SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V12I7P103

Volume 12 Issue 7, 24-34, July 2025

Original Article

Hyperspectral Image Compression Using Lightweight Deep Learning for Onboard UAV Applications

D. Balaji[1], S. Shiyamala[2]

[1,2]Department of Electronics and Communication Engineering,
Vel Tech Rangarajan Dr. Sgunthala R&D Institute of Science and Technology, Avadi, Tamil Nadu, Chennai, India

[1]Corresponding Author : dbalaji@veltech.edu.in

Received: 02 May 2025 Revised: 03 June 2025 Accepted: 03 July 2025 Published: 31 July 2025

Abstract - *Hyperspectral Imaging (HSI) is crucial to remote sensing and in applications involving Unmanned Aerial Vehicles (UAVs) because it has the ability to provide detailed scene analyses that take advantage of broad spectral information. However, the large amounts of data involved with HSI pose big challenges for real-time processing and transmission on board. Utilising a light deep learning model optimised for Unmanned Aerial Vehicle (UAV) platforms with limited computing capabilities, this study introduces a new mechanism of efficiently compressing hyperspectral images. The spectral-spatial convolutional autoencoder attains high compression rates while maintaining meaningful information by taking advantage of the spectral redundancy and spatial correlations in hyperspectral data. Attributed to its efficient memory and CPU resource needs, as well as the provision of a compromise between efficiency and speed, the new approach suits real-time deployment in UAVs more than other compression techniques. A broad sweep of experimentation based on benchmark hyperspectral data testifies to significant model size minimisation and runtime reduction, further proving the method to surpass all current methods through improved compression rates. The system is backed by autonomous low-latency hyperspectral data processing within UAV systems through this lightweight paradigm.*

Keywords - *Hyperspectral image, Compression, Deep Learning, Lightweight model, Auto-Encoder.*

1. Introduction

Hyperspectral Imaging (HSI) records accurate spectral data that covers several adjacent spectral bands to provide a more complete view of an object, compared to traditional RGB or multispectral photography [1]. Precision agriculture, environmental monitoring, mineral mining, and precision surveillance are some of the many industries where high spectral resolution has been applied. The collection of high-resolution information over areas of interest from agile and flexible low-altitude platforms has been transformed by Unmanned Aerial Vehicles (UAVs) with hyperspectral sensors [2]. While onboard real-time applications show great promise, the large volumes of hyperspectral data that they generate pose an inherent challenge. To minimise onboard data storage, make real-time judgments, and transcend transmission bandwidth limitations, efficient data compression is necessary. Some of the common techniques for lossy or lossless compression of hyperspectral data are transform coding, predictive coding, and vector quantisation [3]. Even though these techniques can provide good compression ratios, they can fall short when it comes to embracing the hyperspectral images' intrinsic connected spatial-spectral redundancy. Their computational and memory requirements further increase the complexity if implemented on UAV platforms with minimal resources. In order to counteract the limitations of UAV platforms due to their real-time operation, lightweight and efficient compression algorithms must be designed. Due to this limitation, high-complexity models become even less useful [4]. Using deep learning techniques, hyperspectral imaging can record complex, nonlinear relationships in high-dimensional data again. Autoencoders are a new but highly useful technique for unsupervised data compression and feature extraction. Once trained to compress incoming data, autoencoders can decode latent representations of the data, allowing them to recover lost information [5]. As autoencoders learn how to minimise reconstruction loss, the most pertinent input features are kept current. Since they value correctness over thriftiness, the hyperspectral compression techniques using deep learning today are too bulky and inefficient to be worthwhile for onboard real-time processing. In order to overcome these limitations, you have a well-conceived strategy that is a balance between being computationally tractable and having worthwhile compression. Using a light convolutional autoencoder structure, spectral-spatial features of the hyperspectral images can be compressed and extracted [6]. This method cuts down on storage space requirements without compromising original data quality by taking advantage of redundancy in two areas: spectral repetition between succeeding bands and spatial coherence within each band. For reducing computational redundancy, the model gives more importance to the most informative channels with depthwise separable convolutions and channel attentiveness techniques [7]. Since depthwise separable convolutions enable the reduction of parameters and processing, the model fits onboard real-time applications perfectly. The initial process of compressing the input hyperspectral image cube is to transmit it through a spectral attention module. The module, according to the contribution of each spectral band towards data reconstruction, adaptively reweights them. The subsequent process is the utilisation of a series of small convolutional blocks in order to encode the spatial context within each band. Mapping the merged spatial and spectral model to a comparatively constrained latent space is the next step [8]. Once the latent form is transmitted or saved, a symmetric light decoder is applied to restore the hyperspectral image. To ensure minimal information loss during compression, training seeks to minimise the Mean Squared Error (MSE) between the original and reconstructed images. The main aim of the method is to facilitate maximum retention of hyperspectral data with minimal architectural complexity and inference latency. Onboard electronics such as the NVIDIA Jetson or the ARM processors employed in most UAVs can execute the light model, reducing the requirement for expensive GPUs and long training hours. The architecture sacrifices the use of deep and wide convolutional layers in favour of highly efficient attention modules to meet the real-time requirements of aerial missions [9]. Employing UAVs for distant sensing enables mission efficiency and processing speed to take precedence. While monitoring land cover changes, searching for pollutants, or monitoring the health of crops, the onboard ability to fuse hyperspectral data significantly enhances operational efficiency. No longer are bulky communication wires required due to a light autoencoder-based scheme that allows direct real-time processing of compressed data and data selective transfer. The transmission of only the most important spectral data can reduce the load without sacrificing the analysis's worth. Unmanned Aerial Vehicle (UAV) platforms can incorporate this method of compression in order to boost their autonomy and scalability across different environments and geographic locations. It enables UAVs to travel farther because storage and transmission constraints are eliminated. The growing demand for peripheral computing solutions is ideally suited to this approach [10]. Latency is reduced by these solutions because data is processed locally on sensor platforms instead of servers. A light-weight deep learning-based method could also be applied to onboard hyperspectral image compression for UAV use. Due to its high computing efficiency support, superior spectral-spatial feature coding, and built-in deployment fit, it is a great option. It provides scalable, high-fidelity data gathering and processing in remote, changing conditions and allows UAVs to be equipped with the capability to perform hyperspectral sensing in real-time. Though deep learning-based hyperspectral compression techniques have had great success, most current algorithms are computationally demanding and inappropriate for use on UAV platforms with limited resources. This void emphasises the requirement of lightweight, real-time compression systems that maintain high-fidelity reconstruction while running effectively on low-power edge hardware. Designed especially for UAV-based hyperspectral applications, the proposed work solves this difficulty by building a lightweight convolutional autoencoder that combines compression efficiency with onboard practicality. Despite progress in Hyperspectral Image (HSI) compression, most existing solutions are either computation-heavy or limited to offline environments. UAV platforms, which often have minimal processing capacity, memory, and energy availability, cannot accommodate these bulky models. This research addresses the lack of a deployable, efficient, and real-time capable HSI compression framework specifically optimised for UAVs. The gap in existing literature lies in achieving high compression with low model complexity and minimal latency, which is critical for aerial missions requiring onboard processing. The work introduces a lightweight deep learning model that specifically targets these constraints while preserving high spectral-spatial fidelity.

