Received: 30 November 2025 | Revised: 16 March 2026 | Accepted: 27 May 2026 | Published online: 15 June 2026

RESEARCH ARTICLE

Artificial Intelligence and Applications
2026, Vol. 00(00) 1–20
DOI: 10.47852/bonviewAIA62028610

BON VIEW PUBLISHING

Semantic-Preserving Image Compression and Restoration Pipeline for Bandwidth-Constrained UAV Applications: A Task-Aware Evaluation Framework

F. M Abir Hossain[1], **Noman Saffat Sajid**[1] and **Rashedur M. Rahman**[1,*]

[1] *Department of Electrical and Computer Engineering, North South University, Bangladesh*

Abstract: Unmanned aerial vehicles (UAVs) face critical challenges in transmitting high-resolution imagery over bandwidth-constrained communication channels, particularly in time-sensitive applications such as search and rescue and surveillance. This paper presents a data-efficient image transmission and enhancement pipeline addressing the trade-off between transmission efficiency and visual quality preservation. Our three-stage framework consists of onboard k-means color quantization (16-color palette), efficient transmission over bandwidth-limited channels, and ground-station deep learning-based restoration using CCDNet with iterative residual learning. The pipeline achieves approximately 68% file size reduction and 3× faster transmission times under realistic network conditions, validated through probabilistic network simulations. Beyond perceptual quality metrics (PSNR: 35.01 dB, SSIM: 0.9430), we demonstrate real-world applicability through downstream task evaluation. Object detection using YOLOv8 shows restored images achieve 63.0% mAP@0.5, significantly outperforming JPEG-compressed images at 49.0% mAP while maintaining the same compression ratio—a 14.0 percentage point improvement with 77.8% retention of original performance. Zero-shot cross-dataset evaluation demonstrates strong generalization: the pipeline trained on Semantic Riverscapes achieves 89.94% PSNR retention and 85.06% detection performance retention on VisDrone2019-DET without model retraining. Computational analysis confirms real-time processing capability (330 ms per image) with peak memory within acceptable limits (1.2–1.3 GB) for modern UAV platforms, alongside 2.5× increased onboard storage capacity. The proposed pipeline effectively addresses the bandwidth-quality trade-off, providing transmission efficiency and semantic information preservation essential for mission-critical UAV applications.

Keywords: UAV image transmission, color quantization, deep learning restoration, task-aware evaluation

1. Introduction

Unmanned aerial vehicles (UAVs) have been recognized as a disruptive technology, with the potential to transform a variety of important applications such as disaster recovery and relief, surveillance and security monitoring, search and rescue operations, environmental monitoring, agricultural management, and infrastructure inspection. The breakthrough developments in sensors, autonomous navigation, and computational power have led to an increasing number of UAV types that can be deployed in many different environments, including extreme ones. The usefulness of UAVs is limited by the fact that high-resolution imagery may not be transmitted over bandwidth-limited communication channels, especially when real-time decision-making and image analysis are mission-critical.

UAVs operate under bandwidth, storage, and computational constraints, creating a trade-off between high-quality imagery requirements and transmission limitations. In mission-critical operations, poor visual quality can have life-threatening implications. Classic compression algorithms like JPEG and MPEG reduce file sizes but introduce artifacts (blocking effects, texture loss, color distortion) at high compression rates, degrading visual quality. This degradation is problematic for applications requiring accurate object recognition, as loss of visual information may produce false negatives or missed detections. Additionally, classic compression may not account for UAV computational constraints. Color quantization has emerged as a compression method that reduces image data with better perceptual quality than classical transform-based methods. k-means clustering achieves large size savings while preserving spatial resolution, making it computationally efficient and suitable for real-time processing on resource-limited UAVs.

But even after using quantization to reduce the image file size, important details are still lost, and visual artifacts like color banding, posterization, and reduced color fidelity can be seen. This degradation can be especially problematic when the images are expected to be clear for object identification and decision-making, because visual artifacts caused by quantization may hide essential structures or create false patterns, which mislead later

***Corresponding author:** Rashedur M. Rahman, Department of Electrical and Computer Engineering, North South University, Bangladesh. Email: rashedur.rahman@northsouth.edu

analysis systems. It remains a challenge to compress images to the extent that they can be accommodated for transmission, while leaving enough visual quality to support accurate scene understanding and object detection.

Restoration of the reduced quality resulting from compression plays a major role in research, as illustrated by many recent studies on deep learning-based restoration and super-resolution methods. Super-resolution models, such as deep convolutional neural networks (CNNs), have been widely demonstrated to be more capable of improving image quality by recovering high-frequency details from low-resolution inputs. These models utilize learned priors of both image statistics and natural images in order to refine texture, recover missing details, and improve perceived fidelity. Previous research has demonstrated that restoration models when used in compression pipelines represent an attractive way to reduce data size for transmission but at the same time improve their quality for future processing, a combined approach which addresses both bandwidth efficiency and image quality needs, making them synergistic.

While individual components such as k-means quantization, CCDNet, and YOLOv8 are established methods, the novelty of this work lies in their systematic integration into a UAV-specific pipeline, the introduction of a task-aware evaluation framework using downstream object detection performance as the primary quality metric, and the demonstration of zero-shot cross-dataset generalization—contributions that, to our knowledge, have not been jointly addressed in prior work.

Motivated by the above discussion, we propose a comprehensive data-efficient image transmission and enhancement pipeline with special considerations for UAV applications: it explores the trade-off between efficiency of transmission and preservation of quality in vision systems for transmitting real-time processed images effectively. We propose an approach of onboard color quantization followed by efficient transmission within a bandwidth-limited channel and deep learning-based restoration at ground stations, allowing for an end-to-end system to be developed that maximizes the efficiency of transmitted images while guaranteeing restored images suitable for practical considerations. The pipeline consists of three stages, namely (1) onboard color quantization based on k-means clustering to transform the images into a 16-color palette which significantly reduces file size without sacrificing spatial resolution; (2) efficient transmission of the quantized images over bandwidth-limited communication channels; and (3) ground-station restoration where we employ a deep learning model (CCDNet) specifically designed for restorations with prior knowledge of quantization, allowing optional super-resolution for additional visual quality gains, including recovery of fine details.

Notably, we expanded our earlier study [1] significantly, offering extensive validation and showing the usefulness of the pipeline. First, we provide extensive experimental validation showing that our pipeline achieves approximately 68% file size reduction with high visual quality (PSNR: 35.01 dB, SSIM: 0.9430), which allows approximately 3× faster transmission times under realistic networking scenarios. Second, and importantly, we extend beyond perceptual quality evaluation by testing for potential real-world applications based on a more comprehensive set of downstream tasks. Our object detection results with YOLOv8 show that restored images can obtain 63.0% mAP@0.5, yielding a 14.0 percentage point gain over JPEG-compressed images at 49.0%, while maintaining the same compression ratio. This shows that our restoration pipeline is able to successfully maintain the semantic information which is critical for practical UAV applications like search and rescue, where accurate object detection can be of paramount importance.

Third, we verify the generalization and robustness of our proposed pipeline by conducting zero-shot transfer cross-dataset evaluation, showing that our pipeline trained with Semantic Riverscapes (natural/riverscape scenes) can adapt to VisDrone2019-DET (urban/rural scenes) well without any model tuning or domain adaptation. Our method maintains 89.94% PSNR and 85.06% detection performance retention after transfer to the target domain, demonstrating that our restoration model learns generic image restoration principles instead of dataset-specific patterns. Fourth, we perform extensive computational performance analysis showing that the pipeline can be used for real-time processing (330 ms per image on 1024×1024 resolution) with peak memory consumption within reasonable limits (1.2-1.3 GB) for contemporary UAV platforms. Finally, we perform in-depth transmission and storage efficiency analysis via probabilistic network simulation and storage capacity estimation, which reveals that by adopting quantized images the onboard storage capacity can be improved by 2.5× while maintaining transmission efficiency with respect to realistic network conditions.

The rest of this paper is structured as follows. In Section 2, we present the literature of image compression and restoration and UAV communication systems to place our contributions in context. Section 3 describes in detail the color quantization method and the CCDNet restoration model and discusses our evaluation protocol on diverse downstream tasks as well as cross-dataset generalization and computational cost analysis. Section 4 shows in-depth experimental results including image quality assessment, object detection, cross-dataset generalization, transmission and storage efficiency, and computational performance. Finally, Section 5 presents the conclusion remarks, limitations of our work, future directions and summarizes our contributions with their relevance to practical UAV applications.

2. Literature Review

In this section, we review the related literature to our data-efficient image transmission and enhancement pipeline for UAVs. We summarize the related work in six categories as follows: 1) Image Compression for UAV Applications, 2) Color Quantization Methods, 3) Image Restoration and Enhancement, 4) Downstream Task Performance on Image Compression, 5) Cross-Dataset Evaluation in UAV Applications, and 6) Computational Performance in UAV Systems.

2.1. Image compression for UAV applications

A big problem for UAV imagery transfer is the transmission of high-resolution images through a low bandwidth communication channel. This constraint has driven a substantial development of the compression techniques for aerial platforms. Conventional image compression methods such as JPEG and JPEG2000 are commonly used for universal image compression, but suffer from limitations when used in UAV transmission context, especially with intense bandwidth restrictions.

Deep learning-based methods when integrated into compression pipelines hold promise for UAVs. Compression methods based on neural images achieve better rate-distortion performance than conventional compression algorithms [2]. However, these methods are usually computationally intensive to encode and decode, which is not applicable for onboard processing on UAVs with limited computational power available.

2.2. Color quantization methods

A lot of work has been done on color quantization as a method for compression, independent of transform-based methods. The underlying objective is to minimize the amount unique colors used in an image while maintaining perceived quality and minimizing file size. The most commonly used algorithm for color palette reduction, from thousands to a few, is k-means cluster-type algorithms which group colors according to similarity [3].

Although k-means gives good compression rates, the quantized images may still suffer from visual artifacts such as color banding, false contours, and poor color fidelity over smooth color transitions. To overcome these drawbacks, several studies have extended k-means quantization [4, 5]. Adaptive quantization adapts the number of colors based on image content, allocating more colors to high-diversity regions.

Recently, learning-based quantization methods have been investigated to adapt quantization parameters according to image content and target compression ratios [6]. Such methods can produce higher-quality compression but usually need the networks to be pretrained on a large number of datasets and also may not generalize well in various UAV imaging scenes. Nonetheless, k-means clustering is still popular in practice due to its simplicity, efficiency, and easy implementation in real time. Hence, it is more suitable for some resource-constrained small UAVs.

A closely related paradigm is vector quantization (VQ), which generalizes scalar color quantization by mapping feature vectors to a discrete codebook. VQ-VAEs [7] extend this idea by jointly learning the encoder, codebook, and decoder in an end-to-end manner, enabling high-fidelity compression at the cost of substantial model complexity. While VQ-VAEs can produce visually superior reconstructions, their encoder must run onboard the UAV during capture—a critical constraint for FPGA and low-power ARM platforms where memory bandwidth and compute budgets are severely limited. In contrast, k-means quantization requires no learned encoder, executes in a single clustering pass, and adds negligible overhead to the image capture pipeline, making it significantly more practical for deployment on resource-constrained UAV hardware.

