methods and progress cannot be compared as there are no standard benchmarks and evaluation protocols. In the absence of stable metrics and data sets, it becomes hard to know which methods can actually work under real-life circumstances.

6.3. Emerging Trends and Technological Directions

The future of real-time preprocessing in remote sensing is directly linked to various other emerging trends in technology. The development of edge AI and TinyML is one of the most notable ones and is designed to provide intelligent processing to smaller and smaller devices. These technologies can be used to perform preprocessing on a nanosatellite, sensor node, and other ultra-low-power platforms, dramatically increasing the range of real-time applications. The other trend that is worth considering is the emergence of self-supervised and foundation models of remote sensing. By using big data of unlabeled data, these models can be trained to acquire generalizable representations, which can be further transferred to different preprocessing tasks. This can be used to overcome the weakness of data scarcity and enhance the model's robustness in various areas.

Federated and distributed learning systems are also becoming a focus in applications, especially in cases where data may not be centralized because of bandwidth or privacy limits. These solutions can be used to share the model training between two or more nodes and may enhance performance, and also preserve data locality. Moreover, the development of neural architecture search (NAS) and automated machine learning (AutoML) should be of great importance in lightweight model design. These techniques are able to create task- and hardware platform-specific models (by automating the architecture optimization process) that can be more efficient and perform better.

Innovation in hardware will also be a force in this sector. The creation of specialized accelerators, including AI chips and neuromorphic processors, provides unprecedented ways of efficient and scalable preprocessing. Nevertheless, they can be fully developed by being tightly coupled with algorithm design and software frameworks.

6.4. Toward Unified and Adaptive Preprocessing Frameworks

The most important way to approach research in the future is by creating a set of common preprocessing systems, where two or more tasks are carried out in a common and cohesive system. These frameworks do not separate denoising, correction, and enhancement but instead seek to optimize both together and may yield better efficiency and consistency. Another avenue is adaptive preprocessing systems. These systems are dynamic systems that can change their action in response to the properties of input data, environmental conditions, or application needs. Examples include a system redirecting more computing resources to areas of interest or changing model parameters as the amount of noise changes. The flexibility at this scale will be possible only with the development of means of real-time monitoring and control, not to mention the design of models.

It will also be important to have interoperability and standardization to enable widespread adoption. Integration of various systems and applications can be supported by coming up with common interfaces, data formats, and evaluation protocols. This comes in especially handy in large-scale deployments, where a number of components need to cooperate with each other.

Moreover, both the implementation of upstream analytics and downstream analytics into end-to-end learning models, as well as the incorporation of preprocessing into these models, hold a lot of promise. These strategies have the ability to enhance the overall performance of the entire system and minimize redundancy through the optimization of the whole pipeline. Nevertheless, they also come with some challenges that concern model complexity, training stability, and interpretability.

6.5. Broader Implications and Interdisciplinary Opportunities

The development of lightweight AI-based preprocessing systems is not only associated with remote sensing, but it can also be applied in other areas that demand real-time image and signal processing with resource limitations. Similar methods can be used in fields like medical imaging, autonomous systems, environmental sensing, and so forth, which underscores the cross-disciplinary nature of this field of research.

Interdisciplinary cooperation, such as remote sensing, machine learning, hardware engineering, and systems design, will be necessary to deal with complex challenges that have been identified in this review. Both areas have their own insights and knowledge, and it is essential to combine them to create viable and sustainable solutions. Social and ethical aspects are also becoming more significant. The matter of data privacy, surveillance, and the environmental impact will have to be taken into consideration as remote sensing systems are becoming increasingly capable and widespread. It is a significant issue for future research to design preprocessing systems that are efficient and accurate, not to mention that they have to be responsible and transparent.

6.6. Concluding Remarks

To sum up, remote sensing is a field where the design and implementation of lightweight AI-based real-time preprocessing systems are a critical frontier. These systems can revolutionize the ways remote sensing data are used in a vast number of applications because they can facilitate efficient, scalable, and timely data processing. Although this has come a long way, a lot more needs to be done, especially in the areas of generalization, robustness, and system integration. Their solutions will need further algorithm, hardware, and system design innovation, a standardized framework, and an evaluation methodology.

It is a promising direction for the next-generation remote sensing systems; the combination of lightweight AI, edge computing, and advanced sensing technologies has a promising way to go. Since research in this field is still in development, one can anticipate that it will become a key to making the Earth's environment a more responsive, intelligent, and sustainable way of monitoring.

Funding

This work received no external funding.

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

No new data were created or generated in this study. As this is a review, it is based on data and information from previously published sources, which are cited in the reference list.