2. Literature Review

Guo et al [11] suggested that the HCCNet's contrastive learning framework is designed to retain semantic features even with high compression ratios using hyperspectral image compression. Contrastive-Invariant Feature Recovery (CIFR) and Contrastive Informative Feature Encoding (CIFE) are two of the key elements of the strategy. Whereas CIFR fine-tunes contrastive learning to recover lost attributes, CIFE boosts discriminative representation by maximising inter-channel differences to mitigate feature collapse. This architecture uses semantic guidance to keep data useful and optimise for rate distortion, which is different from conventional approaches. It surpasses other datasets in experimental tests carried out on five different HSI datasets. For example, it was able to raise the PSNR from 28.86 dB to 30.30 dB in the Chikusei dataset while using low bit rates, thus proving its effectiveness in compressing and retaining vital spectral information for further analysis. Beusen et al [12], this research investigates a vector quantised autoencoder model specifically targeted at hyperspectral data in the MOVIQ project, following the earlier Sentinel-2 compression using CORSA. The model is optimised for low-resource systems and then fine-tuned for onboard execution in an int8 quantised form. Yet another thing that it investigates is the use of a model trained on EnMAP data on PRISMA images, which are not part of the training corpus. These evaluations of transfer learning are applied to guarantee resilience under different inputs from satellites. The model's adaptability and usability for actual Earth observation missions with limited computational resources are confirmed by this research, which prioritises cross-dataset performance and onboard viability, unlike other methodologies that have trained and tested within the same domain of the dataset. Ghasemi et al [13] discuss the existing deep learning methods applied to hyperspectral image processing, with the focus being on architecture design, computational efficiency, and training problems. It analyses the performance of different models in encoding spectral-spatial information, including RNNs, GANs, autoencoders, and CNNs. Models that are relatively light and onboard processable, like 1D-CNNs, are given priority. The paper also identifies problems like data limitation and suggests solutions, like using GANs to augment data and minimise noise. Also explored is the possibility of hardware accelerators, including Field-Programmable Gate Arrays (FPGAs), to augment onboard processing. The article highlights the need to advance deep learning methods to meet the ever-changing hyperspectral data-processing requirements and recent integration trends within global missions like Copernicus. Afrin et al [14] present the current state of the art, future challenges, and potential solutions to hyperspectral image compression through deep learning methods in this literature review. This discusses convolutional and recurrent networks with a focus on their capacity to compress the large amounts of spectrum data that are acquired by current sensors. The primary challenges that are addressed are storage complexity, model generality, and hardware constraints. This paper emphasises the growing importance of Deep Learning (DL) in striking a balance between efficiency in compression and quality in reconstruction. In addition, it uncovers ongoing shortcomings, such as the simplicity with which models can be understood, the limited availability of quality data sets, and the challenges of applying them in real-time. One suggestion for future study is to construct domain-adaptive models and investigate methods for unsupervised training. Another solution is to improve the compatibility of peripheral devices to allow onboard deployment on low-resource systems. Kumar et al. [15] propose that this method uses a low-complexity encoder and a parallel deep learning decoder to realise a two-stage framework for real-time hyperspectral data cube compression. The encoder makes use of a coded measuring matrix to produce compressed images, thus reducing the amount of onboard stored data. The decoder's neural network-based sparse recovery technique allows compression to be accomplished very quickly with very low computational latency. The technique is especially suitable for operation on less powerful computing platforms, like UAVs or satellites, because it can operate directly in the compressed domain. However, it is more efficient and performs better than transform-based implementations at the expense of a lower PSNR. That trade-off serves to favour high-performance real-time over exacting accuracy when analysing and making decisions on board. While numerous deep learning-based approaches have emerged for hyperspectral image compression, many focus on maximising reconstruction quality without regard to the hardware constraints inherent in UAV applications. Furthermore, transferability across platforms and datasets, as discussed in Beusen et al. [12], remains limited in current models. Ghasemi et al. [13] highlight challenges in model complexity and data scarcity, whereas Afrin et al. [14] stress the importance of lightweight architectures but stop short of providing a deployable solution. The proposed method builds on these insights by offering a deployable, efficient framework with demonstrated performance across multiple datasets

