DUNIA: Pixel-Sized Embeddings via Cross-Modal Alignment for Earth Observation Applications

Ibrahim Fayad [1 2] **Max Zimmer** [3] **Martin Schwartz** [1] **Fabian Gieseke** [4] **Philippe Ciais** [1] **Gabriel Belouze** [1] **Sarah Brood** [5] **Aurelien De Truchis** [2] **Alexandre d'Aspremont** [5 2]

Abstract

Significant efforts have been directed towards adapting self-supervised multimodal learning for Earth observation applications. However, most current methods produce coarse patch-sized embeddings, limiting their effectiveness and integration with other modalities like LiDAR. To close this gap, we present DUNIA, an approach to learn pixel-sized embeddings through cross-modal alignment between images and full-waveform LiDAR data. As the model is trained in a contrastive manner, the embeddings can be directly leveraged in the context of a variety of environmental monitoring tasks in a zero-shot setting. In our experiments, we demonstrate the effectiveness of the embeddings for seven such tasks: canopy height mapping, fractional canopy cover, land cover mapping, tree species identification, plant area index, crop type classification, and per-pixel waveform-based vertical structure mapping. The results show that the embeddings, along with zero-shot classifiers, often outperform specialized supervised models, even in low-data regimes. In the fine-tuning setting, we show strong performances near or better than the state-of-the-art on five out of six tasks.

1. Introduction

With the rapid expansion of Earth Observation (EO) satellite missions, deep learning has emerged as a powerful solution for key EO applications, ranging from monitoring natural resources (Chen et al., 2023; Fayad et al., 2024; Li et al., 2023; Liu et al., 2022; 2023; Tolan et al., 2024; Lang et al., 2023; Pauls et al., 2024; Schwartz et al., 2023; Pauls et al., 2025) and assessing environmental impacts (Dalagnol et al., 2023; Wagner et al., 2023), to climate monitoring and forecast (Andrychowicz et al., 2023; Schultz et al., 2021), as well as agricultural and land resource management (Ienco et al., 2019; Kussul et al., 2017; Zhang et al., 2019).

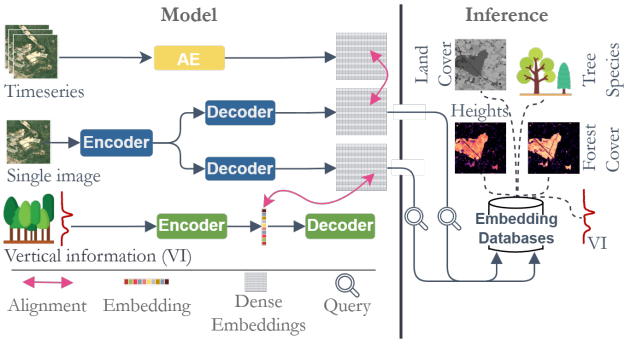

Figure 1. DUNIA's alignment strategy leverages temporal and vertical information at the pixel level, enabling the model to develop a cross-modal understanding from a single satellite image. This approach serves as a foundation for cross-modal retrieval tasks across diverse EO applications. The pre-trained model is represented in navy blue, the multi-temporal autoencoder (AE) in yellow, and the waveform AE in green.

While research of new deep learning architectures and upcoming remote satellite missions is expected to enhance existing EO applications, supervised deep learning approaches have key limitations. First, these models depend on labeled data, often requiring large annotated datasets. For example, the estimation of above-ground biomass (AGB) still heavily relies on traditional methods due to limited ground-based measurements and the need to use other variables as proxies (Morin et al., 2023; Schwartz et al., 2023).

Second, EO models are often tailored to specific tasks, limiting their adaptability and reusability, even when using similar input data sources. For instance, a land cover classification model cannot easily predict canopy height due to differences in input-output relationships, as the network may

[1]Laboratoire des Sciences du Climat et de l'Environnement, LSCE/IPSL, France [2]Kayrros SAS, Paris 75009, France [3]Department for AI in Society, Science, and Technology, Zuse Institute Berlin, Germany [4]Department of Information Systems, University of Münster, Germany [5]Department of Computer Science, CNRS, INRIA & École Normale Supérieure, Paris 75230, France. Correspondence to: Ibrahim Fayad <ifayad@lsce.ipsl.fr>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

prioritize learning features that are highly discriminative for a given task but not the other.

Third, EO models often use optical or radar imagery to predict single-valued targets like canopy height or land cover classes. However, they struggle with more complex outputs like the full vertical structure of vegetation.[1]

Foundation models (FMs) offer a promising solution to the aforementioned limitations by leveraging self-supervised pre-training on vast amounts of unlabeled data (Brown et al., 2020; Devlin, 2018; Kirillov et al., 2023; Minderer et al., 2022; Radford et al., 2021). For EO applications, such models are being developed to process diverse data types, including time series processing (Yuan et al., 2022), adaptation to different input satellite sensors (Xiong et al., 2024) or varying Ground Sample Distances (GSD) (Reed et al., 2023), and multi-modal data fusion (Astruc et al., 2025; 2024; Fuller et al., 2024). Most of current foundation models primarily address the adaptability issue, but not label scarcity. In fact, most EO-focused FMs are pre-trained masked auto-encoders (MAEs), which require extensive fine-tuning (Lehner et al., 2024; Singh et al., 2023). Moreover, existing FMs, even multi-modal ones, only leverage image-based modalities. As such, they excel in EO tasks requiring horizontal structural understanding (e.g., land cover mapping, tree species identification), but struggle with EO applications requiring vertical structure understanding (e.g., canopy height estimation). Additionally, their patch-based approach limits direct pixel-level prediction capabilities. This hinders the integration of LiDAR data, such as full-waveform LiDAR data available as labels at resolutions higher than the patch scale.

In this work, we propose a *Dense Unsupervised Nature Interpretation Algorithm* (DUNIA) that learns to generate pixel-level embeddings by aligning vertical (from a space-borne full waveform LiDAR) and horizontal structure information with satellite imagery through contrastive learning. This dual alignment strategy enables understanding of both vertical and horizontal structures, supporting diverse EO applications with minimal, if any, training. DUNIA achieves strong retrieval performance for several tasks and excels in forest structure benchmarks. In our experiments, we demonstrate the effectiveness of the resulting embeddings for seven key EO applications and show that zero-shot classification operating on these embeddings can outperform specialized supervised EO models in several cases. A simplified version of our approach is presented in Figure 1.

[1]Capturing such vertical structures is essential for understanding biomass allocation and species diversity. Predicting vertical structure from EO imagery is challenging due to its complexity and limited sensor data, requiring cross-modal understanding over direct prediction (Tan et al., 2024).

2. Background & Contribution

We sketch the related works along with our contributions.

2.1. Contrastive Learning

Contrastive learning trains networks to produce similar embeddings for semantically related inputs and dissimilar embeddings for unrelated ones. Related inputs may include different augmentations of the same sample or paired examples from different modalities. In this context, one of the main challenges is to avoid model collapse, where model outputs become constant. This is typically addressed by: 1) architectural designs or 2) specialized objective functions.

Prominent examples of architectural designs are BYOL (Grill et al., 2020) and SimSiam (Chen & He, 2021), which respectively use asymmetry to prevent collapse, BYOL employing a momentum encoder and SimSiam using a stop-gradient operation with a predictor network. Specialized objective functions can be categorized into pair-based contrastive losses, clustering losses, and negative-pair-free losses. Here, pair-based losses, such as those presented in SimCLR (Chen et al., 2020) and MoCo (He et al., 2020), align positive pairs and separate negative pairs, but assume unique positive pairs within a mini-batch, which is not always true for EO data, especially at high resolutions. Clustering methods like SeLa (Asano et al., 2019) and SwAV (Caron et al., 2020) relax this constraint by optimizing a contrastive objective over cluster assignments instead. Negative-pair-free methods, such as Barlow Twins (Zbontar et al., 2021) and VICReg (Bardes et al., 2021), focus on redundancy reduction in the feature dimension. Zero-CL (Zhang et al., 2021) uses whitening transformations on the embeddings to maximize the trace of the cross-covariance matrix, achieving alignment and redundancy reduction without negative pairs.