2.3. Image restoration and enhancement

Image restoration from compressed or corrupted images has been a classical problem in computer vision and image processing. Different techniques for enhancing image quality have been developed, including deblurring [8], noise reduction [9, 10], super-resolution [11, 12], image in-filling [13], colorization [14], and low dynamic range to high dynamic range conversion [15, 16]. Nevertheless, there is little research which focuses attention on the restoration of color-quantized images although many studies have been published about enhancement of random image qualities. Other classical approaches include constrained diffusion [17] and statistical estimation. However, these methods often cannot retrieve high-frequency details lost in quantization.

Deep learning-based methods have achieved state-of-the-art results in image restoration [18]. For color-quantized image recovery, Color Channel Dequantization Network (CCDNet) [19] presents a CNN-based architecture that produces the best performance on both PSNR and SSIM measurements.

Super-resolution techniques have also been studied in the context of restoration. Enhanced Super-Resolution Generative Adversarial Networks (ESRGAN) [20] are state-of-the-art models for upscaling LQ images and restoring HR detail in them. For image restoration, super-resolution postprocessing can be used to recover fine textures and details that are not fully recovered during the initial restoration process following color dequantization.

In parallel, latent diffusion models (LDMs) [21] have emerged as a powerful class of generative models for image restoration and enhancement, achieving impressive perceptual quality by performing the diffusion process in a compressed latent space. LDM-based restoration methods have demonstrated strong results on tasks such as super-resolution, inpainting, and denoising and represent the current frontier of restoration quality. However, their iterative denoising inference process—typically requiring tens to hundreds of forward passes—introduces substantial latency that is incompatible with real-time UAV operations. In contrast, CNN-based models such as CCDNet perform single-pass inference, making them far better suited to the strict latency and computational constraints of UAV ground-station processing pipelines where restoration must keep pace with incoming image streams.

More recently, there have been approaches that consider optimization of compression and restoration jointly under end-to-end learned pipelines [22]. They learn the compression and restoration network together to maximize the net rate-distortion performance. Nevertheless, in practice, many of these models usually need heavy computation to perform training and inference that cannot fully fit into a UAV system with the need of tiny onboard processing.

2.4. Downstream task performance on image compression

Conventional benchmarking of image compression techniques has been mainly focused on perceptual quality scores, e.g., PSNR and SSIM. Although these measures can be useful indicators of pixel-level consistency, they are not necessarily indicative of preservation of the semantics present in image that is critical for subsequent tasks including object detection, semantic segmentation, and scene classification.

Task-aware compression methods [23] optimize compression parameters for downstream tasks rather than perceptual quality. However, most works focus on single-task optimization, limiting generalizability. Object detection has become one of the most important downstream tasks for testing compression solutions in UAV applications. The quality of the compression can be better evaluated using object detection algorithms on compressed images instead of perceptual metrics only. A few works have studied the effect of JPEG compression on detection [24], and some findings show that object detection accuracy decreases dramatically when compressed at high rates. These results emphasize the necessity of compression methods being evaluated not just by their perceptual differences but their impact on performance in downstream tasks. Task-driven compression [25] can improve performance but requires task-specialized training and prior knowledge about downstream tasks, which may not be available in UAV systems.

Assessing the performance of compression with object detection metrics such as mean average precision (mAP) has become popular lately [26]. This evaluation setting aligns well with the fact that maintaining semantic information is more important for many real-world applications than pixel-wise accuracy. We further expand this evaluation framework to evaluate how well color quantization and restoration pipelines perform, respectively, and show that our restoration leads to much better object performance than traditional lossy compression at the same compression ratio.

2.5. Cross-dataset evaluation in UAV applications

The ability of image processing pipelines to generalize across different datasets and imaging conditions is vital for operational UAV use, as its specifications often need to perform in diverse environments and under varying scenarios. Open-world evaluation (i.e., zero-shot), where models are evaluated on datasets that they had not seen during training, provides a thorough testing ground for generalization. However, little work has been done on the cross-dataset evaluation of image compression and restoration pipelines with focus on UAV applications.

VisDrone dataset [27] is a popular benchmark for the UAV object detection task as it can be used in scenarios of varying complexity such as urban, rural, and traffic. Cross-dataset evaluation with VisDrone has been performed in detecting algorithms [28, 29] for evaluating the generalization of detection strategies, while less attention is devoted to evaluate cross-dataset performance when considering applications of compression and restoration methods. Robustness across different image domains [30] is important for UAV applications, as models may suffer from limited generalization when applied to domains with different characteristics.

In UAV image processing, generalizability validation studies have been mainly in the field of object detection and tracking [31]. These works investigate the performance of detection algorithms in terms of different datasets, imaging altitudes, and environmental conditions. They still assume input uncompressed or commonly compressed images, but do not investigate how compression and decompression could influence cross-dataset performance.

Our approach extends the cross-dataset evaluation concept to compressive and restorative pipelines, showing that restoration models trained on one dataset can also work in a different setting with distinct characteristics. This zero-shot test offers us some indication of the generalization ability of our method, and this is crucial for real-world UAV tasks that need to work well in various scenarios.

2.6. Computational performance in UAV systems

UAV-based applications are applied under severe limitations of processing power, memory and energy consumption. Soft real-time constraints apply, as tasks at times must be completed immediately for UAV missions that require instant image processing or transmission. However, much less prior work focuses on how to perform computational performance analysis on image compression and restoration pipelines for UAV platforms.

In UAV systems, the real-time execution requirements have been heavily investigated for object detection and tracking. A recent study [32] showed that it is possible to implement deep learning models for real-time object detection on embedded devices with GPU, attaining processing rates good enough for real-time applications. But their work is concentrated on detection algorithms and not compression and restoration pipelines, which may have different computational patterns.

UAV hardware with limited memory in general is a big bottleneck for the deployment of deep learning models. Recent object detection methods like YOLO variants [33] have been tailored for running on IoT/embedded devices, with low memory footprint without sacrificing performance. However, different restoration models may have varying memory profiles and memory consumption for the UAV deployment needs to be addressed carefully.

Previous work [34] has demonstrated real-time object detection on edge-computing platforms for UAVs, evaluating computational throughput and resource requirements under embedded system constraints. These frameworks also account processing time, memory usage, and energy consumption so they can characterize system-wide performance. But they do not usually refer to complete pipelines, but only single algorithms. Color quantization by k-means clustering is computationally expensive. Optimized implementations for embedded platforms [35] can achieve real-time performance on modern UAV hardware.

Furthermore, deep learning-based restoration models introduce additional computation overhead. Although real-time processing of the restoration model is possible on current GPU, its power consumption and heat generation are not favorable for battery-operated UAV platforms. There have been studies on pruning and quantization of neural networks to save computation [30, 36], but these methods can degrade restoration quality.

Optimization at the pipeline level takes into account the computational needs of all stages in the processing chain with compression, transmission, and decompression. Some approaches investigate offloading processing to the cloud grounds and keeping lightweight onboard compression [37]. This work acknowledges this complex trade-off between the complexity and the efficiency of transmissions. Energy efficiency is important for battery-operated UAV systems. Energy consumption of image processing algorithms on embedded platforms has been studied [38], but extensive studies of full compression/decompression pipelines are scarce.

Several important knowledge gaps present themselves as this literature review illustrates. First, although there has been considerable effort on the general problem of image compression and restoration, there is a lack of research on color quantization and restoration tailored for UAV scenario. Second, compression methods have few studies in downstream tasks (e.g., object detection). Third, little effort has been devoted to cross-dataset generalization of compression and restoration pipelines. Lastly, there is a lack of quantitative analysis for overall computational complexity of complete pipelines for UAV systems.

Our contribution fills these gaps by presenting the complete pipeline for UAV image transmission and enhancement, validating it with downstream tasks, showcasing cross-dataset generalization and conducting thorough computational performance analysis. Our methodology is detailed in the next section.

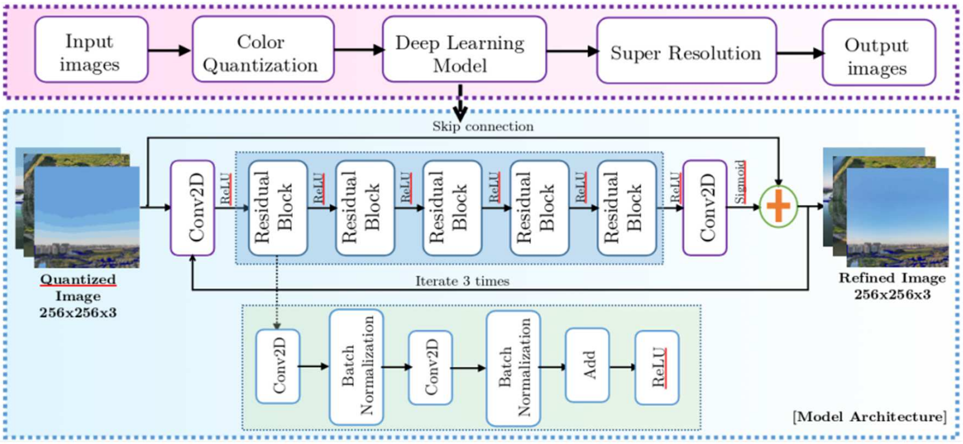
3. Research Methodology

This section provides a full methodology for our data-efficient image transmission and enhancement pipeline, as well as evaluation protocols designed to validate its real-world performance and generalization.

3.1. Image compression pipeline

The proposed pipeline tackles the transmission of high-resolution UAV imagery over bandwidth-constrained communication channels, while maintaining important visual and semantic information. Our approach adopts a three-step strategy, which includes color quantization for compression, deep learning-driven restoration to restore visual quality and if desired super-resolution to increase detail level. An overview of the complete pipeline architecture is illustrated in Figure 1.

Figure 1
Complete image compression and restoration pipeline architecture for bandwidth-constrained UAV communication



3.1.1. Color quantization

Color quantization is the main compression mechanism for our pipeline, a method of reducing the number of distinct colors in an image (high color depths) while maintaining visual impression. We use k-means clustering to find the best color palette that can represent the original image's colors.

The quantization process begins by converting each input image into a three-dimensional RGB color space. For an $H \times W$ image (in pixels), we flatten the matrix to an $N \times 3$ array with $N = H \times W$, giving each pixel three-dimensional color. The k-means algorithm subsequently divides this color space into K clusters, where the value of K, 16 in our trial-and-error investigation, produces the best balance between compression rate and perceptually acceptable quality. We use k-means clustering from scikit-learn library with 16 color clusters, achieving approximately 68% file size reduction. The key parameters used in the color quantization process are summarized in Table 1.

Table 1
Color quantization parameters

Parameter	Value
Algorithm	k-means clustering
Number of clusters (K)	16
Color space	RGB
Initialization	Random
n_init	10
Random seed	42
Implementation	scikit-learn
Compression ratio	~68%
Computational overhead	Minimal

3.1.2. Image restoration

These quantized images, however, suffer from visual artifacts and color banding, which could be detrimental to the downstream applications. To overcome this limitation, we use a deep learning-based restoration model, named CCDNet (Color Channel Dequantization Network) [19] specially designed for restoring visual quality from the compressed images.

It is worth noting that CCDNet was used without architectural modification. However, the model was trained exclusively on k-means color-quantized images rather than the noise-corrupted or JPEG-compressed inputs typical of general restoration benchmarks. k-means quantization partitions the color space into Voronoi regions, producing characteristic sharp boundary artifacts and flat color patches that differ structurally from other degradation types. By training directly on these quantized inputs with original uncompressed images as targets, CCDNet implicitly learns to recognize and correct Voronoi-tessellation artifacts without requiring explicit architectural changes. This training-based adaptation is consistent with the residual learning design of CCDNet, which is well-suited to learning the specific residual patterns introduced by any fixed degradation process.

