Less is More: CORSA's Edge Foundation Model for Constrained AI

Xenia Ivashkovych
Vito Remote Sensing
Antwerp, Belgium
xenia.ivashkovych@vito.be

Dirk Nuyts
Vito Remote Sensing
Antwerp, Belgium
dirk.nuyts@vito.be

Bart Beusen
Vito Remote Sensing
Antwerp, Belgium
bart.beusen@vito.be

Nick Witvrouwen
Vito Remote Sensing
Antwerp, Belgium
nick.witvrouwen@vito.be

Tanja Van Achteren
Vito Remote Sensing
Antwerp, Belgium
tanja.vanachteren@vito.be

***Abstract*— This paper presents the deployment of the CORSA edge foundation model on the Jetson Orin NX. The CORSA encoder and the application-specific decoder are implemented as separate components, although the decoder depends on the encoder for input representations. This modular design enables greater flexibility. The on-board system is capable of both processing sensor data into high-level products and generating compressed representations in real time.**

***Keywords*—artificial intelligence, remote sensing, compression, edge processing, Nvidia Jetson, foundation model, hyperspectral, multispectral, vector quantized variational auto-encoder**

I. Introduction (*Heading 1*)

The discrepancy between data production and downlink capabilities in space has been a well-known issue for decades. The gap is particularly challenging for missions with high spatial or spectral resolutions, such as IPERLITE, which can acquire up to 5 TB of data per day, of which only 20 Gb can be downlinked.

Despite a strong preference for lossless compression methods, even the academic remote sensing community has shown a willingness to consider lossy compression techniques due to this bottleneck. As a result, near-lossless methods, which preserve the full quality of raw images without maintaining exact bit-level fidelity, have gained significant attention.

In this technical paper, we present a deep learning (DL) model which functions not only as a near-lossless compression solution but also as an edge foundation model. This also enables efficient fine-tuning for mission-specific tasks, allowing satellites to prioritise and downlink only the most relevant data.

II. Solution

A. Architecture

Our deep learning solution, CORSA [1], is a hierarchical vector quantized variational auto-encoder (HVQVAE) trained through self-supervised learning on a reconstruction task. A general outline of the solution's architecture is illustrated in Fig 1.

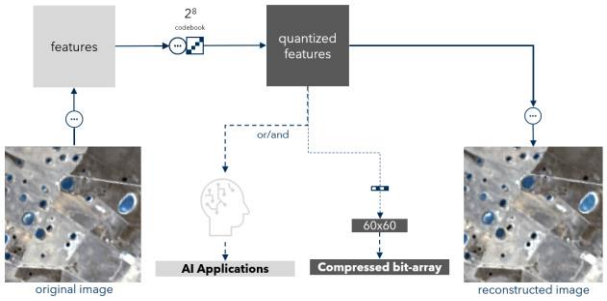

Fig. 1. CORSA model blueprint

The most widely used version of CORSA is a two-level HQVAE with a convolutional backbone. Variants of CORSA include SwinFormer backbones, three-level HVQVAEs, multi-modal fusion backbones, and masked auto-encoder training. The edge foundation model's compression power and semantic expressivity lie in its codebooks, which are a consistent feature of the deep learning model across all versions.

During training, the DL model interprets an input image as an abstract ensemble of features, which are subsequently mapped onto a set of discrete vectors known as the *codebook*. This codebook evolves during self-supervised training and is therefore not universal.

B. Compression

CORSA's compression capabilities derive directly from the existence of the codebook. Once training is complete, the model's codebook is fixed, and each feature vector it contains is associated with a unique index. These indices can be stored as a 4-bit, 6-bit, 8-bit, or 12-bit integer, depending on the codebook's size. Compression is achieved by transforming an image into a collection of feature vectors, which are then mapped to the static codebook and represented by corresponding indices. This collection of indices constitutes the compressed representation of the image.

These indices can be subsequently mapped back to the corresponding feature vectors. A separate model, the CORSA decoder, uses these vectors to reconstruct the output image, as illustrated in Fig. 2.

Fig. 2. Input (top) and output (bottom) of a CORSA model trained on APEX simulated data (10m spatial resolution and 10nm spectral resolution) in the visible spectrum (pseudo-RGB colours)

We extend our thanks to ESA for sponsoring the Smart-Connect project for advanced space technology for disaster response, as part of their Civil Security from Space programme, without whom the research necessary to refine CORSA would not have been possible.