2.2. Self-Supervised Learning for EO

Inspired by masked image modeling (MIM) from vision transformers, several SSL frameworks have adapted MAE-style objectives to the remote sensing domain. These methods mainly focus on reconstructing masked inputs to learn spatial, spectral, and temporal representations. SatMAE (Cong et al., 2022) introduces a spectral-temporal aware masking scheme for Sentinel-2 (S-2) timeseries. SatMAE++ (Noman et al., 2024) extends SatMAE by incorporating multi-scale feature extraction. Scale-MAE (Reed et al., 2023) embeds spatial resolution information into positional encodings for multiscale imagery. DOFA (Xiong et al., 2024) extends masked autoencoding with a dynamic, spectral-aware weighting mechanism, enabling the model to encode diverse sensor modalities through a shared encoder.

Other related works on SSL for EO applications relies on

a contrastive objective to capture the relationships within the data. CROMA (Fuller et al., 2023) aligns radar and optical modalities using a cross-modal contrastive loss, supported by an auxiliary reconstruction head. DeCUR (Wang et al., 2024) explicitly disentangles intra- and inter-modal contrastive signals based on Barlow Twins (Zbontar et al., 2021). AnySat (Astruc et al., 2024) proposed a resolution- and temporal-adaptive framework capable of handling data from various sensors and resolutions, effectively combining multiple EO data sources into a shared embedding space.

2.3. Multi-Modal Learning

Multi-modal SSL approaches can be categorized into three main categories. (1) Modality-agnostic models use a shared network for different modalities (Carreira et al., 2022; Girdhar et al., 2022; Jaegle et al., 2021), but are typically limited to one modality at a time during training, restricting cross-modal knowledge sharing (Srivastava & Sharma, 2024). For EO applications, several models follow this approach. Scale-MAE (Reed et al., 2023) and DOFA (Xiong et al., 2024) both process imagery across varying resolutions, the latter also processes different wavelengths using a common encoder. (2) Fusion-encoder models integrate data from multiple modalities through cross-modal attention, effectively combining information from each modality (Bachmann et al., 2022; Bao et al., 2021; Singh et al., 2022). OmniSat (Astruc et al., 2025) and AnySat (Astruc et al., 2024) exemplify this by integrating data from diverse EO sources into a unified representation. (3) Finally, multi-encoder approaches use separate encoders for each modality, which are then aligned in a shared latent space. This strategy is highly effective for tasks such as cross-modal retrieval and zero-shot classification across different modalities: image/text (Radford et al., 2021; Jia et al., 2021; Yuan et al., 2021; Yu et al., 2022), audio/text (Guzhov et al., 2022), and video/text (Luo et al., 2022; Pei et al., 2023).

2.4. Contribution

Our work aligns closely with the abovementioned third category, focusing on mono- and cross-modal retrieval within the multi-encoder approach. Unlike prior models that align paired data at the instance level (e.g., matching an image to its textual description), here we perform pixel-level alignment. This is necessary to preserve the spatial resolution required to match with full LiDAR waveform footprints.

Specifically, our approach aligns sparse LiDAR waveforms with high-resolution imagery through contrastive learning to understand the vertical structure (i.e., pixel-waveform alignment) and simultaneously performs pixel-pixel alignment for horizontal structure understanding. DUNIA enables per-pixel vertical and horizontal structure retrieval and surpasses specialized models in forest structure benchmarks. For land cover understanding, it performs competitively in zero-shot and low-shot evaluations. Additionally, pixel embeddings can directly generate waveforms representing forest vertical structures — a task not possible with existing methods.

3. Approach

Our objective is to learn two distinct pixel-level embedding spaces corresponding to two EO data modalities — vertical and horizontal — using a single pre-trained model. This enables simultaneous understanding of both horizontal and vertical structures. Vertical information is derived from the Global Ecosystem Dynamics Investigation (GEDI) instrument (Dubayah et al., 2020), a spaceborne LiDAR that provides sparse measurements of 3D structures via 1D waveform signals. Horizontal information comes from 10 m resolution imagery acquired by the S-1 & 2 missions.

While our framework can be implemented in various ways, we adopt a modular design centered around a transformer encoder and two independent convolutional decoders. This encoder-decoder architecture enables us to generate pixel-sized embeddings for each input. The two decoders are designed to disentangle horizontal and vertical structure understanding: one focuses on spatial (horizontal) relationships, the other on vertical structure.

For horizontal structure alignment, we use a multi-temporal autoencoder (AE) that encodes a sequence of satellite images and produces temporally informed, per-pixel embeddings. This design achieves two goals: 1) it provides an augmented input to the pre-trained model, and 2) it allows the model to implicitly capture phenological patterns without requiring a time series at inference time.

For vertical structure alignment, we also use an AE, this time encoding GEDI waveforms. The waveform encoder projects the 1D input into a latent space, which is aligned with the corresponding pixel embedding from the pre-trained model. While the decoder is not needed for the alignment itself, it is necessary for waveform generation, ensuring that the latent space remains semantically and structurally meaningful.

Figure 2 illustrates the main components of our approach, including the inputs to each model, their architectural building blocks, and the layers used for alignment. Below, we describe the main modules in more detail, followed by our pre-training objectives and the latent diffusion model for waveform generation. Full details are provided in Appendix A.

3.1. Pre-trained Model

The pre-trained model in DUNIA (Figure 2) takes an image $\mathcal{I} \in \mathbb{R}^{H \times W \times C}$ and produces pixel level embeddings in $\mathbb{R}^{H \times W \times D_p}$. C is the number of input channels, (H, W) is the image resolution, and D_p is the target projection di-

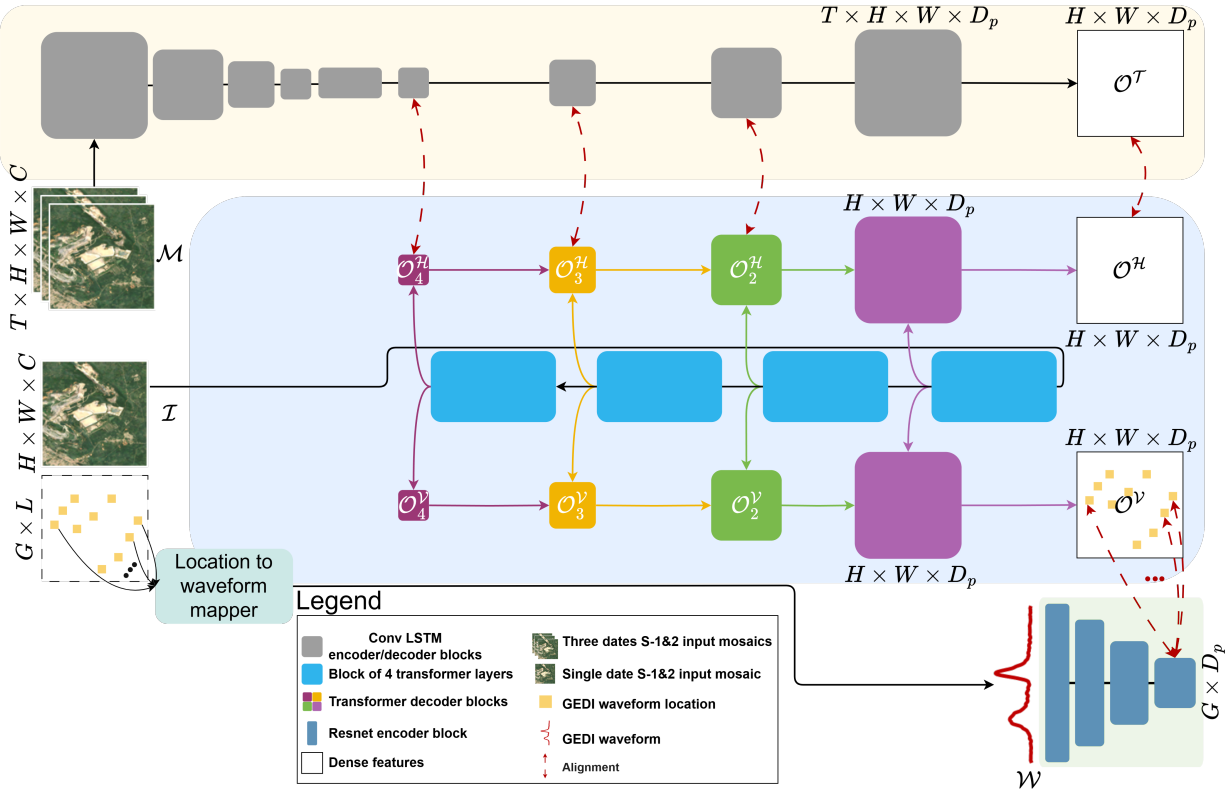

Figure 2. Simplified architectural overview of the proposed framework. The figure illustrates the key encoder and decoder components, input modalities, and the layers where alignment is performed (red dashed arrows). The pre-trained model is shaded in blue. The multi-temporal image AE, is shaded in yellow. The waveform encoder is shaded in green. Additional components, losses and training objectives are omitted for clarity; see Figure A1 for full details.

mension. $\mathcal{I}$ is a median composite image generated from S-1 & 2 observations over several dates. The pre-trained model comprises five building blocks: a patch embedding, a shared encoder, two similar decoders, neighborhood attention layers and several projection heads. Following is a brief description of each of these blocks.