Conflicts of Interest

The authors declare no conflict of interest.

AI Use Statement

The authors declare that no artificial intelligence (AI) tools were used in the preparation of this manuscript.

References

[1] Prost, G.L., 2025. Remote Sensing for Geoscientists: Image Analysis and Integration, 4th ed. CRC Press: Boca Raton, FL, USA. DOI: https://doi.org/10.1201/9781003617310

[2] Yao, H., Qin, R., Chen, X., 2019. Unmanned aerial vehicle for remote sensing applications—A review. Remote Sensing. 11(12), 1443. DOI: https://doi.org/10.3390/rs11121443

[3] Abdulmuttalib, H., Al Doori, M., Barsi, Á., et al., 2024. Remote sensing data quality in the era of AI. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 10, 1–11. DOI: https://doi.org/10.5194/isprs-annals-X-3-2024-1-2024

[4] Pathak, D.K., Bisht, A.K., 2025. Preprocessing Techniques for Raw Satellite Images: A Comprehensive Review. In Proceedings of the 14th International Conference on System Modeling & Advancement in Research Trends (SMART), Moradabad, India, 14–15 November 2025; pp. 585–591. DOI: https://doi.org/10.1109/SMART66937.2025.11389685

[5] Qi, B., Shi, H., Zhuang, Y., et al., 2018. On-board, real-time preprocessing system for optical remote-sensing imagery. Sensors. 18(5), 1328. DOI: https://doi.org/10.3390/s18051328

[6] Fitz, S., Romero, P., 2021. Neural networks and deep learning: A paradigm shift in information processing, machine learning, and artificial intelligence. In: Rau, R., Wardrop, R., Zingales, L. (Eds.). The Palgrave Handbook of Technological Finance. Palgrave Macmillan: Cham, Switzerland. pp. 589–654. DOI: https://doi.org/10.1007/978-3-030-65117-6_22

[7] Krejcar, O., Namazi, H., 2026. AI in remote sensing and satellite image processing—A review. Environmental Earth Sciences. 85(3), 78. DOI: https://doi.org/10.1007/s12665-025-12798-w

[8] Yuan, Q., Shen, H., Li, T., et al., 2020. Deep learning in environmental remote sensing: Achievements and challenges. Remote Sensing of Environment. 241, 111716. DOI: https://doi.org/10.1016/j.rse.2020.111716

[9] Lakshmi, K., Parvathavarthini, S., Thangavel, M., 2026. Applications of Aerial Aided Edge Computing in Disaster Management and Emergency Response, Environmental Monitoring and Conservation, Precision Agriculture and Crop Management, Infrastructure Inspection and Maintenance, Surveillance, and Public Safety. In: Juneja, S., Dhanaraj, R.K., Juneja, A. (Eds.). Edge Computing and Aerial Platforms. Scrivener Publishing LLC: Beverly, MA, USA. pp. 63–102. DOI: https://doi.org/10.1002/9781394336326.ch3

[10] Zeeshan, M., 2024. Efficient Deep Learning Models for Edge IoT Devices—A Review. TechRxiv. DOI: https://doi.org/10.36227/techrxiv.172254372.21002541/v1

[11] Krichen, M., Abdalzaher, M.S., Shaaban, M., et al., 2025. Lightweight AI for Drones: A Survey. In Proceedings of the 7th International Youth Conference on Radio Electronics, Electrical and Power Engineering

(REEPE), Moscow, Russia, 8–10 April 2025; pp. 1–6. DOI: https://doi.org/10.1109/REEPE63962.2025.10970940

[12] Rezvani, S., Soleymani Siahkar, F., Rezvani, Y., et al., 2024. Single image denoising via a new lightweight learning-based model. IEEE Access. 12, 121077–121092. DOI: https://doi.org/10.1109/ACCESS.2024.3450842

[13] Liu, Y., Kong, L., Chen, G., et al., 2021. Light-weight AI and IoT collaboration for surveillance video pre-processing. Journal of Systems Architecture. 114, 101934. DOI: https://doi.org/10.1016/j.sysarc.2020.101934

[14] Hossain, M.E., Tarafder, M.T.R., Ahmed, N., et al., 2023. Integrating AI with edge computing and cloud services for real-time data processing and decision making. International Journal of Multidisciplinary Sciences and Arts. 2(4), 252–261.