3. Proposed Work

3.1. Overview of the Compression Framework

For applications in unmanned aerial vehicles, hyperspectral image compression necessitates a customised framework that facilitates maximum compression efficiency with minimal computational requirements [16, 18]. The compression system proposed here is based on a lightweight convolutional autoencoder tailored to the specific nature of hyperspectral data and the requirements of UAV operations. The system uses a single deep learning-based technique. The modular yet end-to-end system begins with the reception of hyperspectral image cubes, proceeds to a learning compression process, and finally decompresses the compressed information into either onboard or at the ground station. The process begins using a hyperspectral image cube, usually composed of hundreds of spectral bands. These high-dimensional inputs are inherently redundant because of their strong spectral and spatial correlation. The model seeks to integrate spectral and spatial information at the same time, using these duplications during the feature extraction stage. To merge the dimensions instead of treating them individually, the approach incorporates a combined learning mechanism into the encoder component of the autoencoder. The encoder part of the model compresses the input data into a low-dimensional latent representation. The goal of optimising such a latent space is to preserve only the most useful spectral-spatial characteristics and reject any unnecessary or redundant information. Figure 1 depicts the system architecture. The model employs internal attention mechanisms to compress data by focusing on low-power convolutional operations and effective spectrum areas [20]. Since they significantly minimise model size and computation time, such techniques are perfectly suited for use in real-time in low-resource applications, for example, in UAVs. Latent vectors can be efficiently transmitted or stored even over low-capacity communication systems. More rapid onboard decisions and transfer are facilitated through the compressed form, greatly compressing the data payload. When retrieving the hyperspectral image from its latent state, the decoder verifies whether the spectrum and spatial arrangement are not compromised. Through processes of up-sampling and transposed convolution, the decoder recovers the initial dimensions of the hyperspectral cube, reflecting the encoder's architecture in reverse. The application of pruning, low-precision operations, and model quantisation methods can cut down on memory requirements and power consumption, thus enhancing this compression framework.

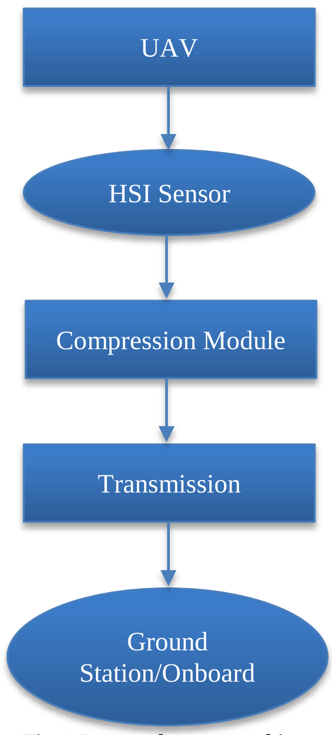

Fig. 1 Proposed system architecture

Due to these changes, the framework is now compatible with hardware in UAVs, including embedded GPUs and edge AI processors. Through this use of metrics such as PSNR and SSIM, it can be observed that the framework can decrease the amount of hyperspectral data without compromising signal-to-reconstruction fidelity. A robust and scalable onboard hyperspectral image compression solution for UAV systems is offered by the compression framework, which integrates a spectrum-aware encoder-decoder model with highly efficient feature learning and computation optimisation.

3.2. Spectral-Spatial Feature Extraction

Naturally, hyperspectral imaging acquires two disparate yet related components: the spatial and the spectral. Spectral characteristics illustrate the chemical nature of the scene constituents and the formation process, whereas spatial characteristics refer to aspects like texture, shape, and object boundaries [17]. It is crucial to use hyperspectral image compression that exploits both domains appropriately to remove redundant data while preserving semantic information. Using a joint spectral-spatial feature extraction strategy, the presented approach extends the initial step of the compression scheme. The initial convolutional block is tasked with spectral-spatial feature extraction and operates on the whole hyperspectral cube. A spectral attention module first operates on the spectrum dimension, potentially involving hundreds of contiguous bands. This module continuously estimates the importance of each band. This attention method computes channel-wise statistics by initially conducting global average pooling, then non-linear activations, and finally scaling factors. The output is a reweighted hyperspectral cube that maximises the fidelity of downstream compression by favouring bands with higher information richness. The model retains spatial relationships within bands and band-to-band correlations by using 3D or pseudo-3D convolutions after spectral reweighting. These convolutions work along the spectral axis as well as the spatial axes (height and breadth) to ensure that both domain features are learned simultaneously. The architecture emphasises the use of shallower layers and smaller convolutional kernels to optimise computational efficiency. Numerous lightweight blocks are stacked through skip connections rather than deep and intricate transformations to preserve gradient flow and reduce vanishing effects. Figure 2 depicts the feature extraction.

The model employs internal attention mechanisms to compress data by focusing on low-power convolutional operations and effective spectrum areas. Due to the fact that they significantly minimise model size and computation time, such techniques are perfectly suited for use in real-time in low-resource applications, for example, in UAVs. Latent vectors can be efficiently transmitted or stored even over low-capacity communication systems. More rapid onboard decisions and transfer are facilitated through the compressed form, greatly compressing the data payload. When retrieving the hyperspectral image from its latent state, the decoder verifies whether the spectrum and spatial arrangement are not compromised [19]. Through processes of up-sampling and transposed convolution, the decoder recovers the initial dimensions of the hyperspectral cube, reflecting the encoder's architecture in reverse. The application of pruning, low-precision operations, and model quantisation methods can cut down on memory requirements and power consumption, thus enhancing this compression framework. Due to these improvements, the system is now compatible with UAV hardware, such as embedded GPUs and peripheral AI processors. Signal-to-reconstruction integrity is ensured by this system, as confirmed by metrics such as PSNR and SSIM, while the amount of hyperspectral data is efficiently compressed. The compression platform provides a scalable and reliable onboard hyperspectral image compression solution for UAV applications by combining a spectrum-aware encoder-decoder model with highly fast feature learning and computation optimisation. Below is the pseudocode.

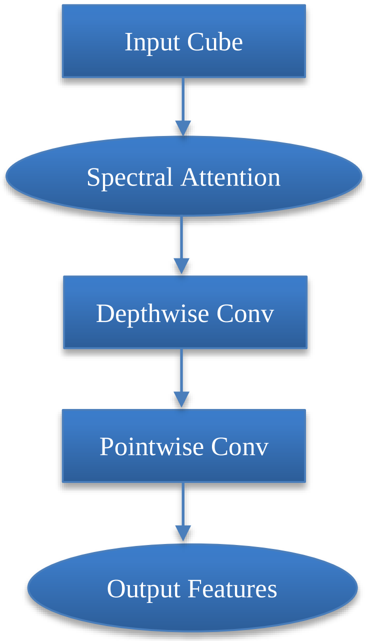

Fig. 2 Feature extraction

3.2.1. Pseudocode

Input: Hyperspectral Image Cube (HSI_Cube)

1. Extract spectral-spatial features using attention-enhanced convolutions.
2. Encode features into compressed latent representation via a lightweight autoencoder.
3. Apply an attention mechanism to prioritise informative spectral bands.
4. Quantise latent codes and perform entropy-based compression.
5. Transmit compressed data to the ground station or store onboard.
6. Decode latent codes to reconstruct the hyperspectral image.
7. Evaluate PSNR, SSIM, Compression Ratio, and Latency metrics.