Patch Embedding. The 2D composite image $\mathcal{I}$ is split into $N = HW/P^2$ patches, each of dimension D using a convolutional layer, with P and D the patch size and the patch embedding dimension respectively.

Encoder Architecture. The encoder consists of 16 standard transformer layers (Vaswani et al., 2017), organized into four Transformer Blocks, of four layers each. Each layer inputs/outputs a sequence of image tokens in $\mathbb{R}^{N \times D}$.

Decoder Architecture. The two decoders transform the encoder outputs into two sets of pixel-sized embeddings ($\mathcal{O}^{\mathcal{V}}$, $\mathcal{O}^{\mathcal{H}} \in \mathbb{R}^{H \times W \times D_p}$) for pixel-waveform and pixel-pixel alignment respectively. They use a hierarchical structure inspired by Ranftl et al. (2021) that progressively upsamples features while combining information from different encoder layers to maintain both fine and coarse details. Each decoder block ($d \in \{1, 2, 3, 4\}$, with $d = 1$ representing the final decoder layer) produces an image-like feature map of shape $\frac{H}{2^{(d-1)}} \times \frac{W}{2^{(d-1)}} \times D_{p_d}$, where $D_{p_d} = 2^{d-1} D_p$ is the feature dimension out of block d. We refer to the output at these stages for a given decoder as $\mathcal{O}_d$. For simplicity, we will henceforth refer to $\mathcal{O}_1$ as $\mathcal{O}$.

Neighborhood Attention (NA). To enhance local spatial relationships, we add two NA layers (Hassani et al., 2023) for each decoder block at the highest resolution ($d = 1$), where each pixel attends to its surrounding window of size w. This complements the global attention from the transformer layers with explicit element relation modeling mechanism.

Projection Head. For the pixel-pixel alignment objective, we follow standard practice and append a projection head for each output of $\mathcal{O}^{\mathcal{H}}_d$. As the projection head becomes more specialized towards the training objective (Xue et al., 2024), its output becomes less generalizable when training and downstream objectives are misaligned, and as such it is

discarded after training. For the pixel-waveform alignment $\mathcal{O}^{\mathcal{V}}$ no projection head is used as the downstream tasks (e.g., waveform generation) align with the training objective.

3.2. Waveform Model

The waveform AE is based on a VQ-VAE (Vector Quantized Variational Autoencoder) (van den Oord et al., 2017) architecture. It consists of three primary components: a waveform encoder (Figure 2), a residual vector quantizer (RVQ), and a waveform decoder. The input to this model are GEDI LiDAR waveforms ($\mathcal{W}$) available only for certain pixel locations in the composite image $\mathcal{I}$.

Waveform Encoder. The waveform encoder ($\mathcal{E}_w$) uses a ResNet-based architecture to process 1D waveform inputs through multiple stages, progressively reducing their length while increasing feature representation capacity. $\mathcal{E}_w$ transforms a waveform ($\mathcal{W}$) of shape $\mathbb{R}^{L\times 1}$ into a latent representation $z_e \in \mathbb{R}^{L/16\times 16}$. To align z_e with its corresponding pixel embedding from $\mathcal{O}^{\mathcal{V}}$ we perform a channel-wise average pooling followed by a linear projection. This process transforms z_e into $\mathcal{O}^{\mathcal{W}} \in \mathbb{R}^{D_p}$.

Residual Vector Quantizer (RVQ). For regularization by enforcing discrete representations, we use an RVQ layer with Q quantizers each containing V codebook vectors. The RVQ iteratively quantizes the latent waveform representation z_e through a sequence of quantizers, with each processing the residual from the previous one to produce the final quantized representation z.

Waveform Decoder. The waveform decoder $\mathcal{D}_w$ mirrors the structure of the encoder but operates in reverse order to reconstruct the waveform from its quantized latent representation z. In essence, $\mathcal{D}_w(z) = \hat{\mathcal{W}}$. While waveform reconstruction is not necessary for the alignment, reconstructing directly from aligned encoded waveforms ensures that the future generation process operates on a waveform latent space that is already structured based on the shared semantic and structural alignment with the pixel embeddings. This approach is beneficial as we are interested in generating waveforms given pixel inputs.

3.3. Multitemporal Image Processing Model

The multitemporal image model (Figure 2) is designed to process temporal sequences of median composite images, following an autoencoder (AE) architecture. The input to this model is a sequence of images $\mathcal{M} \in \mathbb{R}^{T\times H\times W\times C}$ covering the same area as the input to the pre-training model. These images come from T S-1 & 2 acquisition dates that overlap with the dates used to generate the composite image $\mathcal{I}$. This design choice, unlike multi-temporal pre-training models that require time series data during training and inference, or mono-temporal models limited to a single date, is a trade-off that enables the model to capture some phenological traits without needing time series data during inference. Indeed, while image composites such as a median composite may be richer than single-date images, they are still less informative than a full time series, as they only provide a median reflectance value over a given period. Conversely, even this simple form of aggregation preserves significantly more information than a single-date image.

Multi-temporal Image AE. This AE follows a UNet structure, but replaces conventional convolutional blocks with ConvLSTM (Shi et al., 2015) layers to explicitly model temporal correlations. The model takes a multi-temporal input image $\mathcal{M}$ and produces a feature map $\mathcal{X} \in \mathbb{R}^{T\times H\times W\times D_p}$, which is then reconstructed back to a tensor $\hat{\mathcal{M}}$ of the same shape as the inputs. Each decoder block produces an image-like feature map of shape $\frac{H}{2^{(d-1)}} \times \frac{W}{2^{(d-1)}}$ denoted as $\mathcal{X}_d$ for a respective decoder stage.

Temporal Pooling. To align the output embeddings from the multi-temporal image AE with those from the pre-training model, we first perform temporal average pooling. This transforms $\mathcal{X}_d \in \mathbb{R}^{T\times \frac{H}{2^{(d-1)}} \times \frac{W}{2^{(d-1)}} \times D_p}$ to $\mathcal{O}_d^{\mathcal{T}} \in \mathbb{R}^{\frac{H}{2^{(d-1)}} \times \frac{W}{2^{(d-1)}} \times D_p}$, which then passes through a projection head. For the output with the highest resolution (i.e., $\mathcal{O}_1^{\mathcal{T}}$) we also append two NA layers before the projection head. For simplicity we refer to $\mathcal{O}_1^{\mathcal{T}}$ as $\mathcal{O}^{\mathcal{T}}$.

3.4. Pre-Training Objective

DUNIA is pre-trained on pixel-waveform and pixel-pixel alignment, alongside modality reconstruction for the modality-specific AEs. For the alignment task, we rely on two, similarly performing, non-negative-pair contrastive losses, namely VICReg (Bardes et al., 2021) and Zero-CL (Zhang et al., 2021). The selection between either loss functions is determined by the number of available elements within a mini-batch and the requirements of each loss function. VICReg requires large batch sizes to perform well (Bardes et al., 2021), which is a non-issue for the pixel-pixel contrastive objective given the high number of pixels available in each mini-batch of images. However, this is not suitable for the waveforms which are few in number within the mini-batch. On the other hand, Zero-CL performs well even for small batch sizes but faces computational bottlenecks with very large batches.