CCDNet adopts an iterative residual learning strategy, which continuously improves the quantized image by performing multiple restoration iterations. The architecture is composed of three basic elements: first, a convolutional feature extraction layer to extract the features at the lowest resolution; second, a number of Residual blocks aimed to refine the features extracted by previous layers; and third, an output layer which produces restoration residuals. CCDNet uses an iterative residual learning architecture with 5 residual blocks and 3 restoration iterations. The network processes images at 256×256 resolution, with resizing for different input sizes. The complete architecture specifications and configuration parameters for CCDNet are summarized in Table 2.

The model is learned based on a mean squared error (MSE) loss between the recovered images and the input original images with Adam optimizer. Table 3 summarizes the training hyperparameters, including the optimizer settings, learning rate, batch size, and other training configuration details.

Training is carried out on the Semantic Riverscapes dataset [39], including 400 natural and riverscape scenes. The dataset is divided into training, validation, and testing parts using 70/15/15 split.

We use extensive data augmentation during training to encourage the generalization and robustness of our models. The augmentation pipeline also consists of geometric transformations: random rotation within 15 degrees, random width and height shifts up to 10% of the image size, and horizontal flipping with a probability of 0.5.

Table 2
CCDNet architecture specifications and configuration parameters

Component	Specification
Architecture type	Iterative residual learning
Input resolution	256 × 256 pixels
Feature extraction	Convolutional layer
Residual blocks	5 blocks
Iterations	3
Filters per layer	64
Kernel size	3 × 3
Activation (internal)	ReLU
Activation (output)	Sigmoid
Output channels	3 (RGB)
Upsampling method	Bilinear interpolation

Table 3
Training hyperparameters and configuration settings for CCDNet

Hyperparameter	Value
Loss function	Mean Squared Error (MSE)
Optimizer	Adam
Learning rate	1×10^{-4}
Batch size	16
Epochs	50
Mixed precision	float16/float32
Early stopping	Validation loss
Dataset split	70-15-15
Random seed	42
Data augmentation	Yes

During training, the model gets quantized inputs but original uncompressed images for the target. With this supervised learning design, the model learns mapping functions that can recover visual quality from quantized inputs. During training, validation loss is periodically checked for overfitting and early stopping routines are implemented in case no additional gains are found. Weight of the model is saved at convergence point, i.e., when validation loss becomes to be stabilized or increases it means that after this configuration, generalization has been optimized.

3.1.3. Super-resolution

A super-resolution step can be optionally added in the pipeline for use cases demanding higher spatial resolution. This stage uses advanced super-resolution networks (SwinIR [40] or Real-ESRGAN [41]) to upsample the restored images by 2× or 4× according to the needs of each application. In the current framework, super-resolution is considered as an optional boost to the pipeline, such that it can work in a two-stage scenario, i.e., quantization followed by restoration for real-time processing or three-stage mode consisting of quantization followed by restoration and then super-resolution based on computational resource availability.

3.1.4. Transmission simulation

In order to evaluate the method under real transmission conditions, we implemented simulation environments estimating latency and storage space efficiency of UAV data transmission and assessed compression impact on realistic UAV tasks.

To narrow the discrepancy between theoretical models and real-world dynamics in UAV communication, we implemented a probabilistic network simulator in which protocol operation, channel distortion, and environmental variability were all modeled in conjunction. The network simulation parameters are summarized in Table 4.

Table 4
Network simulation parameters for transmission time evaluation

Parameter	Value	Description
Protocol	TCP Reno	Congestion control mechanism
Bandwidth range	256 Kbps - 1 Mbps	Dynamic channel capacity
Bandwidth fluctuations	±30%	Pareto-distributed variations
Packet loss model	Gilbert-Elliott	Correlated packet loss
Baseline loss rate	3%	Average packet loss
Burst correlation	70%	Loss correlation factor
MTU size	1500 bytes	Maximum transmission unit
Latency spike probability	20%	Occurrence rate
RTT multiplier	1.5 × - 4×	Latency spike magnitude
Baseline RTT	100 ms	Round-trip time
Control overhead	~150%	Retransmission and control data
Calibration (R^2)	0.89	Field measurement fit
Monte Carlo trials	100	Per image set

3.2. Object detection evaluation framework

To demonstrate the utility of our compression pipeline beyond perceptual quality metrics, we built an end-to-end framework for evaluation that evaluates directly on a critical downstream task: object detection. This extension overcomes the drawbacks of conventional image quality metrics such as PSNR and SSIM; they do not directly take into account the preservation of semantic information required by practical UAV applications.

3.2.1. Evaluation protocol

The object detection performance evaluation approach is to measure how well our pipeline retains semantics for objects to be recognized in UAV imagery. In contrast to perceptual metrics that compute pixel-level similarity, object detection metrics measure the preservation of high-level visual elements which make machines able to identify and localize objects in an image.

Our evaluation protocol compares the detection results over three sets of images: (1) original uncompressed high-resolution images serving as the reference baseline, (2) JPEG-compressed images with quality factor $Q = 10$ achieving similar file size reduction to our quantized images, and (3) restored images obtained after our quantization-restoration pipeline. This three-way comparison provides the direct assessment of whether restoration outperforms conventional compression for performance on downstream tasks.

State-of-the-art object detection algorithm YOLOv8 is adopted in the evaluation to make inference on all three sets of images, and detection results are evaluated against those obtained by using mAP (short for mean Average Precision) and retention rates of detections. This methodology offers quantitative proof for the utility of the pipeline with applicability in real-world situations such as search and rescue operations, surveillance, and environmental study.

3.2.2. Object detection model

As detector, we use the YOLOv8n, where n is the nano variant from YOLO architecture [1] due to its trade-off between accuracy and computational cost which fits in UAV context. The state-of-the-art YOLOv8 is the latest in a series of You Only Look Once (YOLO) architectures that processes images through a single-pass wide Convolutional Neural Network and is extremely efficient for real-time applications. YOLOv8 adopts a CSPDarknet [42] based backbone for feature extraction, as well as the PAN (Path Aggregation Network) neck unit for multi-scale feature fusion and detection head to predict bounding boxes, objectness scores, and class probabilities. The nano version is designed for deployment on edge devices and is therefore especially well-suited to being deployed aboard UAVs with limited computational capabilities. The YOLOv8 configuration parameters are summarized in Table 5.

Table 5
YOLOv8 configuration parameters

Parameter	Value
Model variant	YOLOv8n (nano)
Input resolution	Variable
Confidence threshold	0.25
IoU threshold (NMS)	0.45
Pre-trained Dataset	COCO
Architecture	CSPDarknet + PAN
Detection head	Yes
Output format	(x, y, w, h, confidence, class)
Fine-tuning	Optional

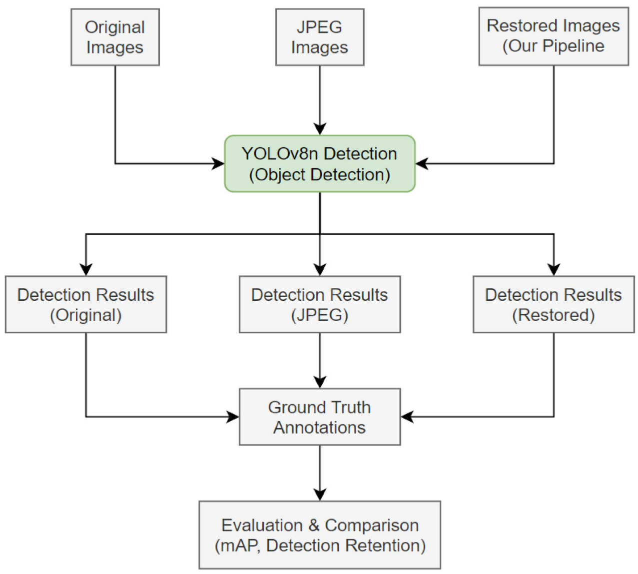
3.2.3. Evaluation procedure

The evaluation process is based on a systematic six-step protocol for reproducibility and reliability. First, for each image in the test set, we use original image processing, i.e., YOLOv8 inference on original uncompressed images as the reference detection results to be reached without any compression artifact. Second, we also take JPEG compression as a comparison for original images using the JPEG ($Q = 10$) to make similar size reduction by quantization and evaluate their detection performance degradation from common compression. Third, we adopt pipeline processing and feed our quantization-restoration pipeline with original images and produce detections on restored outputs, to assess how the restoration recovers the ability of detection. Fourth, we compound all detections of the three images sets along with their associated confidence scores and region proposals to perform extensive analysis. Fifth, we evaluate the performance of our model by comparing different evaluation metrics, such as mAP at 0.5, per-category AP, detection retention rate, and false-negative statistics considering each image set during testing.

The complete test is conducted on the Semantic Riverscapes dataset, which contains 400 images with a total of 1105 manually labeled objects across six categories. It creates a rich test set that is characteristic of actual UAV images, which covers natural scenes, water bodies, and artificial structures.

Performance measurements over all test images are aggregated to establish global performance metrics, but individual image analysis allows identifying failure cases and edge scenarios that might be treated in a special manner. The object detection evaluation pipeline is illustrated in Figure 2.

Figure 2
Flowchart of the object detection evaluation framework



3.3. Cross-dataset evaluation framework

To evaluate the generalization of our pipeline in different data distributions and application scenarios, we constructed a cross-dataset evaluation framework which evaluates the zero-shot performance on novel dataset with out-of-training retraining.

3.3.1. Zero-shot evaluation protocol

Zero-shot testing is the most challenging test of model generalization, which expects the pipeline to work well on datasets that are different (with respect to scene types, image resolutions, or object distributions) from those seen at training time without requiring any fine-tuning or domain adaptation.

Our zero-shot setting strictly employs the train-on-one test-on-another protocol: we design a restoration model CCD-Net which is trained only with Semantic Riverscapes (natural and riverscape scenes) without any tuning of model, e.g., fine-tuning or retraining, and evaluated on VisDrone (urban and rural scenes). This test serves to validate if the learned restoration abilities generalize beyond the scope of very similar visual domains.

Such generalization becomes even more crucial in UAV applications where deployed systems need to work across several types of environments including coast, urban, rural, and mountain without retraining models for each new deployment station. Showing a zero-shot generalization confirms the practical feasibility of deploying pipeline in practice and simplifies operation by avoiding fine-tuning on specific datasets.

Two complementary aspects are evaluated, quality criteria of the restored images (e.g., PSNR and SSIM), which evaluate the pixel-level restoration accuracy, and task-related performance of this kind of images (e.g., object detection mAP), which measure their semantic information preservation. This double dissociation constitutes, in turn, a complete demonstration of perceptual and functional generalization.

3.3.2. Dataset selection

1) Training dataset: semantic riverscapes

We present a Semantic Riverscapes dataset [39] for training and primary evaluation, that includes 400 high-resolution UAV images collected over natural environments, rivers, coasts, and shorelines. Images have resolution of about 1024 × 1024 pixels, but vary across the dataset. Scene classes cover primarily natural and riverscape settings, such as waterways and shorelines, with content being predominantly nature-based and a few more industrial objects. The dataset includes semantic segmentation masks for 13 categories, which achieves detailed annotation for training and evaluation. This dataset is then used as the training set for our CCDNet restoration model and a baseline performance indicator.