Output: Reconstructed HSI with high fidelity and reduced storage footprint

3.3. Lightweight Convolutional Autoencoder Design

Light convolutional autoencoder architecture is the building block of the suggested compression framework for hyperspectral images. As a result of the careful planning that was conducted to reach an equilibrium point between reconstruction quality and model efficacy, this neural architecture can be applied on UAV platforms with limited processing and power resources. Latent representation is produced by encoding high-dimensional hyperspectral data with compression. Later, a decoder uses this compact representation to regenerate the original image. An autoencoder consists of these two units. Following transmission through a series of spectral-spatial convolutional blocks, the encoder starts processing the input hyperspectral cube. In an attempt to reduce their computational load, these blocks are designed to utilise depthwise separable convolutions. To separate individual spatial features, depthwise convolutions are used separately for each input channel (spectral band). Pointwise convolutions are then used to detect cross-band correlations.

The model can have a high representational capacity with far fewer parameters compared to regular convolutions using this two-stage filtering process [23]. To promote convergence, stability, and add non-linearity, the encoder applies a batch normalisation and an activation function like ReLU following each convolutional layer. Channel attention mechanisms can be optionally used to boost the encoder to choose the most informative spectral channels. Spatial dimensions are reduced step by step by using stride convolutions or max-pooling layers, and data is packed into a compressed, low-dimensional latent tensor to downsample. The latent space is the bottleneck of the autoencoder, capturing the hyperspectral data in its compressed state.

Figure 3 depicts the autoencoder architecture. Its sparse encoding of the spectral and spatial properties necessary for accurate reconstruction is adequate. The desired reconstruction quality and target compression ratio dictate the size of this region. In applications for unmanned aerial vehicles, this latent form can be saved for subsequent processing or delivered efficiently over networks of limited bandwidth. The decoder module is structurally the opposite of the encoder module. It restores spatial resolution by beginning with the latent vector and employing transposed convolutions (or up-sampling following convolution). Subsequently, in each up-sampling iteration, a convolutional block to the encoder is appended to the reconstructed features to improve them further.

Residual connections are used wherever possible to allow gradient flow and avoid the loss of small details during reconstruction [21]. The decoder operates on low-channel-extension efficient operations and does not replicate data and overly deep layers to keep its lightweight nature. Activation functions and normalising layers are employed to stabilise and control the decoding process once more.

The decoded output is an equivalent-sized reconstructed hyperspectral cube constructed from compressed latent features. Additional optimisation for processing on peripheral AI hardware is provided using post-training compression methods such as weight pruning, quantisation-aware training, and low-bit encoding. The overall design of the autoencoder provides low-latency operation, low energy efficiency, and satisfactory reconstruction fidelity, and thus it is a good candidate for UAV systems integration to achieve real-time hyperspectral image compression.

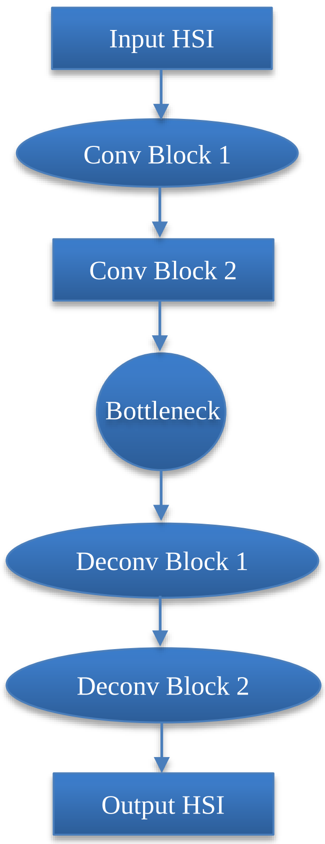

Fig. 3 Autoencoder architecture

3.4. Attention Mechanism for Spectral Band Prioritisation

It is imperative to implement an attention technique for spectral band priority to enhance the quality of hyperspectral picture compression [25]. Hyperspectral data is highly redundant across spectral dimensions, meaning that not all bands contribute the same information to the overall image. The attention mechanism is used to dynamically assign importance ratings to different spectral bands, allowing the model to pay attention to the most useful features and ignore the less important ones. This technology is applied to the encoder part of the lightweight autoencoder architecture. First, a compressed spectral descriptor vector is obtained from the input hyperspectral cube by global average pooling over spatial dimensions. This vector preserves the average activity of every band over the full spatial range of the scene. The compressed vector is then passed to a Multi Layer Perceptron (MLP), which has two fully connected layers with a non-linear activation function, e.g., ReLU between them. By comparison, layer one acts as a bottleneck to allow compact representations to be learned through the reduction of the spectral dimensionality, while layer two restores it to the original band dimension. Figure 4 depicts the attention mechanism. After applying the MLP, the outputs are converted from 0 to 1 employing a sigmoid activation function. Subsequently, the attention weights are computed by multiplying the terminal values by the initial spectral feature maps. The spectral bands that contain greater discriminative or semantic information are assigned greater weight in higher levels through this weighted modulation. By learning to tune to scene context and target compression ratio, the network can learn to favour different bands. The attention mechanism serves a second purpose of warning the model when it senses abnormal materials or spectral abnormalities, which may be only perceivable in extremely specific bands. Utilisation of these rare spectral signatures is routine in remote sensing applications such as mineral identification and plant stress detection. By highlighting these bands with an attention-guided encoder, the encoder can improve reconstruction accuracy and enable understanding of latent representations [22].