Like all DL methods, CORSA exploits patterns and complex correlations in the data to achieve compression. Consequently, its effectiveness depends on the nature of the data compressed. The near-lossless compression ratio, defined as the ratio at which the reconstruction quality remains visually indistinguishable from the original, increases as spatial resolution decreases and spectral resolution increases. As a result, the edge foundation model demonstrated its strongest compression performance when applied to hyperspectral data, as shown in Table 1.

TABLE I. CORSA MODELS

Sensor	*Spatial res.* [a]	*CR* [b]	*PSNR* [e]	*PSF* [f]	*SSIM* [g]	*SAM* [e]
S-2[c] (RGB-NIR)	10m	20	50	N/A	0.99	N/A
S-2[c]	10-30m	~30	50	N/A	0.99	N/A
ENMAP	30m	300	50	0.36	0.98	0.99
APEX Sim.[d]	10m	80	48	0.39	0.98	0.99

[a.] Stands for resolution. Spectral resolution is poorly applicable to multispectral data

[b.] Compression Ratio

[c.] S-2 stands for Sentinel-2

[d.] APEX Sim. refers to data simulated for the MOVIQ project, financed by VLAIO, by resampling the airborne APEX hyperspectral imagery to simulate the imagery produced by a hyperspectral satellite camera

[e.] Peak Signal-to-Noise Ratio relative to a 14-bit signal, for the exception of the Apex Simulated data

[f.] Point-Spread Function: see Annex A for additional information

[g.] Structural Similarity Index Measure: used to estimate the semantic quality of an image

[h.] Spectral Angle Mapping: estimates the quality of reconstructed spectral signature for a given pixel, only applicable to hyperspectral imagery

It is important to note that, unlike classical near-lossless compression techniques, which impose a strict upper bound on the reconstruction error, our statistical approach minimizes the average error over the target distribution. [see Annex B]

Although CORSA models are sensor-specific, knowledge acquired from one sensor can be transferred for another with minimal effort. For example, the compression algorithm trained on ENMAP hyperspectral data has been successfully transferred to PRISMA data, as illustrated in Fig. 3. and Fig. 5. Even one-shot transfer yields interesting, though imperfect, results, as seen in Fig. 4.

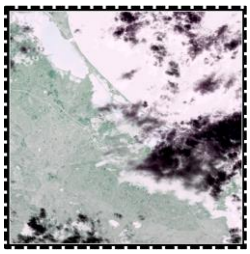

Fig. 3. PRISMA imagery in false-RGB colours

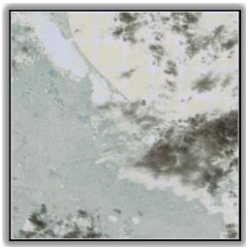

Fig. 4. Reconstruction of the input of Fig. 3. by a CORSA model trained on ENMAP without additional training

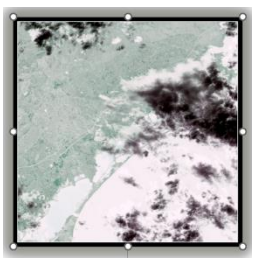

Fig. 5. Ouput of a CORSA model trained on ENMAP and finetuned on less than 10 images of PRISMA

CORSA's cross-sensor transferability may be interpreted as a consequence of the model developing a high-level representation of remote sensing imagery. Although ENMAP and PRISMA differ in spectral resolution, they share the same spatial resolution. CORSA is able to recognize that hyperspectral imagery appears semantically similar when captured from the same altitude. The difference in spectral resolution accounts for the colour inaccuracies seen in output shown in Fig. 4. These are corrected through minimal fine-tuning, as demonstrated in Fig. 5.

C. Downstream applications

CORSA's capabilities as a foundation model have been demonstrated on a range of downstream applications, including super-resolved parcel delineation, flood detection, change detection, and landcover classification. Further details on the specific architectures and performances of the downstream applications are available in the dedicated paper by B. Beusen and A. Luyts [2].

These applications are typically not a fine-tuned versions of the CORSA autoencoder, but rather new decoder models designed to process compressed representation as input for task-specific computations. The architectures and sizes of these decoder components vary according to the requirements of the application, although the overall configuration remains consistent.

This setup enables the simultaneous execution of both compression and high-level data processing on imagery. Multiple downstream applications can also be run in parallel using a single CORSA encoder to extract semantically meaningful representations.