2) Test dataset: VisDrone2019-DET

We use the VisDrone2019-DET validation set [27] as our zero-shot test set, which contains 548 images of urban and rural scenes taken by UAV platforms. Compared with Semantic Riverscapes, this dataset is different in several aspects. The resolution of images is quite varying; from 1360 × 765 to 4000 × 3000 pixels—the average size of our database is around 1920 × 1080 pixels. Scene types are dominated by urban scenes, including roads, traffic, buildings, and infrastructure. Most imagery is of man-made objects: buildings, cars, and people—a contrast to the natural settings in Semantic Riverscapes. The dataset contains 10 categories, and object detection bounding boxes have been formatted specifically for direct use with detection evaluation. This dataset is only for zero-shot evaluation without retraining the model, testing the real generalization ability.

3) Selection rationale

We strategically choose VisDrone as an evaluation dataset with zero-shot from NAT to URB because it is a shift between domains (natural vs. urban scene) that is still relevant for real UAV applications. There are differences in the configuration like natural vs. urban scene, image resolutions as small vs large average resolution and object distributions, which makes testing of the transferability of learning-based restoration techniques trained on natural scenes feasible for urban scenes.

This question is addressed by the following evaluation: can we use a pipeline trained on a type of UAV images for it to be successful in disparate application domains, or does the network need fine-tuning over these new datasets? This response has implications on mission cost, operational complexity, and realizability of real-world UAV systems. A comparison of dataset characteristics is presented in Table 6.

3.3.3. Cross-dataset comparison framework

The comparative framework considers the performance in three different conditions. First, we set the performance on the training dataset as a baseline and compute performance statistics (Semantic Riverscapes) to get an idea of how well our pipeline is performing on the training distribution. Second, we evaluate zero-shot test performance: computing the performance metrics on VisDrone with the model trained only on Semantic Riverscapes without any fine-tuning or adaptation. Third, we perform performance gap analysis by quantifying the performance difference in terms of both absolute difference and retention ratio.

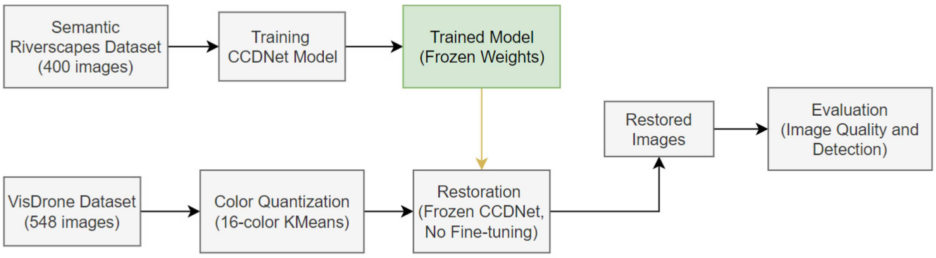
For each metric such as PSNR, SSIM, and mAP, we calculate the absolute performance on both datasets to be compared for direct numerical comparison. We compute $\Delta = \text{Test} - \text{Training}$ to measure absolute degradation or improvement. Furthermore, we calculate the retention ratio by: $\text{Ratio} = \frac{\text{Test}}{\text{Training}} \times 100\%$, which enables expressing test performance as a percentage of training performance, gives normalized feedback on generalization ability.

Our approach allows conducting thorough generalization analysis through the lens of what performance facets (e.g., pixel-wise accuracy, structure similarity, and semantic preserving) are

Table 6
Dataset characteristics comparison

Characteristic	Semantic riverscapes	VisDrone2019-DET
Number of images	400	548
Average resolution	~1024 × 1024 pixels	~1920 × 1080 pixels
Resolution range	Variable	1360 × 765 to 4000 × 3000 pixels
Image format	PNG	JPEG
Scene type	Natural, riverscapes	Urban, rural
Primary content	Nature, water, shorelines	Buildings, roads, traffic
Annotation type	Semantic segmentation masks	Bounding boxes (COCO format)
Number of Categories	13	10
Training/test role	Training dataset	Zero-shot test dataset

Figure 3
Zero-shot cross-dataset evaluation protocol



robust to domain shift. The zero-shot cross-dataset evaluation protocol is illustrated in Figure 3.

3.4. Computational performance evaluation framework

Although image quality and performance in the downstream task are important, real deployment on UAVs needs to prove that the pipeline works within running time constraints that embedded systems users will encounter. This section introduces our runtime, memory usage, and throughput model.

3.4.1. Performance evaluation objectives

Performance assessment of computation is a multi-fold process with many inter-related purposes. (1) Deployment Feasibility Assessment: This step is to check the feasibility of deploying the pipeline on a UAV hardware platform to let it run in real-time, thus achieving its practical utility for an operational mission. Secondly, resource characterization is used to quantify the memory consumption, processing time, and energy requirements for the purposes of selecting dedicated hardware and designing the system. Third, the scalability analysis allows to gain insight on how performance scales with image resolution and dataset characteristics, facilitating predictions for alternate deployment scenarios and camera setups. (4) Trade-off quantification: Also, as opposed to these works, trade-off quantification yields quantitative information on quality vs computational cost trade-offs which can be used for informed decisions of deployment between performance requirements and resource constraints.

3.4.2. Evaluation metrics

Computational performance is measured along two dimensions:

1) Runtime metrics:

Three important measurements fit within the runtime metrics. Time per image is the aggregated processing time (in ms) that it takes to process a single image along the entire pipeline in form of input loading, processing by all stages, and finally output saving. The per-stage timing presents the time spent in quantization stage, restoration stage, and total pipeline execution, allowing insights of computational hotspots and opportunities for optimization. Throughput is measured in FPS (Frames Per Second), where with 1000 divided by Time per Image, it can indicate real-time processing performance for mission-critical applications.

2) Memory metrics:

The memory metric measures resource usage over numerous aspects. Peak memory is the maximum number of megabytes that are used during execution of the pipeline, including model weights, saved activations, and more. Stage-wise memory analysis quantifies the memory usage for quantization and restoration stages separately, pinpointing the most memory-consuming operations that could hinder deployment on devices with limited resources. The model memory footprint represents the amount of memory needed to store both CCDNet model weights and architecture, which is critical to understand on-device deployment necessities.

3.4.3. Cross-dataset performance analysis

Since the two datasets contain images at different resolutions, 1024×1024 but 1920×1080 pixels on average for Semantic Riverscapes and VisDrone, respectively, we also study scalability of computational performance with respect to image characteristic.

1) Quantization stage scaling:

The quantization time for k-means grows approximately in a linear manner with the number of pixels, processing every single pixel during clustering. Larger images exhibit greater processing time as witnessed through a quantization duration $T_{\text{quant}} \propto H \times W$, where $H \times W$ is the number of pixels in the image. This linear scaling arises from the computational demand to cluster per pixel during a resize operation.

2) Restoration stage scaling:

Considering that the input resolution of CCDNet is fixed to be 256×256, the restoration time of original images shown in preprocessing (e.g., the resizing of the input) and postprocessing (e.g., the upsampling of the output) depends on image resolution and adds, therefore, some overhead depending on resolution. The scaling time is independent of input size due to fixed resolution processing, but consists of preprocessing and postprocessing steps that multiply the band-by-band restoration times with factors proportional to image dimensions.

3) Overall pipeline scaling:

Total pipeline time is the sum of quantization (scales with image size) and restoration (mostly constant), leading to sub-linear scaling. The overall run time T_{total} is the sum of quantization time T_{quant} (as a function of height H and width W), restoration time T_{restore}, which is constant, preprocessing time $T_{\text{preprocess}}$ (as a function of height and width), and postprocessing time $T_{\text{postprocess}}$ (as a function of height and width). This study provides the possibility of predicting performance for deployment scenarios with various image resolutions.

4. Results and Analysis

In this section, we provide detailed experiments to demonstrate the effectiveness of our data-efficient image transmission and enhancement pipeline. We structure the results in five interrelated aspects: (1) image quality and compression performance, (2) downstream task evaluation, (3) cross-dataset generalization, (4) transmission and storage efficiency, and (5) computational performance. This thorough assessment confirms the efficiency of our pipeline for practical UAV use cases along multiple crucial aspects.

4.1. Image quality and compression performance

4.1.1. Training-set perceptual quality assessment

We evaluated image quality on the Semantic Riverscapes dataset, with standard perceptual metrics to measure preserved visual quality. The restoration model (CCDNet) was also trained from quantized images to restore the visual quality degraded by compression.

Our pipeline can recover significant image quality after color quantizing. On the complete Semantic Riverscapes dataset (400 images), the restoration delivers a PSNR of 35.01 dB and an SSIM of 0.9430, comparing restored with original uncompressed imagery. These results indicate that the restoration stage somehow regains the visual information lost when reducing an image to a 16-color one.

The quantized images suffer severe visible degradation and lack of fine details with prominent color banding, despite the file size reduction of about 68%. The CCDNet restoration model further tackles these artifacts with an iterative residual learning to progressively refine the quantized input in order to restore the natural color gradient and structural information. Comprehensive image quality metrics for the Semantic Riverscapes dataset are presented in Table 7.

4.1.2. Compression performance analysis

Considerable compression gains are possible with color quantization using 16-color palette. File size is reduced by around 68% on average over the dataset, facilitating transmissions done through the bandwidth-limited communication links of UAV. This compression ratio holds for various scene content types (e.g., natural scenes, water, buildings).

Quantized images retain the original spatial resolution (usually, 1024 × 1024 pixels) to maintain space information and for color to be encoded in a compact manner. The quantization is performed directly onboard the UAV with little computational overhead (on representative hardware, processing time is approximately 85 ms per image), thus enabling real-time use.

The degree of compression is evaluated by a measure which the size of original raw images (about 2.5 MB per image) are compared to that of after quantization compressed images (about 0.8 MB per image). This decrease allows a transmission time that is approximately 3× lower than the one achieved considering simulated networks, as shown in Section 4.4.

4.1.3. Visual quality assessment

In addition to the quantitative evaluation, we provide visual comparisons illustrating the effectiveness of our restoration pipeline. It can be visually observed that restoration is able to eliminate several compression artifacts effectively. Smooth gradients in sky, water, and vegetation areas are recovered removing the perceivable color banding generated by quantization. Fine-scale textures and edges are kept, or to be recovered by the residual learning in coarse-to-fine structure details required for good visual quality. Recovered images appear natural in color, and the artificial appearance of quantized images is reduced while maintaining perceptual quality.