3.4.1. Pixel-Waveform Alignment

Zero-CL replaces the alignment and uniformity terms in negative-pair-based contrastive losses (Arora et al., 2019) with, respectively, an instance-wise contrastive loss ($\mathcal{L}_{\text{Ins}}$) and a feature-wise contrastive loss ($\mathcal{L}_{\text{Fea}}$). The overall

loss $\mathcal{L}_{Zero-CL}$ is $\mathcal{L}_{\text{Fea}} + \mathcal{L}_{\text{Ins}}$. $\mathcal{L}_{Zero-CL}$ is applied on $Z^A \in \mathbb{R}^{G\times D_p}$, the L_2 normalized pixel embeddings from $\mathcal{O}^{\mathcal{V}}$, and $Z^B \in \mathbb{R}^{G\times D_p}$ their corresponding L_2 normalized waveform embeddings from $\mathcal{O}^{\mathcal{W}}$, where G is the number of available GEDI samples in a given mini-batch. The formulation for this loss can be found in Appendix B.

3.4.2. Pixel-Pixel Alignment

VICReg is formulated as three loss terms: variance ($\mathcal{L}_{\text{var}}$), invariance ($\mathcal{L}_{\text{inv}}$), and covariance ($\mathcal{L}_{\text{cov}}$). Let $Z_d^{\mathcal{H}} \in \mathbb{R}^{M_d\times D_{p_d}}$ represent the L_2 normalized pixel embeddings from $\mathcal{O}_d^{\mathcal{H}}$, $M_d = B \times \frac{H}{2^{(d-1)}} \times \frac{W}{2^{(d-1)}}$ with B the mini-batch size, $\frac{H}{2^{(d-1)}}$ and $\frac{W}{2^{(d-1)}}$ the height and width of a decoder's output feature map. $Z_d^{\mathcal{T}} \in \mathbb{R}^{M_d\times D_{p_d}}$ is their corresponding L_2 normalized pixel embeddings from $\mathcal{O}_d^{\mathcal{T}}$. The overall hierarchical pixel-pixel alignment loss is expressed as:

$$\begin{aligned}\mathcal{L}_{\text{VICReg}} = \sum_{d=1}^{4} \alpha_v \mathcal{L}_{\text{var}}(Z_d^{\mathcal{H}}, Z_d^{\mathcal{T}}) \\ +\beta_i \mathcal{L}_{\text{inv}}(Z_d^{\mathcal{H}}, Z_d^{\mathcal{T}}) \\ +\gamma_c \mathcal{L}_{\text{cov}}(Z_d^{\mathcal{H}}, Z_d^{\mathcal{T}})\end{aligned} \tag{1}$$

Following Bardes (2021) we set α_v and β_i to 25.0 and γ_c to 1.0. VICReg formulation can be found in Appendix B.

3.4.3. Reconstruction Losses

In addition to the previously defined contrastive losses, our pre-training objective also has three additional reconstruction losses for the modality-specific AEs (i.e., the waveform AE and the multi-temporal image AE) and on the outputs of $\mathcal{O}^{\mathcal{H}}$ for regularization. The reconstruction losses are simply the mean squared error loss (MSE) on the reconstructed waveform $\hat{\mathcal{W}} \in \mathbb{R}^{1\times L}$ given input waveform $\mathcal{W}$, the reconstructed multi-temporal image $\hat{\mathcal{M}} \in \mathbb{R}^{T\times H\times W\times C}$ given input image $\mathcal{M}$, and the reconstructed mono-temporal image $\hat{\mathcal{I}} \in \mathbb{R}^{H\times W\times C}$ given input image $\mathcal{I}$. The reconstruction term is formulated as:

$$\begin{aligned}\mathcal{L}_{\text{rec}} = MSE(\mathcal{W}, \hat{\mathcal{W}}) + MSE(\mathcal{M}, \hat{\mathcal{M}}) \\ + MSE(\mathcal{I}, \hat{\mathcal{I}})\end{aligned} \tag{2}$$

3.5. Waveform Generation

To generate a full GEDI waveform given a pixel input, we leverage latent diffusion models (LDMs) (Rombach et al., 2022). LDMs are generative models that iteratively refine a latent representation, starting from a normally distributed prior in a learned latent space. These models can also be conditioned on auxiliary inputs y to model conditional distributions $p(z|y)$. In our case, the conditioning variable consists of pixel embedding $\mathcal{O}_{\phi,\lambda}^{\mathcal{V}}$ at coordinate (ϕ, λ), while the latent variable $z \in \mathbb{R}^{L/16\times 16}$ represents the embedding of the quantized waveform obtained from the RVQ layer (Section 3.2). During inference, the waveform generation follows a two-step process. First we sample a representation z given condition $\mathcal{O}_{\phi,\lambda}^{\mathcal{V}}$ using the LDM, and then the frozen $D_w(z)$ yields a waveform.

For a detailed description of the diffusion process, including the denoising objective, noise schedule, and latent variable sampling, we refer the reader to Appendix C.

4. Experimental Evaluation

We provide a detailed experimental evaluation showing that DUNIA is able to achieve high performance across a variety of tasks in zero-shot and fine-tuned settings, including land cover classification, crop mapping, and vertical forest structure analysis (e.g., canopy cover, height and GEDI waveform retrieval). We leverage diverse datasets to inform the model on horizontal structures – S-1 & 2, and vertical structures with GEDI waveforms.

The following section contains a summary of the experimental setup, with full details in the appendix. Model configuration and optimisation can be found in Appendix E. For clarity, during the pre-training phase, we set the input image size to 64×64 pixels, with 14 channels from stacked S-1 & 2 image composites. The embedding dimension is set to 64 (i.e., $D_p = 64$). During inference, the input image can be of any size, but we inferred on 256×256 pixel images. Our code is available at `github.com/AI4Forest/DUNIA`.

4.1. Experimental Setup

For our evaluation, we resort to various datasets and tasks. The datasets and experimental details are described next.

4.1.1. Datasets

Pre-training Datasets. We used S-2 Level-2A surface reflectance data from Google Earth Engine, including 10 m and 20 m spatial resolution bands with the latter upscaled to 10 m. Two sets of mosaics were created over the entire metropolitan French territory: a single leaf-on season mosaic (April-September 2020) for the pre-trained model, and three four-month mosaics (October 2019-September 2020) for the multitemporal AE. Cloud filtering was applied using the S-2 Cloud Probability dataset.

S-1 data were obtained from the S-1A and S-1B satellites operating at C-band, collected in Interferometric Wide swath mode with VV and VH polarizations. The data was calibrated, geometrically corrected, and resampled to 10 m resolution. Similar to S-2, we created two sets of mosaics with normalized backscattering coefficients.

GEDI is a full-waveform LiDAR sensor on the International Space Station (ISS), operational between 51.6° N and 51.6°

S from 2019-2023. We used Level 1B, 2A, and 2B data from April 2019 to December 2021, extracting waveforms, geolocation, height metrics ($\mathcal{W}_{rh}$), canopy cover ($\mathcal{W}_c$), and Plant Area Index ($\mathcal{W}_{pai}$). After quality filtering following Fayad et al. (2024) and seasonal selection, the dataset contained $\approx$ 19 million waveforms, covering less than 1% of the total surface area of France.

Overall, our pre-training dataset consisted of 836K 64×64 pixels S-1 & 2 images with, on average, 26 GEDI waveforms per image. Additional details on the pre-training datasets can be found in Appendix F.

Evaluation Datasets. We evaluated our model on seven downstream tasks using labels from various data products at resolutions matching or lower than our model's output (10 m). *PureForest (PF)* provides a benchmark dataset of ground truth patches for classifying mono-specific forests in France, featuring high-resolution imagery and annotations for over 135K 50×50 m (i.e., 5x5 pixels) patches across 13 tree species (Gaydon & Roche, 2024). *CLC+Backbone (CLS_+)* is a pan-European land cover inventory for 2021, utilizing S-2 time series (2020-2022) and a TempCNN classifier (Pelletier et al., 2019) to produce a 10 m raster indicating the dominant land cover among 11 classes. *PASTIS* is a crop mapping dataset by Garnot et al. (2021), covering 18 crop classes and 1 background class with 2433 densely annotated 128×128 pixels images at 10 m resolution. The *vertical structure dataset* assesses model performance in mapping forest heights, canopy cover, plant area index, and waveform retrieval at 10 m resolution, relying on GEDI-derived products as presented earlier. When available, we used the train/val/test split used by the references (i.e., PF and *PASTIS*). For the CLS_+ dataset, we used the same split as the unsupervised dataset, which followed a 65/10/25 split.