III. HARDWARE IMPLEMENTATION

No compression solution designed for satellites can remain confined to a laptop in a research lab. Accordingly, most recent efforts have focused on deploying existing models on space-grace, or the very least radiation-tolerant, hardware. The Nvidia Jetson Orin NX 16GB quickly emerged as a popular hardware solution for on-board deep learning algorithms. CORSA for ENMAP has been successfully deployed on the Jetson [3], as well as on the Hailo 8 and 8L [4].

To further showcase the versatility of our edge foundation model, we have implemented an CORSA-based downstream application on the Jetson. This includes the CORSA encoder, which also serves as the compression model, and the decoder head tailored to a specific application. This configuration enables simultaneous data compression and high-level processing. Such flexibility is particularly valuable in low-certainty detection scenarios, where both the detection output and the compressed imagery can be downlinked together.

The implementation blueprint for both the CORSA encoder and the application-specific decoder is reusable across any application developed within this framework. At a minimum, deployment requires the model to be stripped of auxiliary training components, converted to ONNX format, and optimised using the TensorRT framework.

One of the most critical aspects of on-board implementation is energy constraint. On Earth, the energy consumption of a deep learning model is rarely a primary concern, except when environmental impact becomes significant. At the edge, however, space, memory, and

available energy are fundamental considerations. The Jetson provides several predefined power profiles for testing in different conditions. However, their labels (10W, 15W, and 15W) are somewhat misleading, as they do not strictly limit the Jetson's power consumption. Instead, they affect the number of active CPU cores and the maximum frequencies of various components, as detailed in Table II.

TABLE II. DEFAULT JETPACK POWER PROFILES

Power Budget	***10W***	***15W***	***25W***
CPU ONLINE	4	4	8
CPU MAX FREQ	1190.4 MHz	1420.8 MHz	1497.6 MHz
GPU MAX FREQ	612 MHz	612 MHz	408 MHz

As previously mentioned, the actual power consumption of the Jetson during computation is not bounded by the nominal values of its power profile. Table III presents the model's throughput, energy consumption, and overall efficiency, averaged over multiple runs. These values represent upper estimates, as the tests did not explore the maximum batch size supported by the Jetson, which would likely result in increased throughput and improved efficiency.

At the time of writing, a land cover classification model has been ported to the Jetson platform without any loss in accuracy, as expected. The CORSA S-2 encoder (10 bands) was used to generate compressed representations, which were then passed to an application-specific decoder. In principle, if only the end application were required, without the need to downlink the compressed data, the quantisation step could be omitted entirely, thereby increasing inference speed. However, the tests were conducted under worst-case assumptions, where limited memory and computational resources, due to the presence of other critical applications, prevent the execution of both operations in quick succession. The corresponding performance figures are presented in Table III.

TABLE III. HARDWARE PERFORMANCE

	Encoding + Processing		
	10W[i]	***15W***[i]	***25W***[i]
Average time (ms)	5.05	5.3	4.6
Throughput (MP/ms)	3.24	3.24	3.82
Energy consumption[j] (W)	10.2	11.0	11.1
Efficiency (MP/ms/W)	0.319	0.294	0.344

[i.] Default power profile on the Jetson, not directly indicative of power consumption, see Table II

[j.] The real energy consumption measured on the Jetson

The data in Table III reveal a noteworthy and perhaps counterintuitive pattern: a more powerful power profile does not necessarily yield greater efficiency. In addition, power consumption remains relatively stable across all profiles. These findings suggest the presence of a computational bottleneck, which is suspected to be the argmax operation used during codebook mapping and reverse mapping.

IV. CONCLUSION AND DISCUSSION

In this paper, we have demonstrated the deployment of a CORSA encoder alongside an application-specific decoder. Even under worst-case conditions (a batch size of one, with compression followed by downstream data processing) the system achieves high efficiency, placing it firmly within the domain of real-time applications.

We have also identified a bottleneck in the deployment pipeline: the argmax operation. This bottleneck has redirected our research focus away from model distillation and pruning towards addressing the so-called 'argmax problem'. The Jetson platform is sufficiently powerful to handle deep learning models with ease, surpassing even the performance benchmarks described in X's paper. As a general observation, a CORSA encoder trained on hyperspectral imagery is significantly more computationally intensive than one trained on multispectral imagery. In our deployment, the application-specific decoder remains relatively lightweight compared to the ENMAP-trained encoder. The reduced size of the model has made the bottleneck imposed by the argmax operation even more pronounced.