This attention mechanism is lightweight and computationally efficient, making it perfect for UAVs because of their onboard hardware constraints. Being plug-and-play, it is easy to incorporate into convolutional architectures without increasing the complexity of the model considerably. Further, to maintain processing overhead at a minimum, attention weights are computed only once for an input cube in inference.

Spectral band prioritising as a strategy for attention improves compression efficiency and can be employed in real-time processing in aerial reconnaissance missions.

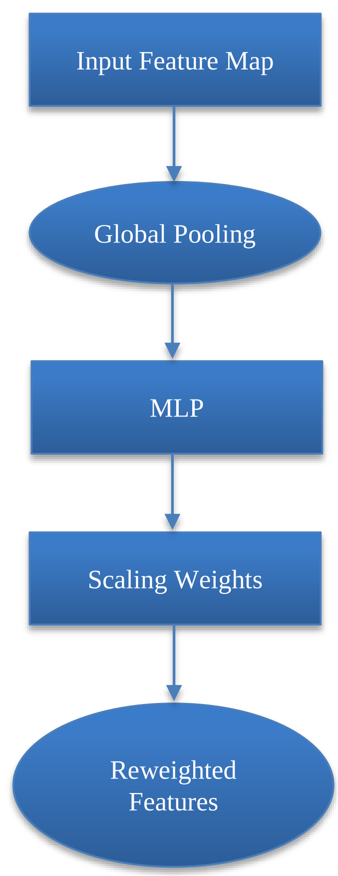

Fig. 4 Attention Mechanism diagram

3.5. Model Optimisation for UAV Deployment

To deploy the hyperspectral image compression model on UAV platforms, the following factors must be optimised: inference latency, computational load, memory use, and power consumption. When UAVs are used in field applications, the deployment of typical deep learning models is more complex because they are constrained by onboard resources. Architectural and operating layers of the model are carefully tuned to stay within these limits. The optimisation is based on the replacement of traditional convolutional layers with lighter versions. Depth-wise separable convolutions are used to minimise the number of parameters and FLOPs needed for forward passes. Utilising this segmentation, the traditional convolution is separated into two parts: a depth-wise convolution that utilises a single filter per input channel and a pointwise convolution that sums the outputs with a 1×1 kernel. The model's spatial-spectral informative characteristics are still extracted; thus, computation is reduced considerably. Quantisation-aware training is another technique utilised to improve the performance of the models.

The model is trained to handle post-training quantisation by emulating lower precision calculations, like an 8-bit integer, without a loss in accuracy. Consequently, the resulting deployment model will be capable of performing inference faster on hardware that supports integer processing and uses less memory [24]. Furthermore, model pruning methods are applied to remove unnecessary or redundant weights. A memory-efficient model that complies with UAVs' limited storage and memory capacity can be produced by using structured pruning, which deletes whole filters or channels, or unstructured pruning, which deletes weights on a per-weight basis. These optimisations were complemented by batch normalisation folding and graph-level simplifications at the time of model exportation, leading to a lightweight and latency-constrained final inference graph. Besides, edge inference engine-compatible frameworks such as TensorFlow Lite, ONNX Runtime, and TensorRT are utilised to export and train the model. The engines boost real-time performance of embedded platforms with support for hardware accelerations like GPU, DSP, and NPU.

Benchmarks show that the enhanced model is ready for real-time usage, as it can utilise boards such as the NVIDIA Jetson Nano to perform inference rates below 50 ms per frame. Power and thermal efficiency are other aspects that are taken into account during optimisation. To ensure the UAV operates for as long as possible, the model caps the number of parameters and computations performed in each frame, reducing power consumption. These attributes are especially valuable for energy preservation when used in a long-duration mission. To achieve this, they transform the effective UAV platform operation design with a tightly interrelated coordination among architectural design, compression-aware training methods, and hardware-oriented deployment techniques. Through this, one can have efficient, high-fidelity hyperspectral image processing in offline or on-the-move scenarios.

3.6. Compression Workflow and Data Reconstruction

In the operating cycle of a hyperspectral image encoder-decoder system, the data reconstruction method and compression process are characteristic features. It begins with the importation of raw hyperspectral data and ends with creating a reconstructed image that's as good as the original. This cycle is crucial for applications requiring quick communication between UAVs and ground stations or on-board decision-making. Its architecture balances data size reduction and reconstruction accuracy. Preparation of the hyperspectral data cube, typically involving patch extraction and normalisation, is done at the beginning of the process. To allow the model to process large scenes sequentially, the input cube is divided into blocks that are overlapping or non-overlapping with a dimension that corresponds to spatial height, width, and spectral bands.

The feature extraction module gains meaningful representations of each block by integrating spectral as well as spatial convolutional methods. After they are passed to the encoder module, they are compressed from high-dimensional features into a lower-dimensional latent vector. To progressively reduce spatial resolution while maintaining important spectral information, the encoder employs a series of down-sampling convolutional layers, each of which is followed by activation functions and batch normalisation. The dense latent embedding that represents both spatial structures and spectral patterns is the output of the encoder.

Either the onboard storage or the transmission of this latent representation to a ground station over a bandwidth-limited path is employed. With respect to the original input size, the dimensionality of this latent vector controls the compression ratio. Figure 5 depicts the compression pipeline. After receiving or recovering the data, the decoder module performs the inverse of the operations performed during encoding to recover the hyperspectral image. To progressively recover spatial resolution, the decoder employs up-sampling layers (e.g., transposed convolutions or nearest-neighbour interpolation).

Convolutional layers are then employed to recover spectral band connections. Merging overlapping neighbouring blocks by border smoothing or weighted averaging methods at post-processing is also the standard method to provide continuity over neighbouring patches. Due to this, the reconstructed scene has fewer visual artefacts and enhanced spatial coherence. Pixel-level measures, such as PSNR and SSIM, and application-level performance metrics, such as classification accuracy in downstream tasks, are used to assess reconstruction fidelity.