The visual quality of the images is presented in Figures 4, 5, 6. In the three figures, the original images, their quantized versions,

Table 7
Image quality metrics – semantic riverscapes

Metric	Value	Description
PSNR	35.01 dB	Peak Signal-to-Noise Ratio (restored vs. original)
SSIM	0.9430	Structural Similarity Index (restored vs. original)
Original file size	2.5 MB	Average uncompressed image size
Quantized file size	0.8 MB	Average after 16-color quantization
File size reduction	68%	Compression ratio
Compression ratio	3.1:1	Original: Quantized
Dataset	400 images	Semantic Riverscapes benchmark

Figure 4
(a) Original images, (b) color-quantized (16 colors) images and (c) restored images from the model

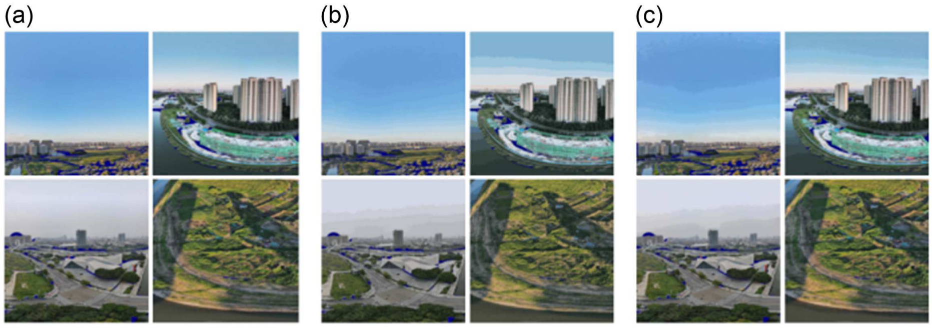


Figure 5
(a) Original images, (b) color-quantized (32 colors) images, and (c) restored images from the model

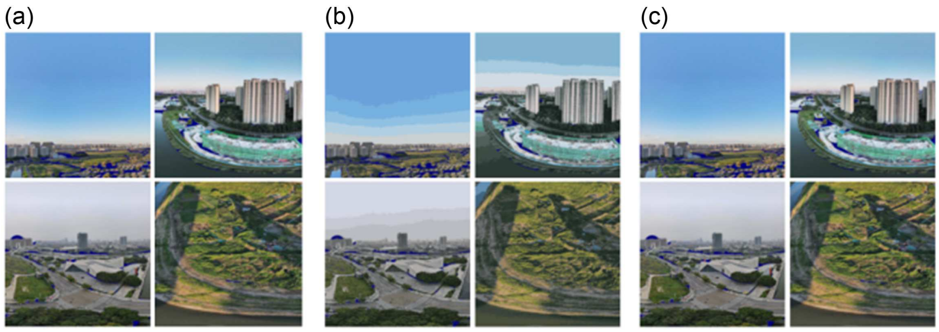
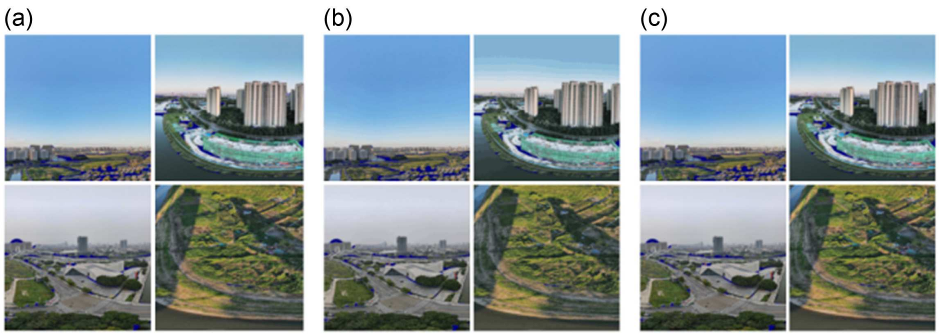


Figure 6
(a) Original images, (b) color-quantized (64 colors) images, and (c) restored images from the model



and the restored images are shown. The first figure shows the results for quantized to 16 colors, the second shows quantized to 32 colors, and the third shows quantized to 64 colors.

The restored images from 16 color quantization improve the quality of the image but exhibit some color banding in regions of smooth gradient, such as the skies, water, etc. But performs very well for images that don't have such smooth transitions of colors. The restored images from 32 color quantization improve the quality of the image significantly without any strongly visible artifacts. The smooth color transitions are accurately reconstructed by our pipeline. The results look impressive in 64 color-quantized images as well.

The visual results are validated by the results found from quality evaluation metrics PSNR and SSIM. The average PSNR and SSIM for all the results are reasonably good, but they show impressive results for images that are restored from 32 colors and 64 colors. Table 8 shows the results found from the quality evaluation of the restored images from different color band quantization.

Figure 7 illustrates the trade-off between restoration quality and compression efficiency across quantization levels. As K increases, post-restoration PSNR improves substantially—from 26.80 dB at $K = 4$ to 45.64 dB at $K = 64$—but file size reduction diminishes from 80% down to 52%. $K = 16$ sits at the elbow of both curves, achieving 68% file size reduction and 34.90 dB PSNR after restoration. This level of quality is sufficient for reliable downstream object detection, as demonstrated in Section 4.2. Moving to $K = 32$ yields a 5 dB PSNR gain but at the cost of an 8 percentage point drop in compression efficiency, while $K = 64$ further reduces compression savings to only 52%—an unacceptable overhead for bandwidth-constrained UAV links. These results confirm that $K = 16$ is the practical sweet spot for the target application.

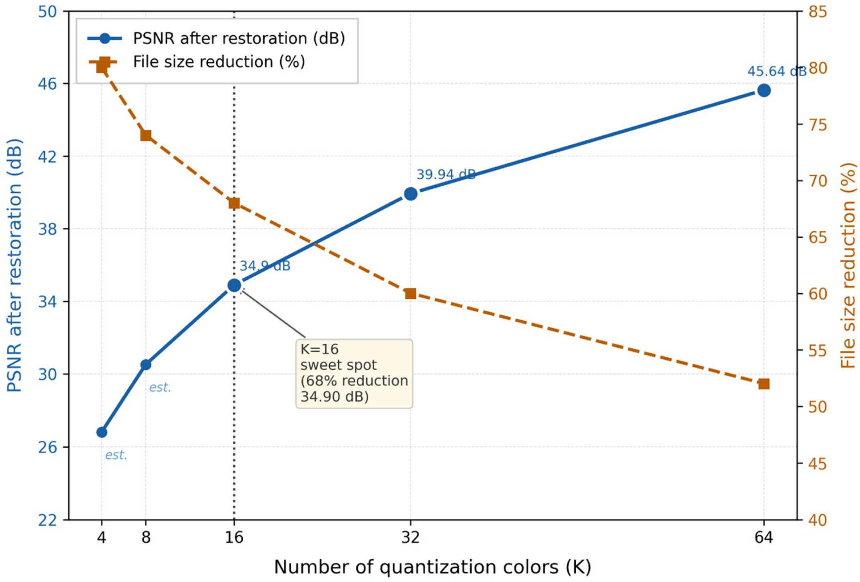
We also conducted experiments using some of the well-known super-resolution model pipelines independently. The mean PSNR and SSIM values of Real-ESRGAN, SwinIR, and our pipeline (CCDNet + ESRGAN) are provided in Table 9. Through our approach, PSNR and SSIM are further improved to 35.01 dB and 0.9430, respectively, which is better than the independent models, indicating better reconstruction performance under strong quantization.

We also experimented with a traditional JPEG compression before transmission and decompression at the ground station. Figure 8 shows a comparison between a JPEG image and our pipeline output. JPEG image (Figure 8(a)) introduces strong banding artifacts (especially in sky-like areas) where as our CCDNet + ESRGAN output (Figure 8(b)) shows fewer artifacts. This makes it a more practical solution in scenarios where image clarity directly impacts decision-making (e.g., search and rescue).

Table 8
Evaluation of the images restored from different levels of quantization

	Original vs restored (16 colors)	Original vs restored (32 colors)	Original vs restored (64 colors)
Avg. PSNR	34.90 dB	39.94 dB	45.64 dB
Avg. SSIM	0.9419	0.9676	0.9895

Figure 7
Quality-compression trade-off of the proposed pipeline across quantization levels K=4 to K=64

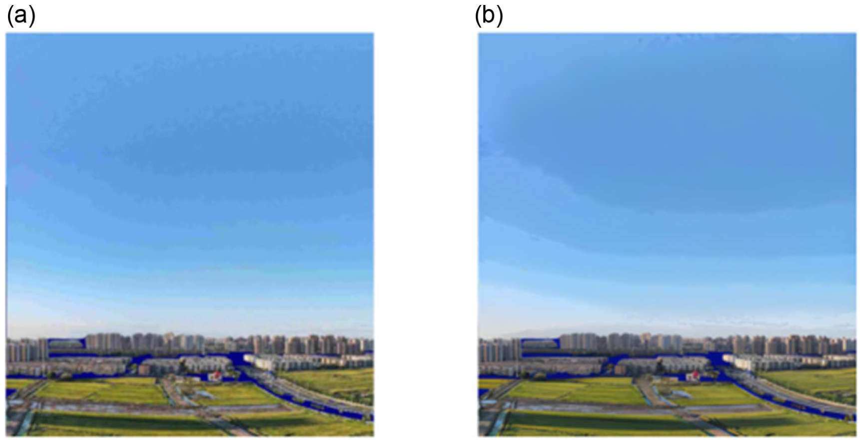


Note: Solid markers denote experimentally validated points; K=4 and K=8 are extrapolated.

Table 9
Performance comparison of super-resolution models on 16 color-quantized UAV images

Method	PSNR (dB)	SSIM
Real-ESRGAN	26.99	0.8888
SwinIR	27.27	0.8903
CCDNet + ESRGAN	35.01 dB	0.9430

Figure 8
(a) JPEG image, (b) proposed pipeline output – less visual artifacts are noticed in the outputs of our pipeline



4.1.4. Ablation study

A brief ablation study of the proposed pipeline is presented in Table 10 to show how the different components of the pipeline affect the metrics. The super-resolution model doesn't have substantial influence on performance metrics, but for the purpose of upscaling the image, removing visual artifacts, and better human visual perception the super-resolution model is important for the pipeline.

4.2. Downstream task evaluation

Image quality metrics (PSNR, SSIM) are useful for the evaluation of visual quality preservation; however, they do not directly quantify the usefulness of the pipeline for real operational UAV applications. We provide complete evaluation results that demonstrate the impact of our pipeline on an important downstream task object detection and validate with it, quality of

Table 10
Ablation study of the pipeline

	Metrics	Original vs restored from 16 colors	Original vs restored from 32 colors	Original vs restored from 64 colors
Deep learning model only	Avg. PSNR	34.90 dB	39.94 dB	45.64 dB
	Avg. SSIM	0.9419	0.9676	0.9895
Deep learning model withsuper-resolution	Avg. PSNR	35.01 dB	40.09 dB	45.65 dB
	Avg. SSIM	0.9430	0.9678	0.9899

content in restored images which is enough to use for practical purposes.

4.2.1. Global detection performance

The object detection experimental setup is summarized in Table 11.

Table 11
Object detection experimental setup

Parameter	Value
Dataset	Semantic Riverscapes
Total images	400
Total annotated objects	1105
Object classes	6
Detection model	YOLOv8n (nano)
Evaluation metric	mAP@0.5
Annotation format	COCO
Comparison sets	3
Compression ratio	~68%

Our pipeline attains a mAP (at 0.5 IoU threshold) of 63.0% on restored images and compare this to the 49.0% mAP on JPEG-compressed images and the 81.0% on original uncompressed images. This is significant 14.0 percentage point gain over JPEG compression with similar compression ratio (mean representing 68% file size reduction). Comprehensive global object detection performance metrics are presented in Table 12. The table presents mean Average Precision (mAP@0.5), detection counts, retention rates, and performance differences for Original, JPEG-compressed, and Restored image sets.

Table 12
Global object detection performance metrics

Metric	Original	JPEG (Q = 10)	Restored
mAP@0.5	81.0%	49.0%	63.0%
Δ vs Original	baseline	–32.0 pts	–18.0 pts
Objects detected	964	711	818
Retention rate	87.2%	64.3%	74.1%
False negatives	143	418	231
False negative rate	12.9%	37.8%	20.9%

Mean average precision comparison across image sets is visualized in Figure 9. Detection retention rates and preservation performance are illustrated in Figure 10.