Notably, the codebook mapping process cannot be omitted, as doing so would eliminate the model's compression capabilities, an essential feature for on-board operations.

Several approaches to mitigating this bottleneck are currently under investigation. One potential solution involves replacing the argmax operation with an alternative mechanism, although the expected performance gains from this substitution are likely to be minimal. A second approach involves modifying power profiles and reconfiguring programmable hardware to allocate greater computational capacity to the argmax operation; this is currently the most promising direction. Additionally, the team has recently begun exploring alternative training and mapping strategies that could eliminate the need for the argmax operation entirely.

REFERENCES

[1] B. Beusen, X. Ivashkovych, and T. Achteren, "Image compression using vector-quantized auto-encoders with semantically meaningful feature extraction," Small Satellites Systems and Services (4S) Symposium, Villamoura, Portugal, 16-20 May 2022.

[2] B. Beusen, A. Luyts, X. Ivashkovych and T. Van Achteren, "Lightweight and Efficient: A Family of Multimodal Earth Observation Foundation Models," IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium, Athens, Greece, 2024, pp. 2841-2846, doi: 10.1109/IGARSS53475.2024.10641132.

[3] Ivashkovych, X., Witvrouwen, N., Beusen, B., Luyts, A., & Van Achteren, T., "Implementation of CORSA on Nvidia Jetson Hardware", Workshop on On-Board Payload Data Compression 2024 (OBPDC 2024), Gran Canaria, Spain, 30 September 2024, https://doi.org/10.5281/zenodo.13863200

[4] Witvrouwen, N., Nuyts, D., Ivashkovych, X., Beusen, B., Persson, A., & Van Achteren, T., "Towards real-time edge EO foundational models: CORSA on Hailo AI accelerator", European Data Handling & Data Processing Conference (EDHPC 2025), Elche, Spain, 13-17 October 2025, unpublished

ANNEXES

A. *Point-Spread-Function in a non-optical system*

The Point Spread Function (PSF) describes how an imaging system responds to a point source of light or radiation. It was originally developed for use in optical systems. However, with the emergence of image

reconstruction techniques, the PSF has also come to serve as a measure of image sharpness, or conversely, its blurriness.

The PSF used to assess the reconstruction quality of the models in this technical paper is not the traditional physical PSF, but rather an approximation. Specifically, it is calculated as the average energy contained in the peaks of the two-dimensional pseudo-point spread functions for each spectral channel. These pseudo-PSFs are derived from the inputs and outputs of the models, approximating the system's response if it were a conventional optical device.

Given the consistently high SSIM scores achieved by our models, this PSF approximation is used both as an indicator of image sharpness and as a proxy for the visual quality of the reconstructed outputs.

B. CORSA vs CCSDS 123.0-b-2/3

A clear illustration of the differences between classical and statistical compression methods can be found in the comparison with the CCSDS 123.0-B-2/3 algorithms, which include both lossless and near-lossless variants. These are considered the gold standard for hyperspectral compression at the time of writing. Unlike the CORSA encoder, the CCSDS algorithm does not exploit intra-channel correlations, or at least not to the same extent as CORSA. This results in more predictable error estimates, but prevents the CCSDS approach from achieving the same compression ratios as CORSA, as shown in Fig. 6.

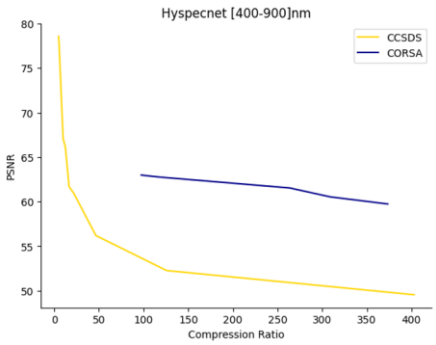

Fig. 6. Evolution of the reconstruction accuracy (measured by the PSNR) against the compression ratio for the CCSDS 123.0-b algorithm and CORSA model on ENMAP data

Deep learning or artificial intelligence-based compression algorithms cannot be evaluated using the same criteria as classical methods. Guaranteeing an upper bound on reconstruction error typically comes at the cost of exploiting the more complex correlations present in the data.

For this reason, fallback mechanisms such as lossless algorithms and out-of-distribution detection are essential for the operational deployment of models like CORSA.