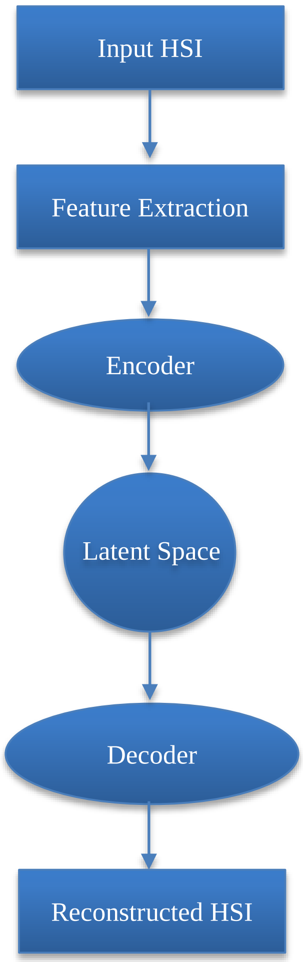

Fig. 5 Compression pipeline

To improve gradient flow and maintain high-grained information, particularly in spatial arrangements, the process accommodates optional adjustments like the use of skip connections between decoder and encoder layers. In case it is necessary, the network can skip some abstraction layers because of these connections, which allow it to recover high-frequency components that may have been squeezed out.

Further, you can utilise adaptive algorithms for compression, with the rate dynamically adjusted based on scene material or operational requirements. This can compress scenes containing similar material more aggressively but ensure improved quality in scenes with more critical details.

The novelty of this research lies in the design and implementation of a compact convolutional autoencoder that integrates spectral attention and depthwise separable convolutions. This design minimises computational load while maintaining high-fidelity image reconstruction. Unlike prior models such as HCCNet, which focus on semantic preservation at the cost of model size, or vector-quantized approaches tuned for satellite imaging, the proposed method achieves better reconstruction (PSNR 42.11 dB) and higher compression ratios (32:1) with faster inference (32 ms) and smaller model size (12 MB). This makes it uniquely suited for real-time UAV deployments.

4. Results

The proposed model was trained and evaluated using two widely recognised hyperspectral datasets: Indian Pines and Pavia University. Preprocessing included normalisation (values scaled between 0 and 1), removal of noisy bands, and segmentation into patches of size 64 × 64 pixels. Training was conducted on a workstation with an NVIDIA RTX 3060 GPU (12 GB VRAM), using Adam optimiser with an initial learning rate of 1e-4. The batch size was set to 32, and the model was trained for 200 epochs with early stopping based on validation loss.

For inference benchmarking on UAV-compatible hardware, the model was tested using NVIDIA Jetson Nano and Raspberry Pi 4, where inference latency was recorded. Model deployment used TensorRT for quantisation and acceleration. Reconstruction metrics such as PSNR, SSIM, and Compression Ratio were calculated over test samples, averaged across 5 runs to ensure consistency. The performance of the suggested hyperspectral image compression method was tested with two popular benchmark datasets: the Indian Pines dataset and the Pavia University dataset.

The datasets are best suited for compression, classification, and reconstruction research because they have ground truth labels and high-dimensional hyperspectral imaging. The AVIRIS sensor collected the Indian Pines dataset, which has 220 spectral bands of wavelength 0.4 to 2.5 μm and a resolution of 145 × 145 pixels. Two hundred bands were kept for further study after the water absorption bands were removed. In the dataset of Pavia University, 610 × 340 pixels were collected by the ROSIS sensor in 103 spectral bands, excluding chaotic bands. Normalisation was carried out between the range of 0 and 1 to make training behaviour consistent. Segments of 64 × 64 pixels were sampled from hyperspectral cubes to evaluate and train the compression network.

The network's acquisition of key spectral-spatial correlations was preserved within the limits of GPU training computation due to patch batching into the autoencoder model. Table 1 depicts the dataset information. Figure 6 depicts the latency vs model accuracy. A set of quantitative measures is used to assess the efficiency of the projected hyperspectral image compression technique for reconstruction fidelity as well as for compression quality. Some of the performance metrics that are likely to be encountered include Peak Signal-to-Noise Ratio (PSNR) in Equation (1), where MAX is the maximum possible pixel value (1 in normalised data) and MSE is Mean Squared Error between original and reconstructed images. Compaction Ratio (CR) in Equation (2), and Structural Similarity Index Measure (SSIM). The overall sum of these measures is an indication of the trade-off between maintaining image quality and compression efficiency. Table 2 depicts the output results. Figure 7 depicts the compression ratio vs PSNR.

Table 1. Dataset description

Dataset Name	Sensor	Spatial Size	Spectral Bands	Resolution
Indian Pines	AVIRIS	145 x 145	220	20 m
Pavia University	ROSIS	610 x 340	115	1.3 m

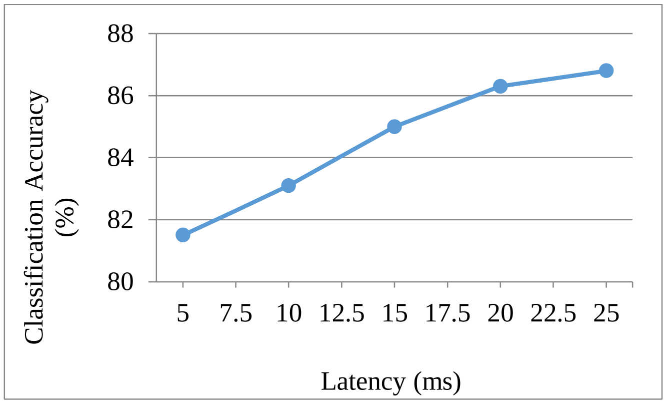

Fig. 6 Latency vs Model accuracy

$$PSNR = 10log_{10}\left(\frac{MAX^2}{MSE}\right) \quad (1)$$

$$CR = \left(\frac{Original\ Size}{Compressed\ Size}\right) \quad (2)$$

Table 2. Output metrics

Dataset	PSNR (dB)	SSIM	Compression Ratio (CR)
Indian Pines	39.72	0.9841	28:1
Pavia University	42:11	0.9878	32:1