4.2.2. Per-class performance analysis

When analyzing per-class, we observe that effectiveness of restoration is slightly different across various object classes; some classes are more benefited from restoration than others. The recovery pipeline brings particularly large gains for mission-critical categories. Per-class AP results are detailed in Table 13. It presents per-class AP (AP@0.5) for each operational category, comparing original, JPEG-compressed, and restored image sets, along with improvement metrics.

4.3. Cross-dataset generalization

We demonstrate the real-world deployment feasibility of our pipeline by evaluating it on new datasets without retraining. This zero-shot cross-dataset evaluation investigates whether restoration knowledge learned from natural/riverscape scenes can also transfer to distinct visual domains (e.g., urban scenes).

4.3.1. Image quality generalization

Zero-shot evaluation protocol and experimental configuration are summarized in Table 14.

On Semantic Riverscapes (training set), we obtain 35.01 dB PSNR and 0.9430 SSIM. For zero-shot evaluation on VisDrone, we obtain 31.49 dB PSNR and 0.8182 SSIM, achieving 89.94% and 86.8% retention ratios, respectively. These results demonstrate strong generalization with PSNR approaching 90% of training performance despite domain shift from natural/riverscape to urban scenes. Comprehensive cross-dataset image quality metrics are presented in Table 15. Cross-dataset image quality comparison is visualized in Figure 11.

4.3.2. Object detection generalization

In addition to image quality measures, we test whether the restoration performance would result in preserved detection effectiveness across domains. On Semantic Riverscapes (training set), restored images achieve 62.90% mAP. On VisDrone (zero-shot), we achieve 53.50% mAP, indicating 85.06% performance retention. Restoration improves results over JPEG for both datasets, with a 3.29 percentage point mAP advantage on VisDrone (53.50% vs. 50.21%). This demonstrates strong generalization ability without fine-tuning. Cross-dataset object detection performance comparison is detailed in Table 16. Cross-dataset object detection performance comparison is visualized in Figure 12.

Figure 9
Mean average precision (mAP@0.5) comparison across original, JPEG-compressed, and restored image sets, demonstrating the restoration pipeline's effectiveness in preserving detection performance

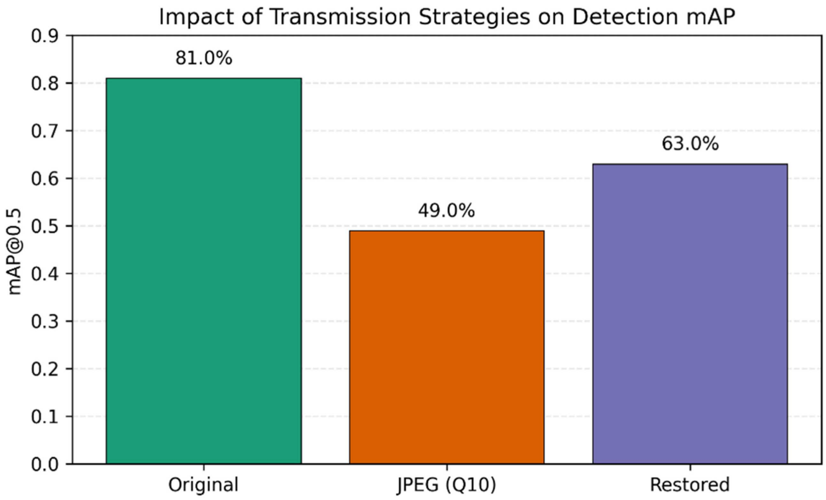


Figure 10
Detection retention rates showing the percentage of objects successfully detected for original, JPEG-compressed, and restored image sets, with visual representation of retained vs. missed detections

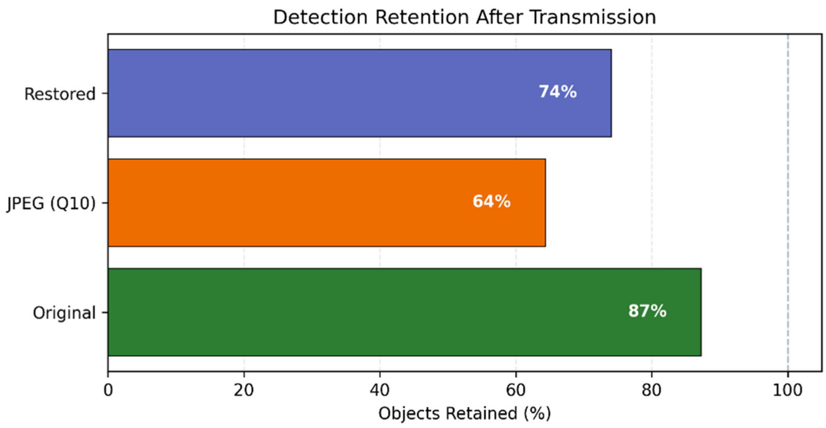


Table 13
Per-class average precision (AP@0.5)

Object Class	Original	JPEG (Q = 10)	Restored	Improvement vs JPEG
Car/Vehicle	88.0%	58.0%	69.0%	+11.0 pts
Human/Person	76.0%	37.0%	53.0%	+16.0 pts
Boat	69.0%	29.0%	44.0%	+15.0 pts
Building	82.0%	47.0%	63.0%	+16.0 pts
Tree	78.0%	44.0%	57.0%	+13.0 pts

4.4. Transmission and storage efficiency

Beyond image quality and downstream task performance, practical deployment of UAVs requires evaluation of transmission efficiency and storage capacity. This section presents network simulation and storage capacity analysis results demonstrating the practical benefits of compression for UAV deployment.

4.4.1. Transmission time analysis

Using the probabilistic network simulation framework described in Section 3.1.4, we evaluated transmission performance under realistic UAV communication conditions. Experiments showed that quantizing an image allows it to be transmitted 3× faster than transmitting the original uncompressed images in the same network environment. This improvement leads to

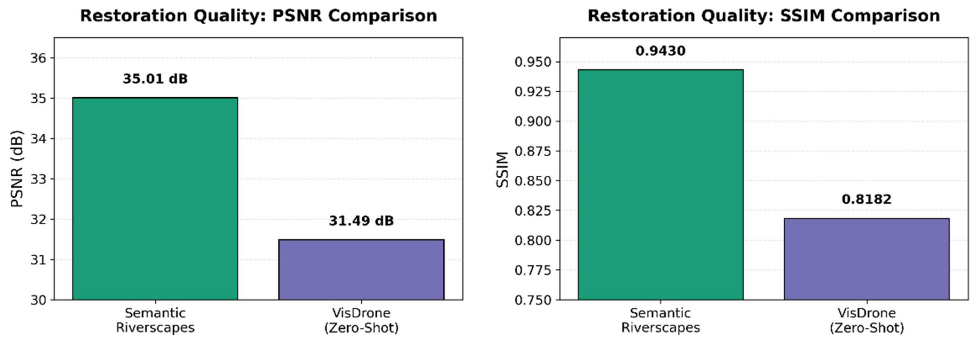
Table 14
Zero-shot evaluation setup

Parameter	Value	Description
Evaluation type	Zero-shot	No model retraining or adaptation
Training dataset	Semantic Riverscapes	Natural/riverscape scenes
Training images	400	Full benchmark dataset
Test dataset	VisDrone2019-DET	Urban/rural scenes
Test images	548	Validation set
Model adaptation	None	No fine-tuning or domain adaptation
Evaluation metrics	PSNR, SSIM, mAP	Image quality and downstream task
Image quality focus	Pixel-level restoration	PSNR and SSIM metrics
Downstream task focus	Semantic preservation	Object detection mAP
Domain shift	Natural → Urban	Different visual characteristics

Table 15
Cross-dataset image quality metrics

Metric	Semantic Riverscapes (Training)	VisDrone (Zero-Shot)	Absolute Difference	Performance Retention
PSNR (dB)	35.01	31.49	–3.52 dB	89.94%
SSIM	0.9430	0.8182	–0.1248	86.8%
Dataset Size	400 images	548 images	–	–
Domain Type	Natural/Riverscape	Urban/Rural	–	–

Figure 11
Cross-dataset image quality comparison showing PSNR and SSIM metrics for Semantic Riverscapes (training) and VisDrone (zero-shot), demonstrating generalization performance across domains



a low latency for image transmission from UAV to ground station, which is highly desirable in time-critical applications such as search and rescue. Figure 13 shows how transmission time drops for images after color quantization in the pipeline.

4.4.2. Storage capacity analysis

The capability to store images on board the UAV itself has a major influence on mission length and flexibility as well. We first studied onboard storage on a UAV with 32 GB of flash memory under which original uncompressed images were stored to judge the performance of image compression.

The results based on compressed images written suggest that as many as 2.5× more images can be stored than if the original uncompressed images are used for storage. More precisely, 12,870 original images could be stored (average of 2.6 MB per image) and 39,745 quantized images could be stored (average of 0.84 MB per image). This extended storage capability directly results in increased mission endurance without the need to offload data, larger coverage per mission flight, enhanced probability of collecting significant events over long operations, and reduced requirement for an active communication links.

4.4.3. Transmission efficiency summary

The combined benefits of compression and restoration create an effective balance between transmission efficiency and visual quality. The pipeline obtains 68% reduction in file size with color quantization, 3× faster transmission time under a simulated network, 2.5× increment of onboard image storage, and high-quality visual restoration (35.01 dB PSNR and 0.9430 SSIM). These results confirm the efficiency of the pipeline for bandwidth-constrained UAV applications where both transmission efficiency and image quality are concerned. Table 17 compares storage capacity, mission duration impact, and operational benefits

Table 16
Cross-dataset object detection performance

Metric	Semantic Riverscapes (Training)	VisDrone (Zero-Shot)
Original mAP@0.5	81.30%	73.20%
JPEG (Q=10) mAP@0.5	48.70%	50.21%
Restored mAP@0.5	62.90%	53.50%
Restored Retention (vs Original)	77.8%	73.1%
Restored vs JPEG Improvement	+14.2 pts	+3.29 pts
Original detections	964	698
Jpeg detections	711	590
Restored detections	818	560
Original retention rate	87.20%	78.50%
Jpeg retention rate	64.30%	67.10%
Restored retention rate	74.10%	62.40%
Performance retention ratio	-	85.06%

Figure 12
Cross-dataset object detection performance comparison showing mAP and retention rates for Semantic Riverscapes (training) and VisDrone (zero-shot) across different compression strategies