4.1.2. Performance Evaluation

We evaluated our model's retrieval capacities for both zero-shot dense prediction tasks and its performance in the fine-tuned setting. For both settings, we used the weighted F1 score (*w*F1) for classification tasks, root mean squared error (RMSE) and Pearson's correlation coefficient (r) for regression tasks. To evaluate the similarity between acquired and retrieved/generated waveforms, we used Pearson's correlation coefficient, computed between the time-aligned retrieved/generated waveforms and the acquired waveforms.

For zero-shot classification, we constructed retrieval databases based on the downstream tasks. For vertical structure tasks, outputs from $\mathcal{O}^{\mathcal{V}}$ served as queries, and we created a single database with L_2 normalized waveform embeddings from $\mathcal{O}^{\mathcal{W}}$ as keys, paired with four target labels: $\mathcal{W}_{rh}$, $\mathcal{W}_c$, $\mathcal{W}_{pai}$, and the complete waveform ($\mathcal{W}$). For horizontal structure tasks, outputs from $\mathcal{O}^{\mathcal{H}}$ were used as queries, creating a separate database for each target as not all targets were available at all pixels simultaneously. Here, keys are L_2 normalized pixel embeddings from $\mathcal{O}^{\mathcal{H}}$, with targets being a land cover class (i.e., CLC_+), crop type (i.e., $PASTIS$), or tree species (i.e., PF). For tree species identification, the labels cover a 5×5 pixel area, so queries and keys are the averaged embeddings over this window. Next, given an input image and its L_2 normalized pixel embeddings from $\mathcal{O}^{\mathcal{V}}$ or $\mathcal{O}^{\mathcal{H}}$, we retrieve the k nearest neighbors (KNN) for each pixel based on cosine similarity and assign the target class by distance-weighted voting. For the KNN retrieval, keys were obtained from the training split while the queries were obtained from the test split.

For the low-shot fine-tuning, we froze the entire pre-trained network except for the last NA layer in each decoder, which were appended with an output head consisting of two sequential 1×1 convolutional layers. The first layer halves the input channels, and the second projects the reduced representation to the desired output size.

We evaluated zero-shot classification and low-shot fine-tuning in a low-data regime. We define a dataset with a low number of labels based on the type of labels usually available for this dataset. For CLC_+ and $PASTIS$, labeled data are available as densely annotated images. For PF, $\mathcal{W}$, $\mathcal{W}_{rh}$, $\mathcal{W}_c$ and $\mathcal{W}_{pai}$, labeled data are available as single annotated pixels.

4.1.3. Competing Models.

We compared DUNIA in the fine-tuned setting to five current state-of-the-art Earth Observation FMs: SatMAE (Cong et al., 2022), DOFA (Xiong et al., 2024), DeCUR (Wang et al., 2024), CROMA (Fuller et al., 2023) and AnySat (Astruc et al., 2024). SatMAE takes as input S-2 imagery, DOFA, DeCUR, and CROMA take as input S-1 & 2 imagery, while Anysat is pre-trained using S-1 & 2 times series as well as very high-resolution imagery. For a fair comparison, we pre-trained all competing models (using pre-trained weights when available) for 250K steps using our datasets and the training details from the respective papers. For AnySat, we used multi-temporal S-1 & 2 mosaics with three time steps and included SPOT images at 1.5 m resolution as an additional input modality during pre-training but fine-tuned using only S-1 & 2. All competing models were evaluated on all downstream tasks except waveform generation, as they do not support this task.

4.2. Results

Our evaluation shows that embeddings from our proposed framework, combined with simple zero-shot classifiers, surpass specialized supervised models in several tasks. In the fine-tuned setting, our model demonstrates strong low-shot performance, rivaling or exceeding state-of-the-art methods.

Table 1. Top-1 retrieval-based zero-shot classification performance of DUNIA for different retrieval settings. SOTA results, when available, represent the best-performing supervised models for any given dataset. † are retrained models for a particular dataset. and are DUNIA query embeddings from $\mathcal{O}^{\mathcal{V}}$ and $\mathcal{O}^{\mathcal{H}}$ respectively. S represents the number of samples in the retrieval database, with *im* meaning a 64×64 pixels fully annotated image, and *l* meaning a single annotated pixel. KNN represents the k nearest neighbors used in the distance-weighted voting. $\mathcal{W}^{**}$ represents performance results for vertical structures higher than 5 m. Best scores are in **bold**.

DATASET	METRIC	SOTA		S	DUNIA		
					KNN = 50	KNN = 5	KNN =1
$\mathcal{W}_{rh}$	RMSE (r)	SCHWARTZ ET AL. (2023) LIU ET AL. (2023) TOLAN ET AL. (2024) LANG ET AL. (2023)	5.2 (.77) 5.2 (.76) 8.5 (.59) 5.6 (.76)	50K *l* *	**2.0** (.93)	2.1 (.93)	2.1 (.91)
$\mathcal{W}_c$	RMSE (r)	SCHWARTZ ET AL. (2023)†	22.1 (.54)	50K *l*	**11.7** (.89)	12.1 (.84)	12.3 (.86)
$\mathcal{W}_{pai}$	RMSE (r)	SCHWARTZ ET AL. (2023)†	1.5 (.35)	50K *l*	**0.71** (.75)	0.74 (.73)	0.74 (.72)
CLC_+	*w*F1	—		500 *im*	**80.1**	74.3	69.5
$PASTIS$	OA	GARNOT & LANDRIEU (2021)	**84.2**	500 *im*	56.2	54.1	49.5
PF	*w*F1	GAYDON & ROCHE (2024)	74.6	50K *l*	73.8	**76.0**	75.8
$\mathcal{W}^{**}$	r	—		50K *l*	—	—	**.70**

* This number of samples is considered as a low data regime due to: 1) a total coverage area of $\approx 31Km^2$ 2) only $\approx 0.25\%$ of data required compared to some supervised models.

4.2.1. ZERO-SHOT CLASSIFICATION PERFORMANCE

Zero-shot results in Table 1 demonstrate that for vertical structure products ($\mathcal{W}_{rh}$, $\mathcal{W}_c$, $\mathcal{W}_{pai}$), DUNIA with KNN=50 outperforms specialist models. It achieves RMSE (r) improvements of 3.2 m (.16), 10.4% (.35), and 0.79 (.4) for, respectively, canopy height, cover, and plant area index, and, matching the mapping quality of supervised methods (Figures I1 and I2). For tree species identification (PF), DUNIA slightly underperforms by 0.8% at KNN=50 but exceeds the baseline by 1.4% when decreasing the number of neighbors (KNN=5) used for the distance-weighted voting. For land cover mapping (CLC_+), DUNIA achieves strong performance with a *w*F1 score of 80.1%. However, for crop type mapping ($PASTIS$), DUNIA significantly underperforms compared to the state-of-the-art, which is likely due to the variability of the phenological cycles of crops that cannot be well captured by a single median composite.

For lower KNN values, the performance of vertical structure-related products and tree species identification remains stable. In contrast, land cover and crop type mapping experience significant performance drops, indicating that the model needs more samples for accurate classification. For retrieval databases with fewer samples (Table G1), vertical structure-related products ($\mathcal{W}_{rh}$, $\mathcal{W}_c$, $\mathcal{W}_{pai}$) maintain their performance and map qualities (Figure I4), while other products (CLC_+, $PASTIS$, PF) show a substantial degradation in performance.

Regarding waveform ($\mathcal{W}$) retrieval performance, we observe that our model is able to retrieve relevant waveforms given a pixel embedding as a query, with a correlation coefficient of .70 for S=50k *l*, and decreases by a small factor with smaller retrieval database sizes (Table G1). This is to be expected given the small subset of waveforms in the retrieval database that might not reflect all possible variations for a particular height class. Retrieved vs. reference waveforms can be found in Figures I5 to I8.

4.2.2. FINE-TUNING PERFORMANCE

Fine-tuning results in Table 2 demonstrate that DUNIA outperforms the other models in vertical structure-related products, with all models except AnySat trailing significantly. In comparison to DUNIA's zero-shot results, all the models underperformed in the finetuning setting for $\mathcal{W}_c$ and $\mathcal{W}_{pai}$ due to the long-tailed distribution of these two products that are not well modeled with the simple L_1 loss that we used. Qualitatively, as shown in Figures I1 and I2, DUNIA and AnySat show a similar level of detail; however, DUNIA is capable of producing higher values, which we attribute to the alignment with the vertical structure in the pre-training stage. For CROMA, Figures I1 and I2 shows that this model produces much smoother maps than the two other models.