[15] Singh, M.P., Gayathri, V., Chaudhuri, D., 2022. A simple data preprocessing and postprocessing techniques for SVM classifier of remote sensing multispectral image classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. 15, 7248–7262. DOI: https://doi.org/10.1109/JSTARS.2022.3201273

[16] Ficili, I., Giacobbe, M., Tricomi, G., et al., 2025. From sensors to data intelligence: Leveraging IoT, cloud, and edge computing with AI. Sensors. 25(6), 1763. DOI: https://doi.org/10.3390/s25061763

[17] Zhang, S., Xue, Y., Zhou, X., et al., 2022. State of the art: High-performance and high-throughput computing for remote sensing big data. IEEE Geoscience and Remote Sensing Magazine. 10(4), 125–149. DOI: https://doi.org/10.1109/MGRS.2022.3204590

[18] Sang, H., Zhang, L., Chen, T., et al., 2026. Onboard Deployment of Remote Sensing Foundation Models: A Comprehensive Review of Architecture, Optimization, and Hardware. Remote Sensing. 18(2), 298. DOI: https://doi.org/10.3390/rs18020298

[19] McCall, A., 2025. Edge AI: Challenges and Opportunities in Real-Time Processing. Available from: https://www.researchgate.net/publication/392757984_Edge_AI_Challenges_and_Opportunities_in_Real-Time_Processing (cited 5 April 2026).

[20] Liu, P., 2015. A survey of remote-sensing big data. Frontiers in Environmental Science. 3, 45. DOI: https://doi.org/10.3389/fenvs.2015.00045

[21] Vali, A., Comai, S., Matteucci, M., 2020. Deep learning for land use and land cover classification based on hyperspectral and multispectral earth observation data: A review. Remote Sensing. 12(15), 2495. DOI: https://doi.org/10.3390/rs12152495

[22] Palaniyandi, M., Manivel, P., Sharmila, T., et al., 2021. The use of multispectral (MSS) and synthetic aperture radar (SAR) microwave remote sensing data to study environment variables, land use/land cover changes, and recurrent weather condition for forecast malaria: A systematic review. Applied Ecology and Environmental Sciences. 9(4), 490–501. DOI: https://doi.org/10.12691/aees-9-4-10

[23] Prudyus, I., Tkachenko, V., Kondratov, P., et al., 2015. Factors affecting the quality of formation and resolution of images in remote sensing systems. Computational Problems of Electrical Engineering. 5(1), 41–46.

[24] Elachi, C., van Zyl, J.J., 2021. Introduction to the Physics and Techniques of Remote Sensing. John Wiley & Sons: Hoboken, NJ, USA. DOI: https://doi.org/10.1002/9781119523048

[25] Kobayashi, S., Sanga-Ngoie, K., 2008. The integrated radiometric correction of optical remote sensing imageries. International Journal of Remote Sensing. 29(20), 5957–5985. DOI: https://doi.org/10.1080/01431160701881889

[26] Richter, R., 2011. Atmospheric correction methods for optical remote sensing imagery of land. In: Weng, Q. (Ed.). Advances in Environmental Remote Sensing: Sensors, Algorithms, and Applications. CRC Press: Boca Raton, FL, USA. pp. 161–172. DOI: https://doi.org/10.1201/b10599-14

[27] Toutin, T., 2011. State-of-the-art of geometric correction of remote sensing data: A data fusion perspective. International Journal of Image and Data Fusion. 2(1), 3–35. DOI: https://doi.org/10.1080/19479832.2010.539188

[28] Zhai, H., Zhang, H., Zhang, L., et al., 2018. Cloud/shadow detection based on spectral indices for multi/hyperspectral optical remote sensing imagery. ISPRS Journal of Photogrammetry and Remote Sensing. 144, 235–253. DOI: https://doi.org/10.1016/j.isprsjprs.2018.07.006

[29] Yang, Q., Jin, C., Li, T., et al., 2022. Research progress and challenges of data-driven quantitative remote sensing. National Remote Sensing Bulletin. 26(2), 268–285. (in Chinese)

[30] Wang, X., Yi, J., Guo, J., et al., 2022. A review of image super-resolution approaches based on deep learning and applications in remote sensing. Remote Sensing. 14(21), 5423. DOI: https://doi.org/10.3390/rs14215423

[31] Jin, Z., Iqbal, M.Z., Bobkov, D., et al., 2019. A flexible deep CNN framework for image restoration. IEEE Transactions on Multimedia. 22(4), 1055–1068. DOI: https://doi.org/10.1109/TMM.2019.2938340

[32] Liu, X., Lu, D., Zhang, A., et al., 2022. Data-driven machine learning in environmental pollution: Gains and problems. Environmental Science & Technology. 56(4), 2124–2133. DOI: https://doi.org/10.1021/acs.est.1c06157