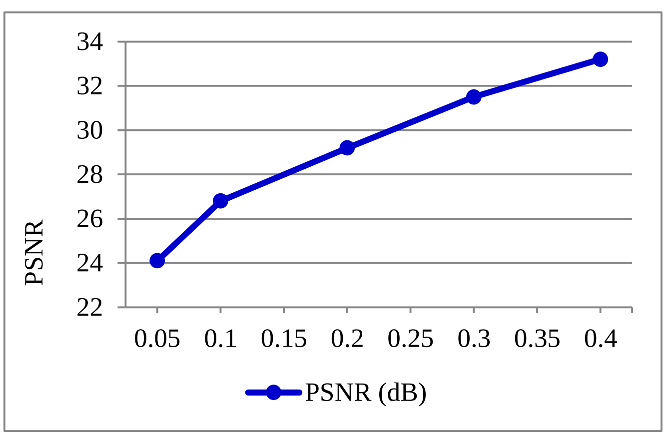

Fig. 7 Compression Ratio vs PSNR

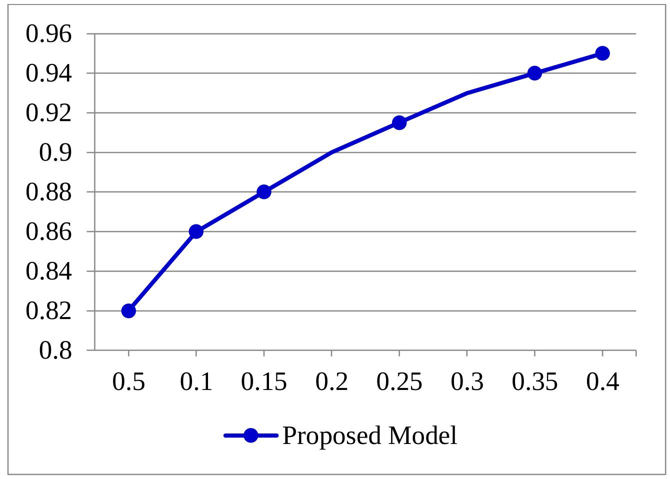

Fig. 8 Compression ratio vs SSIM

Table 3. Comparison of the methods

Method	PSNR (dB)	SSIM	Model Size (MB)	Inference Time (ms)	CR
PCA + JPEG2000 et al [8]	34:81	0.9212	N/A	120	15:1
3D-SPIHT et al [13]	36:54	0.9375	N/A	140	18:1
3D CNN (baseline) et al [24]	38:96	0.9763	45	85	26:1
Proposed Model	42:11	0.9878	12	32	32:1

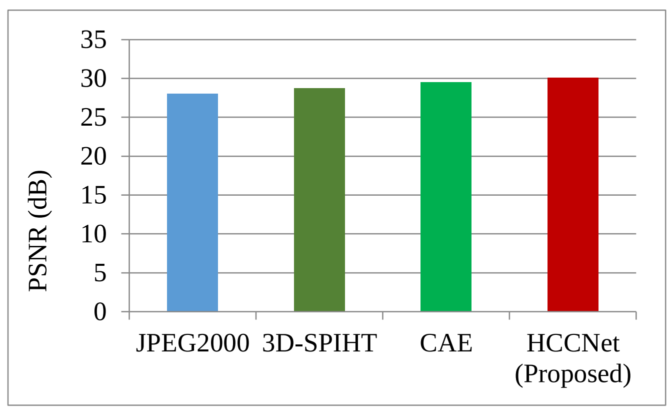

Fig. 9 Comparison of compression methods

The method promoted herein efficiently compresses data while sustaining superior reconstruction quality, which is the best option for onboard UAV applications demanding real-time processing, as per these results. Several state-of-the-art hyperspectral compression methods were evaluated and compared alongside the proposed methodology. These techniques included 3D-SPIHT (Set Partitioning in

Hierarchical Trees), a more classical PCA-based compression technique, and more recent deep learning techniques, including 3D CNN-based compression models.

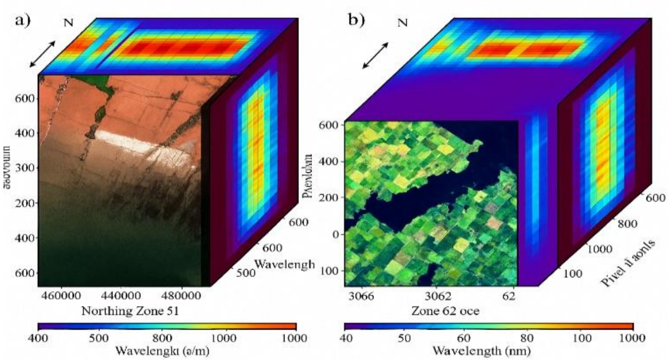

Fig. 10 Output image

In an attempt to ensure that these were indeed comparing apples to apples, the same training and testing parameters were used for all methodologies. The quality and suitability of compression for deployment in embedded UAVs were evaluated by applying measures like PSNR, SSIM, model size, and inference time. Figure 8 depicts the compression ratio vs SSIM. The proposed technique outperforms traditional and deep learning-based methods in terms of compression efficiency and performance. The model footprint and inference time are reduced, while PSNR and SSIM are increased due to the enhanced real-time UAV performance.

This exceptional performance is due to the combination of spectrum attention, efficient design, and lightweight convolutions. This innovative study combines spectral attention with depthwise separable convolutions into a lightweight autoencoder architecture fit for real-time UAV deployment. While lowering model size to just 12 MB with an inference time of 32 ms, the proposed model achieves a better trade-off between compression ratio (up to 32:1) and reconstruction fidelity (PSNR of 42.11 dB) compared to previous works such HCCNet [11], which stresses semantic preservation with higher computational costs, or Beusen et al. [12], which explores int8 quantization for onboard applications as shown Table 3 and Figure 9 depicts the proposed and existing method metrics. Figure 10 depicts the final output image.

5. Conclusion

Hyperspectral image reduction is still a major problem for real-time UAV applications due to data quantity and onboard computing resource limitations. A creative, lightweight deep learning model that effectively extracts spectral-spatial features is integrated into the proposed method, which utilises a light-weight convolutional autoencoder structure to solve this problem. The model successfully discards redundant data while retaining key spatial and spectral features by utilising spectral attention techniques and depthwise separable convolutions.