For CLC_+ and PF, DUNIA's performance in the fine-tuned setting increased in comparison to the zero-shot setting with results on par with the much more data-heavy AnySat model (Table 2). However, DUNIA underperformed by a significant margin ($\downarrow 4.1\%$) in comparison to AnySat for the $PASTIS$ dataset. Qualitatively (Figure I3) both DUNIA and AnySat show similar level of detail for the CLC_+ dataset, in contrast to CROMA which also showed very smooth maps with fewer details.

In the novel task of waveform generation, we observed a correlation (r) increase of .08 between reference and gener-

Table 2. Fine-tuning performance of DUNIA and the five competing models. and are DUNIA's embeddings from $\mathcal{O}^{\mathcal{V}}$ and $\mathcal{O}^{\mathcal{H}}$ respectively. S represents the number of samples used for the fine-tuning, with *im* meaning a 64×64 pixels fully annotated image, and *l* meaning a single annotated pixel. $\mathcal{W}^{**}$ represents performance results for vertical structures higher than 5 m. Best scores are in **bold**.

DATASET	METRIC	SAMPLES (S)	DUNIA	ANYSAT	CROMA	DOFA	DECUR	SATMAE
$\mathcal{W}_{rh}$	RMSE (r)	5K *im*	**1.3** (.95)	2.8 (.89)	3.5 (.78)	11.0 (.51)	11.0 (.55)	10.5 (.52)
$\mathcal{W}_{c}$	RMSE (r)	5K *im*	**9.8** (.85)	12.1 (.79)	14.2 (.73)	29.2 (.50)	28.5 (.48)	30.2 (.48)
$\mathcal{W}_{pai}$	RMSE (r)	5K *im*	**0.62** (.71)	0.95 (.67)	1.2 (.61)	1.5 (.38)	1.6 (.38)	1.6 (.39)
CLC_{+}	*w*F1	5K *im*	**90.3**	90.1	86.4	72.0	75.1	75.0
$PASTIS$	*w*F1	1.5K *im*	77.0	**81.1**	73.3	54.5	57.3	55.2
PF	*w*F1	50K *l*	82.2	**82.3**	80.5	79.8	78.9	78.8
$\mathcal{W}^{**}$	r	≈19M *l*	.78	—	—	—	—	—

ated waveforms compared to retrieval (r=.70 with S=50K *l*, KNN=1). This correlation dropped to .75 when the diffusion model was trained on just 20% of the available waveforms (Table G2). Retrieved vs. generated waveforms can be found in Figures I5 to I8.

To further analyze the contributions of different architectural choices, including the use of separate decoders and the impact of our alignment strategies, we provide additional ablation studies in Appendix H.

5. Discussion

We introduced DUNIA, a novel framework for Earth Observation applications that learns dense representations through mono-modal and cross-modal alignment. This strategy enables our model to perform mono-modal and cross-modal retrieval tasks on several datasets with high precision. It also allows novel capabilities like retrieving or generating highly complex outputs from pixel inputs. To the best of our knowledge, this is the only model capable of generating GEDI waveforms at this scale.

Our main contributions lie in several design choices at the decoding stage. The proposed pre-trained model encoder is flexible and can be adapted to components from other existing approaches; for instance, modules that support multi-temporal input, or handle images with varying resolutions. Nonetheless, in this work we focused on input data - S-1 & 2, alongside GEDI - that are freely accessible globally to promote easier adoption.

One of the main limitations of this work is also coincidentally the reliance on static inputs from S-1 & 2 data, which has limitations for datasets requiring multi-temporal images, such as crop type mapping. However, our design choices prioritize mono-temporal data to reduce storage requirements and enable applications in areas where time series data are not readily available. Another limitation, common to all EO models, is that the current pre-trained model is tailored to the area and year on which it was trained on; we expect that it will require pre-training when used elsewhere. Nonetheless, the model was designed to be pre-trained with limited computational resources.

Acknowledgements

This work is supported via the AI4Forest project, which is funded by the French National Research Agency (ANR; grant number ANR-22-FAI1-0002) and the German Federal Ministry of Education and Research (BMBF; grant number 01IS23025A).

Impact Statement

Forests are the third-most important terrestrial carbon sink for climate change mitigation by absorbing about one-third of anthropogenic CO_2 emissions (Friedlingstein et al., 2022). However, these ecosystems are under increasing threat from deforestation, degradation, and climate-induced disturbances such as wildfires, insect outbreaks, and droughts (Anderegg et al., 2022). Global initiatives, including the Glasgow Declaration and the Sustainable Development Goals[2] emphasize the need for accurate forest monitoring to support conservation, restoration, and carbon sequestration projects. Unfortunately, despite these high praises, many other major initiatives have struggled due to a lack of reliable, high-resolution data on forest carbon stocks and how they change over time.

Current forest monitoring systems heavily rely on ground-based inventories, which provide robust statistical estimates of biomass and carbon at national or regional scales but lack the spatial granularity needed for tracking localized carbon gains and losses. Additionally, the largest forested regions—particularly in boreal and tropical areas—remain vastly under-sampled. While satellite data offers global coverage, traditional approaches are often constrained by their reliance on extensive labeled datasets, limiting their ability to map forest properties comprehensively.

DUNIA introduces a novel self-supervised learning ap-

[2] https://sdgs.un.org/2030agenda

proach that overcomes these limitations by generating pixel-level embeddings from freely accessible satellite data, including optical and radar imagery. By aligning these embeddings with sparse but highly informative LiDAR waveforms, our approach enables the estimation of multiple forest attributes—including canopy height, fractional cover, land cover, tree species, and vertical structure—without requiring task-specific labels. This method allows for zero-shot and low-shot predictions of key forest variables, paving the way for scalable monitoring of global ecosystems.

By providing a unified, multimodal representation of forests, DUNIA democratizes access to advanced Earth Observation tools, enabling researchers, policymakers, and conservationists to make informed decisions with minimal reliance on expensive field data collection. This work lays the foundation for high-resolution, large-scale ecosystem monitoring, offering a transformative approach to quantifying forest structure, biomass, and biodiversity in support of climate change mitigation and sustainable land management.

References

Anderegg, W. R., Wu, C., Acil, N., Carvalhais, N., Pugh, T. A., Sadler, J. P., and Seidl, R. A climate risk analysis of Earth's forests in the 21st century. *Science*, 377(6610): 1099–1103, 2022.

Andrychowicz, M., Espeholt, L., Li, D., Merchant, S., Merose, A., Zyda, F., Agrawal, S., and Kalchbrenner, N. Deep learning for day forecasts from sparse observations. *arXiv preprint arXiv:2306.06079*, 2023.

Arora, S., Khandeparkar, H., Khodak, M., Plevrakis, O., and Saunshi, N. A theoretical analysis of contrastive unsupervised representation learning. *arXiv preprint arXiv:1902.09229*, 2019.

Asano, Y. M., Rupprecht, C., and Vedaldi, A. Self-labelling via simultaneous clustering and representation learning. *arXiv preprint arXiv:1911.05371*, 2019.

Astruc, G., Gonthier, N., Mallet, C., and Landrieu, L. AnySat: An Earth observation model for any resolutions, scales, and modalities. *arXiv preprint arXiv:2412.14123*, 2024.

Astruc, G., Gonthier, N., Mallet, C., and Landrieu, L. Omnisat: Self-supervised modality fusion for earth observation. In *European Conference on Computer Vision*, pp. 409–427. Springer, 2025.

Bachmann, R., Mizrahi, D., Atanov, A., and Zamir, A. Multimae: Multi-modal multi-task masked autoencoders. In *European Conference on Computer Vision*, pp. 348–367. Springer, 2022.

Bao, H., Wang, W., Dong, L., Liu, Q., Mohammed, O. K., Aggarwal, K., Som, S., and Wei, F. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *arXiv preprint arXiv:2111.02358*, 2021.