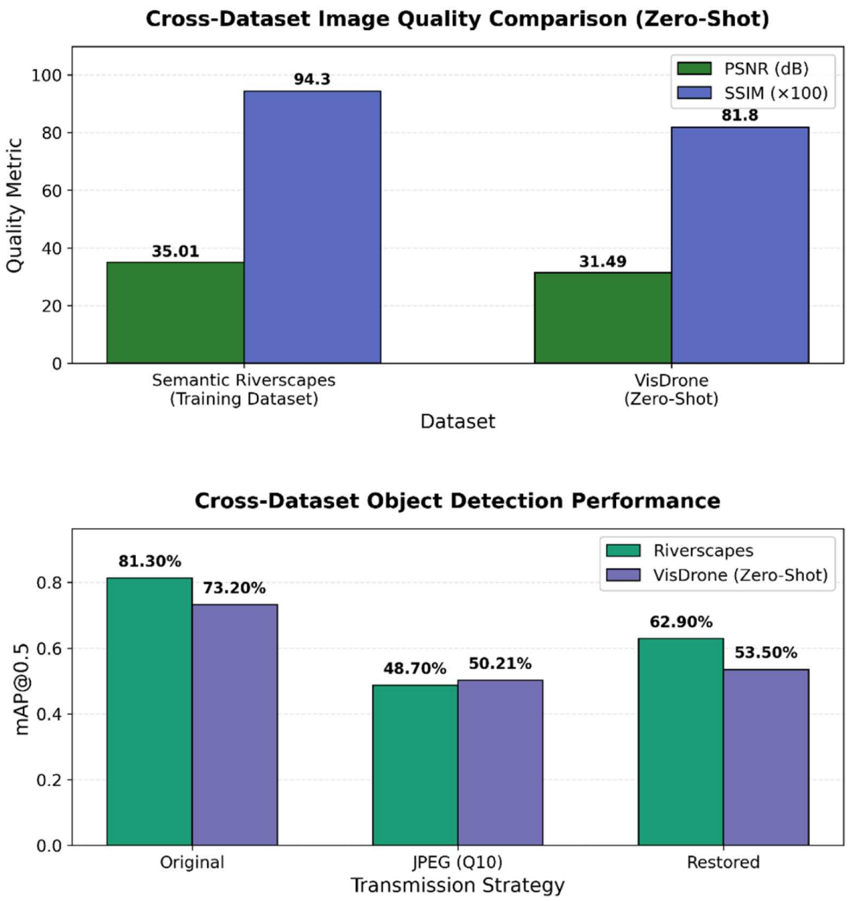


Figure 13
Transmission time for the original and quantized images

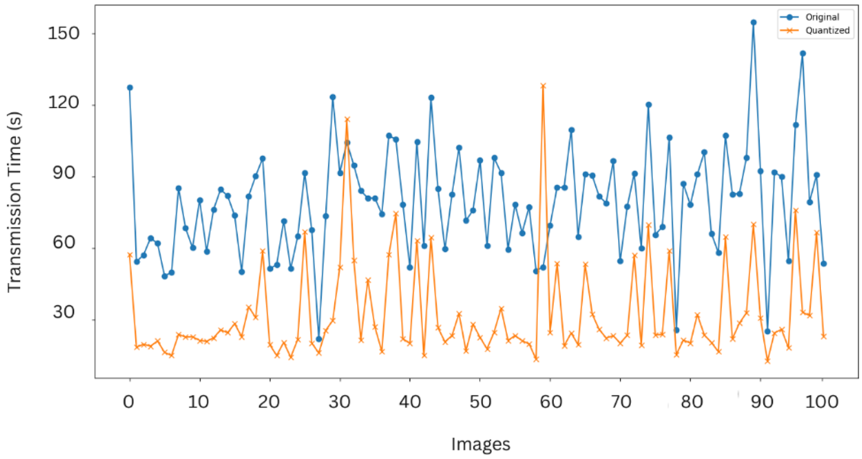


Table 17
Transmission efficiency summary

Metric	Original	Quantized	Improvement
Average Transmission Time	82.99 s	27.28 s	55.71 s or 3.08 times faster per image
Average File Size	2.6 MB	0.84 MB	68.1% smaller
Image Capacity (32 GB storage)	~12,807 images	~39,745 images	~26,938 additional images

between original and quantized image storage on a 32 GB UAV system.

4.5. Computational performance analysis

Although image quality and task performance are of paramount importance, real-world deployment of UAVs also demands validation that the pipeline operates within computational constraints compatible with embedded systems. This section presents comprehensive performance analysis in terms of runtime and memory usage.

4.5.1. Runtime performance

Real-time processing capability, critical for UAV applications, is primarily determined by runtime performance. For our training dataset, Semantic Riverscapes: 85.0 ms per image at the quantization stage, and 245.0 ms per image at restoration stage, which in total takes 330.0 ms per. In terms of runtime on VisDrone zero-shot evaluation dataset, the quantization takes 115.5 ms per image while restoration takes 257.2 ms per image, yielding 376.8 ms for entire pipeline per single image.

The restoration phase takes the majority of total runtime (74% on Riverscapes, 68% on VisDrone), as anticipated for deep learning inference. Quantization runtime takes 26% of total execution time for Riverscapes and 32% for VisDrone, where the latter has a higher fraction due to larger images that requires more processing time. Detailed runtime performance metrics for both datasets are presented in Figure 14 and in Table 18.

4.5.2. Memory usage

Memory constraints are an important consideration for embedded UAV platforms, which typically have limited RAM resources. On the training dataset, Semantic Riverscapes, the quantization and restoration stages require 512 MB and 1024

Figure 14
Runtime performance comparison across Semantic Riverscapes and VisDrone datasets, showing stage-wise breakdowns

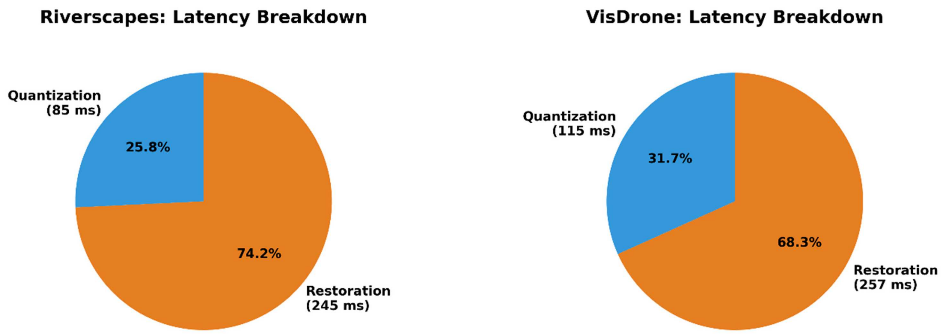
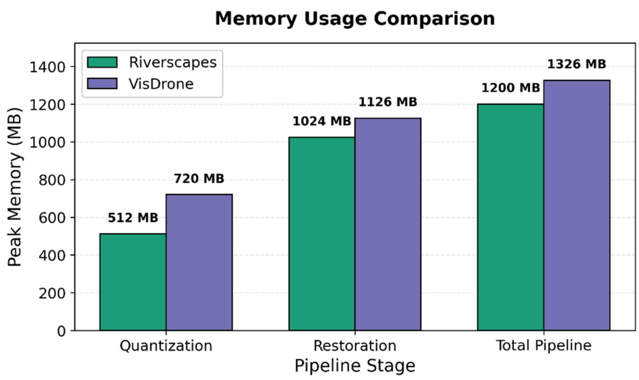


Table 18
Runtime performance metrics

Stage	Riverscapes (ms)	VisDrone (ms)	Ratio (VisDrone: Riverscapes)	% of Total (Riverscapes)	% of Total (VisDrone)
Quantization	85.0	115.5	1.41×	25.8%	31.7%
Restoration	245.0	257.2	1.05×	74.2%	68.3%
Total Pipeline	330.0	376.8	1.14×	–	–

Figure 15
Memory usage comparison between Semantic Riverscapes and VisDrone, showing stage-wise memory consumption and peak memory requirements



MB, leading to a peak memory of 1200 MB. On the zero-shot evaluation dataset, VisDrone, the quantization stage consumes 720 MB, restoration stage costs 1126 MB, and the peak memory is 1326 MB. Figure 15 shows memory usage comparison between the two datasets.

5. Conclusion

5.1. Summary of contributions

Our work has several important contributions to the field of UAV image transmission and enhancement. We demonstrate a three-stage strategy (onboard color quantization, high-efficiency transmission, and ground-station restoration), which leads to around 68% file size reduction with visually high quality (PSNR: 35.01 dB, SSIM: 0.9430), achieving approximately 3× faster of transmission under realistic network scenarios.

Apart from standard perceptual metrics, we expand our evaluation pipeline to evaluate performance on downstream tasks. We evaluate our restoration pipeline on object detection with YOLOv8 and obtain 63.0% mAP@0.5 vs 49.0% of equivalent compression factor JPEG-compressed images, representing a 14.0 percentage point advantage that preserves vital information in mission-critical applications.

We carry out zero-shot cross-dataset evaluation to validate the generalizability of our pipeline. Training only on Semantic Riverscapes and testing on VisDrone2019-DET without model fine-tuning shows that the restoration method learns generic image restoration principles, with 89.94% PSNR and 85.06% detection performance preserved across datasets.

5.2. Limitations

There are several limitations in the present study. The pipeline uses a fixed 16-color palette, regardless of scene complexity, which might be a limitation for images that have different degrees of color variation. The baseline comparison in this study is limited to JPEG compression; stronger baselines such as JPEG2000, BPG, and learned compression codecs were not evaluated, which limits the scope of the compression performance claims. Future work should include these comparisons to better contextualize the pipeline's compression efficiency. Additionally, the reported performance metrics are based on single experimental runs without statistical significance testing such as confidence intervals or repeated trials, which is a limitation of the current evaluation methodology. Cross-dataset evaluation is solely performed on two aerial imagery datasets; the performance on significantly different domains with drastic domain gaps (nighttime, infrared, extreme weather) has not been explored. Our evaluation is limited to one downstream task (object detection with YOLOv8); important other tasks like semantic segmentation or visual SLAM were not considered. Computational performance assessment is done on specific hardware, and real-world deployment would involve platform-specific tuning. Simulation of transmission even with realistic network models is a simulation, not one that has been field tested using actual UAVs as the platform. Evaluation is on still images; video sequences where temporal smoothness matters have not been handled. The restoration models could have biases from the training set, and might not handle image features that are less present in the data it has been trained on.

5.3. Future work

There are several directions for future research stemming from this work. An interesting improvement might be to make the color quantization strategies adaptive with the number of colors adjusted based on image complexity using lightweight metrics such as image entropy or edge density computed onboard, to compensate for one key limitation that could be a more effective representation. Further evaluation on more downstream tasks, such as semantic segmentation, instance segmentation, and visual SLAM could allow for a better utility estimation. Studying task-aware compression approaches where parameters are optimized for the downstream task performance could lead to more effective models. Reconsidering more advanced restoration architectures (vision transformers, GANs) might be beneficial quality-wise, but take into account that computational constraints remain important. Customizing hardware-specific optimizations such as model pruning or FPGA acceleration would increase the deployment feasibility. Real UAV platforms would validate a complete field test under real-world conditions. Studying the performance of video sequences would also generalize this to real-time streaming scenarios. The performance of specialized scenarios could be improved by developing domain-specific optimizations for particular applications (search and rescue, infrastructure inspection). Investigating federated learning techniques can potentially facilitate collaborative enhancement of models while maintaining data privacy. Studying the integration with edge-computing architectures will permit hybrid concepts that balance onboard and ground-station processing.

In summary, this paper provides a complete pipeline for data-efficient image transmission and enhancement in the context of UAV applications. We experimentally validate the pipeline and show its ability to achieve significant improvements in transmission efficiency and image quality while also being computationally amenable to real-time deployment. The proven enhancements in transmission efficiency and computational feasibility, combined with the ability to preserve semantic information that other approaches might blunt or remove, make this pipeline a valid approach for bandwidth-limited UAV applications where image quality and information preservation are critical. Though the work has its limitations and future research needs to be done, the contributions enhanced the UAV image transmission technology extent, which serves as a solid foundation for further exploration and practical application.

Ethical Statement

This study does not contain any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

The data that support the findings of this study are openly available. The Semantic Riverscapes dataset is available in Landscape and Urban Planning at https://doi.org/10.1016/j.landurbplan.2022.104569, in reference [39], and the VisDrone2019-DET dataset is available in the 2019 IEEE/CVF ICCV Workshop Proceedings at https://doi.org/10.1109/ICCVW.2019.00030, in reference [27].