[33] Zhu, X.X., Tuia, D., Mou, L., et al., 2017. Deep

learning in remote sensing: A comprehensive review and list of resources. IEEE Geoscience and Remote Sensing Magazine. 5(4), 8–36. DOI: https://doi.org/10.1109/MGRS.2017.2762307

[34] Tuia, D., Persello, C., Bruzzone, L., 2016. Domain adaptation for the classification of remote sensing data: An overview of recent advances. IEEE Geoscience and Remote Sensing Magazine. 4(2), 41–57. DOI: https://doi.org/10.1109/MGRS.2016.2548504

[35] Huang, D., Zhou, Z., Zhang, Z., et al., 2025. From Application-Driven Growth to Paradigm Shift: Scientific Evolution and Core Bottleneck Analysis in the Field of UAV Remote Sensing. Applied Sciences. 15(15), 8304. DOI: https://doi.org/10.3390/app15158304

[36] Sara, U., Akter, M., Uddin, M.S., 2019. Image quality assessment through FSIM, SSIM, MSE and PSNR—A comparative study. Journal of Computer and Communications. 7(3), 8–18. DOI: https://doi.org/10.4236/jcc.2019.73002

[37] Baig, M.A., Moinuddin, A.A., Khan, E., 2019. PSNR of highest distortion region: An effective image quality assessment method. In Proceedings of the International Conference on Electrical, Electronics and Computer Engineering (UPCON), Aligarh, India, 8–10 November 2019; pp. 1–4. DOI: https://doi.org/10.1109/UPCON47278.2019.8980171

[38] Wu, H., Li, Z.-L., 2009. Scale issues in remote sensing: A review on analysis, processing and modeling. Sensors. 9(3), 1768–1793. DOI: https://doi.org/10.3390/s90301768

[39] Kamusoko, C., 2019. Pre-processing. In Remote Sensing Image Classification in R. Springer: Singapore. pp. 25–66. DOI: https://doi.org/10.1007/978-981-13-8012-9_2

[40] Xu, Q., Guan, X., Cao, J., et al., 2021. MPR-GAN: A Novel Neural Rendering Framework for MLS Point Cloud with Deep Generative Learning. IEEE Transactions on Geoscience and Remote Sensing. 60, 1–16. DOI: https://doi.org/10.1109/TGRS.2022.3212389

[41] Ma, Y., Wu, H., Wang, L., et al., 2015. Remote sensing big data computing: Challenges and opportunities. Future Generation Computer Systems. 51, 47–60. DOI: https://doi.org/10.1016/j.future.2014.10.029

[42] Rathore, M.M.U., Paul, A., Ahmad, A., et al., 2015. Real-time big data analytical architecture for remote sensing application. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. 8(10), 4610–4621. DOI: https://doi.org/10.1109/JSTARS.2015.2424683

[43] Liu, H.-I., Galindo, M., Xie, H., et al., 2024. Lightweight deep learning for resource-constrained environments: A survey. ACM Computing Surveys. 56(10), 1–42. DOI: https://doi.org/10.1145/3657282

[44] Bablu, T.A., Rashid, M.T., 2025. Edge computing and its impact on real-time data processing for IoT-driven applications. Journal of Advanced Computing Systems. 5(1), 26–43.

[45] Xie, R., Tang, Q., Wang, Q., et al., 2020. Satellite-terrestrial integrated edge computing networks: Architecture, challenges, and open issues. IEEE Network. 34(3), 224–231. DOI: https://doi.org/10.1109/MNET.011.1900369

[46] Yang, C., Chen, X., Shao, G., et al., 2025. Integrating Conventional Models and AI for Intelligent Sensor Data Processing: A Review. IEEE Sensors Journal. 25(20), 37569–37585. DOI: https://doi.org/10.1109/JSEN.2025.3602686

[47] Zangari, A., Marcuzzo, M., Rizzo, M., et al., 2024. Hierarchical Text Classification and Its Foundations: A Review of Current Research. Electronics. 13(7), 1199. DOI: https://doi.org/10.3390/electronics13071199

[48] Oneto, L., Navarin, N., Micheli, A., et al., 2026. Informed machine learning for complex data. Neurocomputing. 669, 132505. DOI: https://doi.org/10.1016/j.neucom.2025.132505

[49] Dinh, T., Wong, H., Lisik, D., et al., 2026. Data clustering: A fundamental method in data science and management. Data Science and Management. 9(1), 100161. DOI: https://doi.org/10.1016/j.dsm.2025.08.001

[50] Choi, H., Jeong, J., 2019. Speckle Noise Reduction Technique for SAR Images Using Statistical Characteristics of Speckle Noise and Discrete Wavelet Transform. Remote Sensing. 11(10), 1184. DOI: https://doi.org/10.3390/rs11101184