Bardes, A., Ponce, J., and LeCun, Y. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

Bardes, A., Ponce, J., and LeCun, Y. Vicregl: Self-supervised learning of local visual features. *Advances in Neural Information Processing Systems*, 35:8799–8810, 2022.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

Carreira, J., Koppula, S., Zoran, D., Recasens, A., Ionescu, C., Henaff, O., Shelhamer, E., Arandjelovic, R., Botvinick, M., Vinyals, O., et al. Hip: Hierarchical perceiver. *arXiv preprint arXiv:2202.10890*, 2022.

Chen, C., Jing, L., Li, H., Tang, Y., and Chen, F. Individual tree species identification based on a combination of deep learning and traditional features. *Remote Sensing*, 15(9): 2301, 2023.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Chen, X. and He, K. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15750–15758, 2021.

Chen, X., Liang, C., Huang, D., Real, E., Wang, K., Pham, H., Dong, X., Luong, T., Hsieh, C.-J., Lu, Y., et al. Symbolic discovery of optimization algorithms. *Advances in neural information processing systems*, 36, 2024.

Cong, Y., Khanna, S., Meng, C., Liu, P., Rozi, E., He, Y., Burke, M., Lobell, D., and Ermon, S. Satmae: Pre-training transformers for temporal and multi-spectral satellite imagery. *Advances in Neural Information Processing Systems*, 35:197–211, 2022.

Dalagnol, R., Wagner, F. H., Galvão, L. S., Braga, D., Osborn, F., da Conceição Bispo, P., Payne, M., Junior, C. S.,

Favrichon, S., Silgueiro, V., et al. Mapping tropical forest degradation with deep learning and planet nicfi data. *Remote Sensing of Environment*, 298:113798, 2023.

Devlin, J. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Dubayah, R., Blair, J. B., Goetz, S., Fatoyinbo, L., Hansen, M., Healey, S., Hofton, M., Hurtt, G., Kellner, J., Luthcke, S., et al. The global ecosystem dynamics investigation: High-resolution laser ranging of the earth's forests and topography. *Science of remote sensing*, 1:100002, 2020.

European Environment Agency. Clc+backbone 2021 (raster 10 m), europe, 3-yearly, jun. 2024, 2024. URL https://sdi.eea.europa.eu/catalogue/srv/api/records/71fc9d1b-479f-4da1-aa66-662a2fff2cf7?language=all.

Fayad, I., Ciais, P., Schwartz, M., Wigneron, J.-P., Baghdadi, N., de Truchis, A., d'Aspremont, A., Frappart, F., Saatchi, S., Sean, E., et al. Hy-tec: a hybrid vision transformer model for high-resolution and large-scale mapping of canopy height. *Remote Sensing of Environment*, 302: 113945, 2024.

Friedlingstein, P., Jones, M. W., O'Sullivan, M., Andrew, R. M., Bakker, D. C. E., Hauck, J., Le Quéré, C., Peters, G. P., Peters, W., Pongratz, J., Sitch, S., Canadell, J. G., Ciais, P., Jackson, R. B., Alin, S. R., Anthoni, P., Bates, N. R., Becker, M., Bellouin, N., Bopp, L., Chau, T. T. T., Chevallier, F., Chini, L. P., Cronin, M., Currie, K. I., Decharme, B., Djeutchouang, L. M., Dou, X., Evans, W., Feely, R. A., Feng, L., Gasser, T., Gilfillan, D., Gkritzalis, T., Grassi, G., Gregor, L., Gruber, N., Gürses, O., Harris, I., Houghton, R. A., Hurtt, G. C., Iida, Y., Ilyina, T., Luijkx, I. T., Jain, A., Jones, S. D., Kato, E., Kennedy, D., Klein Goldewijk, K., Knauer, J., Korsbakken, J. I., Körtzinger, A., Landschützer, P., Lauvset, S. K., Lefèvre, N., Lienert, S., Liu, J., Marland, G., McGuire, P. C., Melton, J. R., Munro, D. R., Nabel, J. E. M. S., Nakaoka, S.-I., Niwa, Y., Ono, T., Pierrot, D., Poulter, B., Rehder, G., Resplandy, L., Robertson, E., Rödenbeck, C., Rosan, T. M., Schwinger, J., Schwingshackl, C., Séférian, R., Sutton, A. J., Sweeney, C., Tanhua, T., Tans, P. P., Tian, H., Tilbrook, B., Tubiello, F., van der Werf, G. R., Vuichard, N., Wada, C., Wanninkhof, R., Watson, A. J., Willis, D., Wiltshire, A. J., Yuan, W., Yue, C., Yue, X., Zaehle, S., and Zeng, J. Global carbon budget 2021. *Earth System Science Data*, 14(4):1917–2005, 2022.

Fuller, A., Millard, K., and Green, J. R. Croma: Remote sensing representations with contrastive radar-optical masked autoencoders. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Fuller, A., Millard, K., and Green, J. Croma: Remote sensing representations with contrastive radar-optical masked autoencoders. *Advances in Neural Information Processing Systems*, 36, 2024.

Garnot, V. S. F. and Landrieu, L. Panoptic segmentation of satellite image time series with convolutional temporal attention networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4872–4881, 2021.

Gaydon, C. and Roche, F. Pureforest: A large-scale aerial lidar and aerial imagery dataset for tree species classification in monospecific forests. *arXiv preprint arXiv:2404.12064*, 2024.

Girdhar, R., Singh, M., Ravi, N., Van Der Maaten, L., Joulin, A., and Misra, I. Omnivore: A single model for many visual modalities. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16102–16112, 2022.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

Guzhov, A., Raue, F., Hees, J., and Dengel, A. Audioclip: Extending clip to image, text and audio. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 976–980. IEEE, 2022.

Hassani, A., Walton, S., Li, J., Li, S., and Shi, H. Neighborhood attention transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6185–6194, 2023.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

Ienco, D., Interdonato, R., Gaetano, R., and Minh, D. H. T. Combining sentinel-1 and sentinel-2 satellite image time series for land cover mapping via a multi-source deep learning architecture. *ISPRS Journal of Photogrammetry and Remote Sensing*, 158:11–22, 2019.

Jaegle, A., Gimeno, F., Brock, A., Vinyals, O., Zisserman, A., and Carreira, J. Perceiver: General perception with iterative attention. In *International conference on machine learning*, pp. 4651–4664. PMLR, 2021.

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q., Sung, Y.-H., Li, Z., and Duerig, T. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pp. 4904–4916. PMLR, 2021.

Karras, T., Aittala, M., Aila, T., and Laine, S. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35: 26565–26577, 2022.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4015–4026, 2023.

Kussul, N., Lavreniuk, M., Skakun, S., and Shelestov, A. Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geoscience and Remote Sensing Letters*, 14(5):778–782, 2017.

Lang, N., Jetz, W., Schindler, K., and Wegner, J. D. A high-resolution canopy height model of the earth. *Nature Ecology & Evolution*, 7(11):1778–1789, 2023.

Lehner, J., Alkin, B., Fürst, A., Rumetshofer, E., Miklautz, L., and Hochreiter, S. Contrastive tuning: A little help to make masked autoencoders forget. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 2965–2973, 2024.

Li, S., Brandt, M., Fensholt, R., Kariryaa, A., Igel, C., Gieseke, F., Nord-Larsen, T., Oehmcke, S., Carlsen, A. H., Junttila, S., et al. Deep learning enables image-based tree counting, crown segmentation, and height prediction at national scale. *PNAS nexus*, 2(4):pgad076, 2023.

Li, S., Liu, Z., Tian, J., Wang, G., Wang, Z., Jin, W., Wu, D., Tan, C., Lin, T., Liu, Y., et al. Switch ema: A free lunch for better flatness and sharpness. *arXiv preprint arXiv:2402.09240*, 2024.

Liu, S., Brandt, M., Nord-Larsen, T., Chave, J., Reiner, F., Lang, N., Tong, X., Ciais, P., Igel, C., Pascual, A., et al. The overlooked contribution of trees outside forests to tree cover and woody biomass across europe. *Science Advances*, 9(37):eadh4097, 2023.

Liu, X., Su, Y., Hu, T., Yang, Q., Liu, B., Deng, Y., Tang, H., Tang, Z., Fang, J., and Guo, Q. Neural network guided interpolation for mapping canopy height of china's forests by integrating gedi and icesat-2 data. *Remote Sensing of Environment*, 269:112844, 2022.

Loshchilov, I. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., and Li, T. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 508:293–304, 2022.