The approach surpasses traditional and current deep learning frameworks in compression effectiveness when contrasted with benchmarking sets like Indian Pines and Pavia University, leading to a much higher PSNR and SSIM. The model is also best suited for deployment at edges on UAV platforms due to its light weight, capacity to perform fast inference, and low energy usage. The experimental results validate that such an approach supports efficient data transmission and onboard decision-making by largely compressing the data while maintaining high-quality reconstruction. Overall, the presented compression scheme presents a deployable, scalable, and robust solution to modern hyperspectral imaging missions operated by Unmanned Aerial Vehicles (UAVs).

Acknowledgements

I would like to express my sincere gratitude to all those who contributed to completing this research paper.

References

[1] Linmi Tao, and Atif Mughees, *Deep Learning for Hyperspectral Image Analysis and Classification*, 1st ed., Springer Singapore, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] Yaman Dua, Ravi Shankar Singh, and Vinod Kumar, "Compression of Multi-Temporal Hyperspectral Images Based on RLS Filter," *The Visual Computer*, vol. 38, pp. 65-75, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] Yaman Dua et al., "Convolution Neural Network Based Lossy Compression of Hyperspectral Images," *Signal Processing: Image Communication*, vol. 95, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[4] Md. Rakibul Haque et al., "A Lightweight 3D-2D Convolutional Neural Network for Spectral-Spatial Classification of Hyperspectral Images," *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, vol. 43, no. 1, pp. 1241-1258, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[5] Yaman Dua, Vinod Kumar, and Ravi Shankar Singh, "Comprehensive Review of Hyperspectral Image Compression Algorithms," *Optical Engineering*, vol. 59, no. 9, pp. 1-39, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[6] Jannick Kuester et al., "Adaptive Two-Stage Multisensor Convolutional Autoencoder Model for Lossy Compression of Hyperspectral Data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1-22, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Riccardo La Grassa et al., "Hyperspectral Data Compression Using Fully Convolutional Autoencoder," *Remote Sensing*, vol. 14, no. 10, pp. 1-14, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] Bart Beusen, Xenia Ivashkovyc, and T.V. Achteren, "Image Compression Using Vector-Quantized Auto-Encoders with Semantically Meaningful Feature Extraction," *4S Symposium*, pp. 1-13, 2022. [Google Scholar]

[9] S. Navaneethan, and N. Nandhagopal, "RE-PUPIL: Resource Efficient Pupil Detection System Using the Technique of Average Black Pixel Density," *Sādhanā*, vol. 46, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] Jiahui Yu et al., "Vector-Quantized Image Modeling with Improved VQGAN," *Arxiv*, pp. 1-17, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[11] Yuanyuan Guo, Yanwen Chong, and Shaoming Pan, "Hyperspectral Image Compression via Cross-Channel Contrastive Learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1-18, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[12] Bart Beusen et al., "On-Board Hyperspectral Image Compression Using Vector-Quantized Auto Encoders," *IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium*, Athens, Greece, pp. 1703-1707, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[13] Nafiseh Ghasemi et al., "Onboard Processing of Hyperspectral Imagery: Deep Learning Advancements, Methodologies, Challenges, and Emerging Trends," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 18, pp. 4780-4790, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[14] Afsana Afrin, and Md. Al Mamun, "A Comprehensive Review of Deep Learning Methods for Hyperspectral Image Compression," *2024 3rd International Conference on Advancement in Electrical and Electronic Engineering*, Gazipur, Bangladesh, pp. 1-6, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] Saurabh Kumar et al., "Onboard Hyperspectral Image Compression Using Compressed Sensing and Deep Learning," *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pp. 1-13, 2019. [Google Scholar] [Publisher Link]

[16] Michael Iliadis, Leonidas Spinoulas, and Aggelos K. Katsaggelos, "Deep Fully-Connected Networks for Video Compressive Sensing," *Digital Signal Processing*, vol. 72, pp. 9-18, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[17] Raúl Guerra et al., "A New Algorithm for the On-Board Compression of Hyperspectral Images," *Remote Sensing*, vol. 10, no. 3, pp. 1-41, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[18] Nan Zhao et al., "The Paradigm Shift in Hyperspectral Image Compression: A Neural Video Representation Methodology," *Remote Sensing*, vol. 17, no. 4, pp. 1-20, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[19] Yuting Wan et al., "Tailings Reservoir Disaster and Environmental Monitoring Using the UAV-Ground Hyperspectral Joint Observation and Processing: A Case of Study in Xinjiang, the Belt and Road," *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, Yokohama, Japan, pp. 9713-9716, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[20] Lili Luo et al., "Combining Different Transformations of Ground Hyperspectral Data with Unmanned Aerial Vehicle (UAV) Images for Anthocyanin Estimation in Tree Peony Leaves," *Remote Sensing*, vol. 14, no. 9, pp. 1-20, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[21] Bogdan Kovalenko, Vladimir Lukin, and Benoit Vozel, "BPG-Based Lossy Compression of Three-Channel Noisy Images with Prediction of Optimal Operation Existence and its Parameters," *Remote Sensing*, vol. 15, no. 6, pp. 1-24, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[22] Zizhang Li et al., "E-NeRV: Expedite Neural Video Representation with Disentangled Spatial-Temporal Context," *Computer Vision – ECCV 2022: 17th European Conference*, Tel Aviv, Israel, pp. 267-284, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[23] Shao Xiang, and Qiaokang Liang, "Remote Sensing Image Compression Based on High-Frequency and Low-Frequency Components," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1-15, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[24] Yuanyuan Guo et al., "Edge-Guided Hyperspectral Image Compression with Interactive Dual Attention," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1-17, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[25] Lili Zhang et al., "Compressing Hyperspectral Images Into Multilayer Perceptrons Using Fast-Time Hyperspectral Neural Radiance Fields," *IEEE Geoscience and Remote Sensing Letters*, vol. 21, pp. 1-5, 2024. [CrossRef] [Google Scholar] [Publisher Link]