[51] Vaddepally, D., 2025. Lightweight Deep Learning for Resource-Constrained Devices. International Journal on Science and Technology. 16(2). DOI: https://doi.org/10.71097/IJSAT.v16.i2.6224

[52] Yang, R., Chen, Z., Wang, B., et al., 2023. A lightweight detection method for remote sensing images and its energy-efficient accelerator on edge devices. Sensors. 23(14), 6497. DOI: https://doi.org/10.3390/s23146497

[53] Faris, W.F., Gupta, N., Dhadwal, P., et al., 2025. Lightweight AI Models for Deployment. In: Goar, V., Gajrani, J., Yousuf, M.A., et al. (Eds.). Proceedings of the International Conference on Computational Complexity and Intelligent Algorithms: IC3IA 2025. Springer: Singapore. pp. 313–327. DOI: https://doi.org/10.1007/978-981-96-7473-2_22

[54] Mittal, A., 2025. The Evolution of Edge AI: A New Paradigm in Decentralized Cloud Computing. Iconic Research and Engineering Journals. 8(11), 2185–2196. Available from: https://www.researchgate.net/profile/Anant-Mittal-7/publication/392401393_The_Evolution_of_Edge_AI_A_New_Paradigm_in_Decentralized_Cloud_Computing/links/68407c488a76251f22eb7b41/The-Evolution-of-Edge-AI-A-New-Paradig

m-in-Decentralized-Cloud-Computing.pdf

[55] Cari, K.U.-D., Mercury, M., Alaba, G., 2024. A snapshot of tiny AI: Innovations in model compression and deployment. AUTHOREA. DOI: https://doi.org/10.22541/au.172894062.27932664/v1

[56] Hubara, I., Courbariaux, M., Soudry, D., et al., 2018. Quantized neural networks: Training neural networks with low precision weights and activations. Journal of Machine Learning Research. 18, 1–30. Available from: https://jmlr.org/papers/volume18/16-456/16-456.pdf

[57] Fang, L., Yu, X., Cai, J., et al., 2026. Knowledge distillation and dataset distillation of large language models: Emerging trends, challenges, and future directions. Artificial Intelligence Review. 59(1), 17. DOI: https://doi.org/10.1007/s10462-025-11423-3

[58] Badawy, H., 2025. Deep Learning-Based Change Detection in High-Resolution Remote Sensing Imagery [PhD Thesis]. The American University in Cairo: Cairo, Egypt.

[59] Li, M., 2025. Application of lightweight convolutional neural networks in image classification [Bachelor's Thesis]. Lappeenranta–Lahti University of Technology LUT: Lappeenranta, Finland.

[60] Hung, K.-W., Zhang, Z., Jiang, J., 2019. Real-time image super-resolution using recursive depthwise separable convolution network. IEEE Access. 7, 99804–99816. DOI: https://doi.org/10.1109/ACCESS.2019.2929223

[61] Fan, X., Yan, C., Fan, J., et al., 2022. Improved U-net remote sensing classification algorithm fusing attention and multiscale features. Remote Sensing. 14(15), 3591. DOI: https://doi.org/10.3390/rs14153591

[62] Yuan, Q., Zhang, L., Shen, H., 2012. Hyperspectral image denoising employing a spectral-spatial adaptive total variation model. IEEE Transactions on Geoscience and Remote Sensing. 50(10), 3660–3677. DOI: https://doi.org/10.1109/TGRS.2012.2185054

[63] Li, L., Li, X., Jiang, L., et al., 2021. A review on deep learning techniques for cloud detection methodologies and challenges. Signal, Image and Video Processing. 15(7), 1527–1535. DOI: https://doi.org/10.1007/s11760-021-01885-7

[64] Zhang, H., Li, Y., Jiang, Y., et al., 2019. Hyperspectral classification based on lightweight 3-D-CNN with transfer learning. IEEE Transactions on Geoscience and Remote Sensing. 57(8), 5813–5828. DOI: https://doi.org/10.1109/TGRS.2019.2902568

[65] Wang, C.-H., Huang, K.-Y., Yao, Y., et al., 2022. Lightweight deep learning: An overview. IEEE Consumer Electronics Magazine. 13(4), 51–64. DOI: https://doi.org/10.1109/MCE.2022.3181759

[66] Chaganti, K.C., 2025. A scalable, lightweight AI-driven security framework for IoT ecosystems: Optimization and game theory approaches. TechRxiv. DOI: https://doi.org/10.36227/techrxiv.173834640.08527422/v1