Author Contribution Statement

F. M Abir Hossain: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization. **Noman Saffat Sajid:** Conceptualization, Methodology, Software, Investigation, Writing – original draft. **Rashedur M. Rahman:** Supervision, Project administration.

References

[1] Hossain, F. M. A., Sajid, N. S., & Rahman, R. M. (2025). Data-efficient image transmission and enhancement for UAVs: A color quantization and super-resolution approach. In *Data Science, AI and Applications: First International Conference*, 389–402. https://doi.org/10.1007/978-3-032-11355-9_27

[2] Minnen, D., Ballé, J., & Toderici, G. (2018). Joint autoregressive and hierarchical priors for learned image compression. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 10794–10803.

[3] Xiao, E. (2024). Comprehensive k-means clustering. *Journal of Computer and Communications*, *12*(03), 146–159. https://doi.org/10.4236/jcc.2024.123009

[4] Abernathy, A., & Celebi, M. E. (2022). The incremental online k-means clustering algorithm and its application to color quantization. *Expert Systems with Applications*, *207*, 117927. https://doi.org/10.1016/j.eswa2202.117927

[5] Jaques, L. E., Depoian, II., C, A., Murrell, E., Xie, D., Bailey, C. P., & Guturu, P. (2022). Novel L1 PCA informed k-means color quantization. In *Big Data IV: Learning, Analytics, and Applications*, *12097*, 151–157. https://doi.org/10.1117/12.2619139

[6] Lei, Z., Hong, X., Shi, J., Su, M., Lin, C., & Xia, W. (2023). Quantization-based adaptive deep image compression using semantic information. *IEEE Access*, *11*, 118061–118077. https://doi.org/10.1109/ACCESS.2023.3326718

[7] van den Oord, A., Vinyals, O., & Kavukcuoglu, K. (2017). Neural discrete representation learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 6309–6318.

[8] Cho, S.-J., Ji, S.-W., Hong, J.-P., Jung, S.-W., & Ko, S.-J. (2021). Rethinking coarse-to-fine approach in single image deblurring. In *2021 IEEE/CVF International Conference on Computer Vision*, 4621–4630. https://doi.org/10.1109/ICCV48922.2021.00460

[9] Liu, J., Malekzadeh, M., Mirian, N., Song, T.-A., Liu, C., & Dutta, J. (2021). Artificial intelligence-based image enhancement in pet imaging. *PET Clinics*, *16*(4), 553–576. https://doi.org/10.1016/j.cpet.2021.06.005

[10] Ullah, F., Kumar, K., Rahim, T., Khan, J., & Jung, Y. (2025). A new hybrid image denoising algorithm using adaptive and modified decisionbased filters for enhanced image quality. *Scientific Reports*, *15*(1), 8971. https://doi.org/10.1038/s41598-025-92283-3

[11] Lepcha, D. C., Goyal, B., Dogra, A., & Goyal, V. (2023). Image super-resolution: A comprehensive review, recent trends, challenges and applications. *Information Fusion*, *91*, 230–260. https://doi.org/10.1016/j.inffus.2022.10.007

[12] Wang, X., Xie, L., Dong, C., & Shan, Y. (2021). Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data. *2021 IEEE/CVF International Conference on Computer Vision Workshops*, 1905–1914. https://doi.org/10.1109/ICCVW54120.2021.00217

[13] Qin, Z., Zeng, Q., Zong, Y., & Xu, F. (2021). Image inpainting based on deep learning: A review. *Displays*, *69*, 102028. https://doi.org/10.1016/j.displa.2021.102028

[14] Žeger, I., Grgic, S., Vuković, J., & Šišul, G. (2021). Grayscale image colorization methods: Overview and evaluation. *IEEE Access*, *9*, 113326–113346. https://doi.org/10.1109/ACCESS.2021.3104515

[15] Chen, X., Liu, Y., Zhang, Z., Qiao, Y., & Dong, C. (2021). HDRUNet: Single image HDR reconstruction with denoising and dequantization. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 354–363. https://doi.org/10.1109/CVPRW53098.2021.00045

[16] Fu, Z., Peng, C., Huang, X., Xia, M., Xu, J., & Li, X. (2024). Bit depth enhancement method based on visual contrast perception features. *Engineering Applications of Artificial Intelligence*, *136*, 108838. https://doi.org/10.1016/j.engappai.2024.108838

[17] Keysers, D., Lampert, C., & Breuel, T. (2006). Color image dequantization by constrained diffusion. In *Color Imaging XI: Processing, Hardcopy, and Applications: Proceedings of SPIE*, *6058*, 605803. https://doi.org/10.1117/12.648713

[18] Dudhane, A., Zamir, S. W., Khan, S., Khan, F. S., & Yang, M.-H. (2023). Burstormer: Burst image restoration and enhancement transformer. In *CVF Conference on Computer Vision and Pattern Recognition*, 5703–5712. https://doi.org/10.1109/CVPR52729.2023.00552

[19] Wang, Y., Huang, H., Wang, C., He, T., Wang, J., & Hoai, M. (2019). Gif2video: Color dequantization and temporal interpolation of GIF images. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1419–1428. https://doi.org/10.1109/CVPR.2019.00151

[20] Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., . . . , & Qiao, Y. (2019). ESRGAN: Enhanced super-resolution generative adversarial networks. In *Computer Vision – ECCV 2018 Workshops* (pp. 63–79). https://doi.org/10.1007/978-3-030-11021-5_5

[21] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10674–10685. https://doi.org/10.1109/CVPR52688.2022.01042

[22] Agustsson, E., Tschannen, M., Mentzer, F., Timofte, R., & van Gool, L. (2019). Generative adversarial networks for extreme learned image compression. In *2019 IEEE/CVF International Conference on Computer Vision*, 221–231. https://doi.org/10.1109/ICCV.2019.00031

[23] Hoshikawa, L., Conde, M. V., Ohashi, T., & Irie, A. (2025). Extreme compression of adaptive neural images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 4042–4052.

[24] Gandor, T., & Nalepa, J. (2022). First gradually, then suddenly: Understanding the impact of image compression on object detection using deep learning. *Sensors*, *22*(3), 1104. https://doi.org/10.3390/s22031104

[25] Lu, G., Ge, X., Zhong, T., Hu, Q., & Geng, J. (2024). Preprocessing enhanced image compression for machine vision. *IEEE Transactions on Circuits and Systems for Video Technology*, *34*(12), 13556–13568. https://doi.org/10.1109/TCSVT.2024.3441049

[26] Tang, Z., Yi, X., & Wang, H. (2025). Toward learned image compression for multiple semantic analysis tasks. *IEEE Transactions on Broadcasting*, *71*(4), 1022–1033. https://doi.org/10.1109/TBC.2025.3597154

[27] Du, D., Zhu, P., Wen, L., Bian, X., Lin, H., Hu, Q., . . . , & Z, , Liu. (2019). VisDrone-DET2019: The vision meets drone object detection in image challenge results. In *2019 IEEE/CVF International Conference on Computer Vision Workshop*, 213–226. https://doi.org/10.1109/ICCVW.2019.00030

[28] Rodriguez, A. L., & Mikolajczyk, K. (2019). *Domain adaptation for object detection via style consistency*. arXiv. https://doi.org/10.48550/ARXIV.1911.10033

[29] Li, W., & Zhang, Y. (2025). Enhanced small object detection in UAV aerial imagery through attention gated backbone and context aware fusion. *Scientific Reports*, *16*(1), 2366. https://doi.org/10.1038/s41598-025-32074-y

[30] Cong, R., Pan, Z., & Wang, H. (2025). Cross-domain landscape restoration and fusion based on hybrid deep learning model. In *Smart Wireless Communications: DL-based Methods and Algorithms: Proceedings of the Third International Conference on Artificial Intelligence and Communication Technologies*, *1*, 126–136. https://doi.org/10.1007/978-3-032-06372-4_11

[31] Ramachandran, A., & Sangaiah, A. K. (2021). A review on object detection in unmanned aerial vehicle surveillance. *International Journal of Cognitive Computing in Engineering*, *2*, 215–228. https://doi.org/10.1016/j.ijcce.2021.11.005

[32] Meimetis, D., Daramouskas, I., Patrinopoulou, N., Lappas, V., & Kostopoulos, V. (2025). Comparative analysis of object detection models for edge devices in UAV swarms. *Machines*, *13*(8), 684. https://doi.org/10.3390/machines13080684

[33] Hussain, M. (2024). YOLOv1 to v8: Unveiling each variant—a comprehensive review of YOLO. *IEEE Access*, *12*, 42816–42833. https://doi.org/10.1109/ACCESS.2024.3378568

[34] Zhu, W., & Chen, K. (2026). Real-time object detection for unmanned aerial vehicles based on vision transformer and edge computing. *Scientific Reports*, *16*, 6814. https://doi.org/10.1038/s41598-026-37938-5

[35] Shokraneh, S., Lighvan, M. Z., & Asadpour, M. (2024). Enhancing k-means clustering efficiency on FPGA: A three-design approach to parallel implementation. In *2024 IEEE East-West Design & Test Symposium* (pp. 1–5). https://doi.org/10.1109/EWDTS63723.2024.10873792

[36] Singh, S., Sharma, K., Karna, B. K., & Raj, P. (2022). Pruning and quantization for deeper artificial intelligence (AI) model optimization. In *International Conference on Robotics, Control, Automation and Artificial Intelligence*, 933–945. https://doi.org/10.1007/978-981-99-4634-1_73

[37] Pérez, J., Díaz, J., Berrocal, J., López-Viana, R., & González-Prieto, Á. (2022). Edge computing. *Computing*, *104*(12), 2711–2747. https://doi.org/10.1007/s00607-022-01104-2

[38] Huang, B., Abtahi, A., & Aminifar, A. (2024). Energy-aware integrated neural architecture search and partitioning for distributed Internet of Things (IoT). *IEEE Transactions on Circuits and Systems for Artificial Intelligence*, *1*(2), 257–271. https://doi.org/10.1109/TCASAI.2024.3493036

[39] Luo, J., Zhao, T., Cao, L., & Biljecki, F. (2022). Semantic Riverscapes: Perception and evaluation of linear landscapes from oblique imagery using computer vision. *Landscape and Urban Planning*, *228*, 104569. https://doi.org/10.1016/j.landurbplan.2022.104569

[40] Liang, J., Cao, J., Sun, G., Zhang, K., van Gool, L., & Timofte, R. (2021). SwinIR: Image restoration using Swin Transformer. In *2021 IEEE/CVF International Conference on Computer Vision Workshops*, 1833–1844. https://doi.org/10.1109/ICCVW54120.2021.00210

[41] Basioti, K., & Moustakides, G. V. (2021). Single image restoration with generative priors. In *2021 IEEE International Conference on Image Processing*, 1679–1683. https://doi.org/10.1109/ICIP42928.2021.9506768

[42] Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2021). Scaled-YOLOv4: Scaling cross stage partial network. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13024–13033. https://doi.org/10.1109/CVPR46437.2021.01283

How to Cite: Hossain, F. M. A., Sajid, N. S., & Rahman, R. M. (2026). Semantic-Preserving Image Compression and Restoration Pipeline for Bandwidth-Constrained UAV Applications: A Task-Aware Evaluation Framework. *Artificial Intelligence and Applications*. https://doi.org/10.47852/bonviewAIA62028610