[67] Jiang, C., Hou, M., Wang, H., 2026. An Adaptive Compression Method for Lightweight AI Models of Edge Nodes in Customized Production. Sensors. 26(2), 383. DOI: https://doi.org/10.3390/s26020383

[68] Balemans, D., Reiter, P., Steckel, J., et al., 2022. Resource efficient AI: Exploring neural network pruning for task specialization. Internet of Things. 20, 100599. DOI: https://doi.org/10.1016/j.iot.2022.100599

[69] Merenda, M., Porcaro, C., Iero, D., 2020. Edge Machine Learning for AI-Enabled IoT Devices: A Review. Sensors. 20(9), 2533. DOI: https://doi.org/10.3390/s20092533

[70] Kazanskiy, N., Khabibullin, R., Nikonorov, A., et al., 2025. A comprehensive review of remote sensing and artificial intelligence integration: Advances, applications, and challenges. Sensors. 25(19), 5965. DOI: https://doi.org/10.3390/s25195965

[71] Baciu, V.-E., Stiens, J., da Silva, B., 2024. MLino bench: A comprehensive benchmarking tool for evaluating ML models on edge devices. Journal of Systems Architecture. 155, 103262. DOI: https://doi.org/10.1016/j.sysarc.2024.103262

[72] Chivaran, N., Ni, J., 2025. LAID: Lightweight AI-Generated Image Detection in Spatial and Spectral Domains. In Proceedings of the 22nd Annual International Conference on Privacy, Security, and Trust (PST), Fredericton, NB, Canada, 26–28 August 2025; pp. 1–10. DOI: https://doi.org/10.1109/PST65910.2025.11268874

[73] Wang, A., Liu, X., Zhang, K., et al., 2025. Neural architecture search for hyperspectral image classification: A comprehensive review and future perspectives. Remote Sensing. 17(15), 2727. DOI: https://doi.org/10.3390/rs17152727

[74] Somvanshi, S., Islam, M.M., Chhetri, G., et al., 2025. From tiny machine learning to tiny deep learning: A survey. ACM Computing Surveys. 58(7), 1–33. DOI: https://doi.org/10.1145/3776588

[75] Gui, J., Chen, T., Zhang, J., et al., 2024. A survey on self-supervised learning: Algorithms, applications, and future trends. IEEE Transactions on Pattern Analysis and Machine Intelligence. 46(12), 9052–9071. DOI: https://doi.org/10.1109/TPAMI.2024.3415112

[76] Barretta, D., Marini, L., Razzano, F., et al., 2026. Towards Real-Time Earth System Intelligence through Advancements in Onboard AI Processing. IEEE Geoscience and Remote Sensing Magazine. 14(3), 392–441. DOI: https://doi.org/10.1109/MGRS.2026.3665978

[77] Panda, S.S., Rao, M.N., Thenkabail, P.S., et al., 2024. Remote Sensing Systems—Platforms and Sensors: Aerial, Satellite, UAV, Optical, Radar, and LiDAR. In: Thenkabail, P.S. (Ed.). Remote Sensing Handbook, Volume I, 2nd ed. CRC Press: Boca Raton,

FL, USA. pp. 3–86. DOI: https://doi.org/10.1201/9781003541141-2

[78] Zheng, Y., He, B., Li, T., 2022. Research on the lightweight deployment method of integration of training and inference in artificial intelligence. Applied Sciences. 12(13), 6616. DOI: https://doi.org/10.3390/app12136616

[79] Wang, L., Ma, Y., Yan, J., et al., 2018. pipsCloud: High performance cloud computing for remote sensing big data management and processing. Future Generation Computer Systems. 78, 353–368. DOI: https://doi.org/10.1016/j.future.2016.06.009

[80] Qiu, Y., Niu, J., Zhu, X., et al., 2022. Mobile edge computing in space-air-ground integrated networks: Architectures, key technologies and challenges. Journal of Sensor and Actuator Networks. 11(4), 57. DOI: https://doi.org/10.3390/jsan11040057

[81] Capra, M., Peloso, R., Masera, G., et al., 2019. Edge computing: A survey on the hardware requirements in the internet of things world. Future Internet. 11(4), 100. DOI: https://doi.org/10.3390/fi11040100

[82] Malviya, R.K., Vankayalapati, R.K., Sondinti, L.R.K., et al., 2023. Optimizing Real-Time Data Processing: Edge and Cloud Computing Integration for Low-Latency Applications in Smart Cities. Letters in High Energy Physics. 2023, 170–183. Available from: https://www.researchgate.net/publication/385977636_Optimizing_Real-Time_Data_Processing_Edge_and_Cloud_Computing_Integration_for_Low-Latency_Applications_in_Smart_Cities

[83] Adámek, K., Novotný, J., Thiyagalingam, J., et al., 2021. Efficiency near the edge: Increasing the energy efficiency of FFTs on GPUs for real-time edge computing. IEEE Access. 9, 18167–18182. DOI: https://doi.org/10.1109/ACCESS.2021.3053409

[84] Lou, H., 2024. Comparative Analysis and Enhancement of FPGA and ASIC Performance in Targeted Applications. In: Jain, L.C., Kountcheva, R., Nakamatsu, K., et al. (Eds.). Proceedings of the Second International Conference on Artificial Intelligence and Communication Technologies (ICAICT 2024). Springer: Singapore. pp. 295–308. DOI: https://doi.org/10.1007/978-981-96-0096-0_23

[85] Choudhary, S., Vijitha, S., Bhavani, D.D., et al., 2025. Edge AI Deploying Artificial Intelligence Models on Edge Devices for Real-Time Analytics. ITM Web of Conferences. 76, 01009. DOI: https://doi.org/10.1051/itmconf/20257601009

[86] Choudhary, T., Mishra, V., Goswami, A., et al., 2020. A comprehensive survey on model compression and acceleration. Artificial Intelligence Review. 53(7), 5113–5155. DOI: https://doi.org/10.1007/s10462-020-09816-7

[87] Tantalaki, N., Souravlas, S., Roumeliotis, M., 2020. A review on big data real-time stream processing and its scheduling techniques. International Journal of Parallel, Emergent and Distributed Systems. 35(5), 571–601. DOI: https://doi.org/10.1080/17445760.2019.1585848

[88] Roosta, S.H., 2012. Parallel Processing and Parallel Algorithms: Theory and Computation. Springer: New York, NY, USA.

[89] Neeli, S.S.S., 2020. Real-time data management with in-memory databases: A performance-centric approach. Journal of Advances in Developmental Research. 11(2), 1–8. Available from: https://www.ijaidr.com/papers/2020/2/1174.pdf

[90] Mustafa, A., Hameed, N., 2023. Advanced computing techniques for real-time data processing and high-performance computing. Journal of Advanced Computing Systems. 3(8), 1–8.

[91] Rapp, L., Santos, C., 2019. Satellite imagery, very high-resolution and processing-intensive image analysis: Potential risks under the GDPR. Air and Space Law. 44(3), 275–295. DOI: https://doi.org/10.54648/AILA2019018

[92] Vetter, J.S., Reed, D.A., 2000. Real-time performance monitoring, adaptive control, and interactive steering of computational grids. The International Journal of High Performance Computing Applications. 14(4), 357–366. DOI: https://doi.org/10.1177/109434200001400407

[93] Shahriar, M.T., Li, H., 2020. A study of image pre-processing for faster object recognition. arXiv preprint. arXiv:2011.06928. DOI: https://doi.org/10.48550/arXiv.2011.06928

[94] Mishra, P., Biancolillo, A., Roger, J.M., et al., 2020. New data preprocessing trends based on ensemble of multiple preprocessing techniques. TrAC Trends in Analytical Chemistry. 132, 116045. DOI: https://doi.org/10.1016/j.trac.2020.116045

[95] Sivathapandi, M.M.P., Krishnaswamy, P., 2022. Advanced AI algorithms for automating data preprocessing in healthcare: Optimizing data quality and reducing processing time. Journal of Science and Technology. 3(4), 126–167. Available from: https://philarchive.org/rec/PRAAAA-10

[96] Sumon, M.F.I., Khan, M.A., Rahman, A., 2023. Machine learning for real-time disaster response and recovery in the U.S. International Journal of Machine Learning Research in Cybersecurity and Artificial Intelligence. 14(1), 700–723.

[97] Manavalan, R., 2017. SAR image analysis techniques for flood area mapping—Literature survey. Earth Science Informatics. 10(1), 1–14. DOI: https://doi.org/10.1007/s12145-016-0274-2

[98] Zhang, L., Zhang, L., 2022. Artificial intelligence for remote sensing data analysis: A review of challenges and opportunities. IEEE Geoscience and Remote Sensing Magazine. 10(2), 270–294. DOI: https:

//doi.org/10.1109/MGRS.2022.3145854

[99] Sishodia, R.P., Ray, R.L., Singh, S.K., 2020. Applications of remote sensing in precision agriculture: A review. Remote Sensing. 12(19), 3136. DOI: https://doi.org/10.3390/rs12193136

[100] Sabir, R.M., Mehmood, K., Sarwar, A., et al., 2024. Remote Sensing and Precision Agriculture: A Sustainable Future. In: Kanga, S., Singh, S.K., Shevkani, K., et al. (Eds.). Transforming Agricultural Management for a Sustainable Future. Springer: Cham, Switzerland. pp. 75–103. DOI: https://doi.org/10.1007/978-3-031-63430-7_4

[101] Žalohar, J., 2025. Remote Sensing in Conservation and Farming: From Agricultural Fields to High Mountains. Available from: https://www.researchgate.net/publication/397795990_Remote_Sensing_in_Conservation_and_Farming (cited 5 April 2026).

[102] Ogunwumi, O.T., Matindike, R., 2025. AI Applications in Environmental Monitoring: Case Studies on Deforestation and Pollution Assessment. In: Garwi, J., Masengu, R. (Eds.). AI and Ecological Change for Sustainable Development. IGI Global Scientific Publishing: Hershey, PA, USA. pp. 65–100. DOI: https://doi.org/10.4018/979-8-3373-0680-3.ch003

[103] Vishnubhatla, S., 2020. Adaptive Real-Time Decision Systems: Bridging Complex Event Processing and Artificial Intelligence. International Journal of Science, Engineering and Technology. 8(2). Available from: https://www.researchgate.net/publication/397141200_Adaptive_Real-Time_Decision_Systems_Bridging_Complex_Event_Processing_and_Artificial_Intelligence

[104] Reksten, J.H., Salberg, A.-B., 2021. Estimating traffic in urban areas from very-high resolution aerial images. International Journal of Remote Sensing. 42(3), 865–883. DOI: https://doi.org/10.1080/01431161.2020.1815891

[105] Butilă, E.V., Boboc, R.G., 2022. Urban traffic monitoring and analysis using unmanned aerial vehicles (UAVs): A systematic literature review. Remote Sensing. 14(3), 620. DOI: https://doi.org/10.3390/rs14030620

[106] Silva, B.N., Khan, M., Jung, C., et al., 2018. Urban planning and smart city decision management empowered by real-time data processing using big data analytics. Sensors. 18(9), 2994. DOI: https://doi.org/10.3390/s18092994

[107] Trigka, M., Dritsas, E., 2025. Edge and cloud computing in smart cities. Future Internet. 17(3), 118. DOI: https://doi.org/10.3390/fi17030118

[108] Zhang, Z., Zhu, L., 2023. A review on unmanned aerial vehicle remote sensing: Platforms, sensors, data processing methods, and applications. Drones. 7(6), 398. DOI: https://doi.org/10.3390/drones7060398

[109] Wang, G., Wan, G., Su, Z., et al., 2025. High-performance on-orbit intelligent computing and real-time services for remote sensing satellites based on large-scale computing power in space. IEEE Access. 13, 92114–92133. DOI: https://doi.org/10.1109/ACCESS.2025.3573932

[110] Chatar, K., Kitamura, K., Cho, M., 2024. Onboard data prioritization using multi-class image segmentation for nanosatellites. Remote Sensing. 16(10), 1729. DOI: https://doi.org/10.3390/rs16101729

[111] Nandkishor, P.O., Chandra, B., 2025. Artificial Intelligence in Real-Time Market Decision-Making: A Comparative Study of AI-Driven vs. Traditional Approaches. Journal of Integrated Engineering Sciences. 1(4), 60–70. DOI: https://doi.org/10.63686/jies.v1i4020

[112] Janga, B., Asamani, G.P., Sun, Z., et al., 2023. A review of practical AI for remote sensing in earth sciences. Remote Sensing. 15(16), 4112. DOI: https://doi.org/10.3390/rs15164112

[113] Seddik, E., 2021. Advanced pre-processing techniques for cloud-based degradation detection using artificial intelligence (AI) [PhD Thesis]. McMaster University: Hamilton, ON, Canada.

[114] Bala, B., Behal, S., 2024. A brief survey of data preprocessing in machine learning and deep learning techniques. In Proceedings of the 8th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Kirtipur, Nepal, 3–5 October 2024; pp. 1755–1762. DOI: https://doi.org/10.1109/I-SMAC61858.2024.10714767

[115] Khan, M.N., Suhail, M., Batyrova, O., et al., 2025. Balancing Innovation and Accountability: Threats and Challenges of AI in Remote Sensing Applications. In: Bansal, J.C., Jamwal, P., Hussain, S. (Eds.). Proceedings of the International Conference on AI and Robotics. Springer: Cham, Switzerland. pp. 259–274. DOI: https://doi.org/10.1007/978-3-032-05548-4_21