Minderer, M., Gritsenko, A., Stone, A., Neumann, M., Weissenborn, D., Dosovitskiy, A., Mahendran, A., Arnab, A., Dehghani, M., Shen, Z., et al. Simple open-vocabulary object detection. In *European Conference on Computer Vision*, pp. 728–755. Springer, 2022.

Morin, D., Planells, M., Mermoz, S., and Mouret, F. Estimation of forest height and biomass from open-access multi-sensor satellite imagery and gedi lidar data: high-resolution maps of metropolitan france. *arXiv preprint arXiv:2310.14662*, 2023.

N. Baghdadi, M. Bernier, R. G. and Neeson, I. Evaluation of c-band sar data for wetlands mapping. *International Journal of Remote Sensing*, 22(1):71–88, 2001. doi: 10.1080/014311601750038857. URL https://doi.org/10.1080/014311601750038857.

Noman, M., Naseer, M., Cholakkal, H., Anwer, R. M., Khan, S., and Khan, F. S. Rethinking transformers pre-training for multi-spectral satellite imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27811–27819, 2024.

Pauls, J., Zimmer, M., Kelly, U. M., Schwartz, M., Saatchi, S., Ciais, P., Pokutta, S., Brandt, M., and Gieseke, F. Estimating canopy height at scale. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=ZzCY0fRver.

Pauls, J., Zimmer, M., Turan, B., Saatchi, S., Ciais, P., Pokutta, S., and Gieseke, F. Capturing temporal dynamics in large-scale canopy tree height estimation. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=ri1cs3vtXX.

Pei, R., Liu, J., Li, W., Shao, B., Xu, S., Dai, P., Lu, J., and Yan, Y. Clipping: Distilling clip-based models with a student base for video-language retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18983–18992, 2023.

Pelletier, C., Webb, G., and Petitjean, F. Temporal convolutional neural network for the classification of satellite image time series. *Remote Sensing*, 11(5):523, March 2019. ISSN 2072-4292. doi: 10.3390/rs11050523. URL http://dx.doi.org/10.3390/rs11050523.

Press, O., Smith, N. A., and Levy, O. Train short, test long: Attention with linear biases enables input length extrapolation. *Proceedings of the NeurIPS*, 2022. doi: 10.48550/arXiv.2108.12409.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

Ranftl, R., Bochkovskiy, A., and Koltun, V. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 12179–12188, 2021.

Reed, C. J., Gupta, R., Li, S., Brockman, S., Funk, C., Clipp, B., Keutzer, K., Candido, S., Uyttendaele, M., and Darrell, T. Scale-mae: A scale-aware masked autoencoder for multiscale geospatial representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4088–4099, 2023.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Schultz, M. G., Betancourt, C., Gong, B., Kleinert, F., Langguth, M., Leufen, L. H., Mozaffari, A., and Stadtler, S. Can deep learning beat numerical weather prediction? *Philosophical Transactions of the Royal Society A*, 379 (2194):20200097, 2021.

Schwartz, M., Ciais, P., De Truchis, A., Chave, J., Ottlé, C., Vega, C., Wigneron, J.-P., Nicolas, M., Jouaber, S., Liu, S., et al. Forms: Forest multiple source height, wood volume, and biomass maps in france at 10 to 30 m resolution based on sentinel-1, sentinel-2, and global ecosystem dynamics investigation (gedi) data with a deep learning approach. *Earth System Science Data*, 15(11): 4927–4945, 2023.

Shaw, P., Uszkoreit, J., and Vaswani, A. Self-attention with relative position representations. In *Proceedings of the NAACL-HLT*, 2018. doi: 10.18653/v1/N18-2074.

Shen, K., Ju, Z., Tan, X., Liu, Y., Leng, Y., He, L., Qin, T., Zhao, S., and Bian, J. Naturalspeech 2: Latent diffusion models are natural and zero-shot speech and singing synthesizers. *arXiv preprint arXiv:2304.09116*, 2023.

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., and Woo, W.-c. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28, 2015.

Singh, A., Hu, R., Goswami, V., Couairon, G., Galuba, W., Rohrbach, M., and Kiela, D. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15638–15650, 2022.

Singh, M., Duval, Q., Alwala, K. V., Fan, H., Aggarwal, V., Adcock, A., Joulin, A., Dollár, P., Feichtenhofer, C., Girshick, R., et al. The effectiveness of mae pre-pretraining for billion-scale pretraining. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5484–5494, 2023.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. PMLR, 2015.

Srivastava, S. and Sharma, G. Omnivec: Learning robust representations with cross modal sharing. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 1236–1248, 2024.

Su, J., Lu, Y., Pan, S., et al. Roformer: Enhanced transformer with rotary position embedding. *Proceedings of the ICLR*, 2021. doi: 10.48550/arXiv.2104.09864.

Tan, X., Qin, T., Bian, J., Liu, T.-Y., and Bengio, Y. Regeneration learning: A learning paradigm for data generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 22614–22622, 2024.

Tolan, J., Yang, H.-I., Nosarzewski, B., Couairon, G., Vo, H. V., Brandt, J., Spore, J., Majumdar, S., Haziza, D., Vamaraju, J., et al. Very high resolution canopy height maps from rgb imagery using self-supervised vision transformer and convolutional decoder trained on aerial lidar. *Remote Sensing of Environment*, 300:113888, 2024.

Topouzelis, K., Singha, S., and Kitsiou, D. Incidence angle normalization of wide swath sar data for oceanographic applications. *Open Geosciences*, 8(1):450–464, 2016. doi: doi:10.1515/geo-2016-0029. URL https://doi.org/10.1515/geo-2016-0029.

van den Oord, A., Vinyals, O., and Kavukcuoglu, K. Neural discrete representation learning. *CoRR*, abs/1711.00937, 2017. URL http://arxiv.org/abs/1711.00937.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need, 2017. URL https://arxiv.org/abs/1706.03762.

Wagner, F. H., Dalagnol, R., Silva-Junior, C. H., Carter, G., Ritz, A. L., Hirye, M. C., Ometto, J. P., and Saatchi, S. Mapping tropical forest cover and deforestation with

planet nicfi satellite images and deep learning in mato grosso state (brazil) from 2015 to 2021. *Remote Sensing*, 15(2):521, 2023.

Wang, Y., Albrecht, C. M., Braham, N. A. A., Liu, C., Xiong, Z., and Zhu, X. X. Decoupling common and unique representations for multimodal self-supervised learning. In *European Conference on Computer Vision*, pp. 286–303. Springer, 2024.

Xiong, Z., Wang, Y., Zhang, F., Stewart, A. J., Hanna, J., Borth, D., Papoutsis, I., Le Saux, B., Camps-Valls, G., and Zhu, X. X. Neural plasticity-inspired foundation model for observing the earth crossing modalities. *arXiv e-prints*, pp. arXiv–2403, 2024.

Xue, Y., Gan, E., Ni, J., Joshi, S., and Mirzasoleiman, B. Investigating the benefits of projection head for representation learning. *arXiv preprint arXiv:2403.11391*, 2024.

Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., and Wu, Y. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

Yuan, L., Chen, D., Chen, Y.-L., Codella, N., Dai, X., Gao, J., Hu, H., Huang, X., Li, B., Li, C., et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

Yuan, Y., Lin, L., Liu, Q., Hang, R., and Zhou, Z.-G. Sits-former: A pre-trained spatio-spectral-temporal representation model for sentinel-2 time series classification. *International Journal of Applied Earth Observation and Geoinformation*, 106:102651, 2022.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. Barlow twins: Self-supervised learning via redundancy reduction. In *International conference on machine learning*, pp. 12310–12320. PMLR, 2021.

Zhang, C., Sargent, I., Pan, X., Li, H., Gardiner, A., Hare, J., and Atkinson, P. M. Joint deep learning for land cover and land use classification. *Remote sensing of environment*, 221:173–187, 2019.

Zhang, S., Zhu, F., Yan, J., Zhao, R., and Yang, X. Zero-cl: Instance and feature decorrelation for negative-free symmetric contrastive learning. In *International Conference on Learning Representations*, 2021.

A. Implementation Details

In the main text, we provided a high-level overview of the architecture and training setup. Here, we include additional implementation details for key modules and components of our framework. These details were omitted from the main body for brevity and are provided here for completeness and reproducibility.

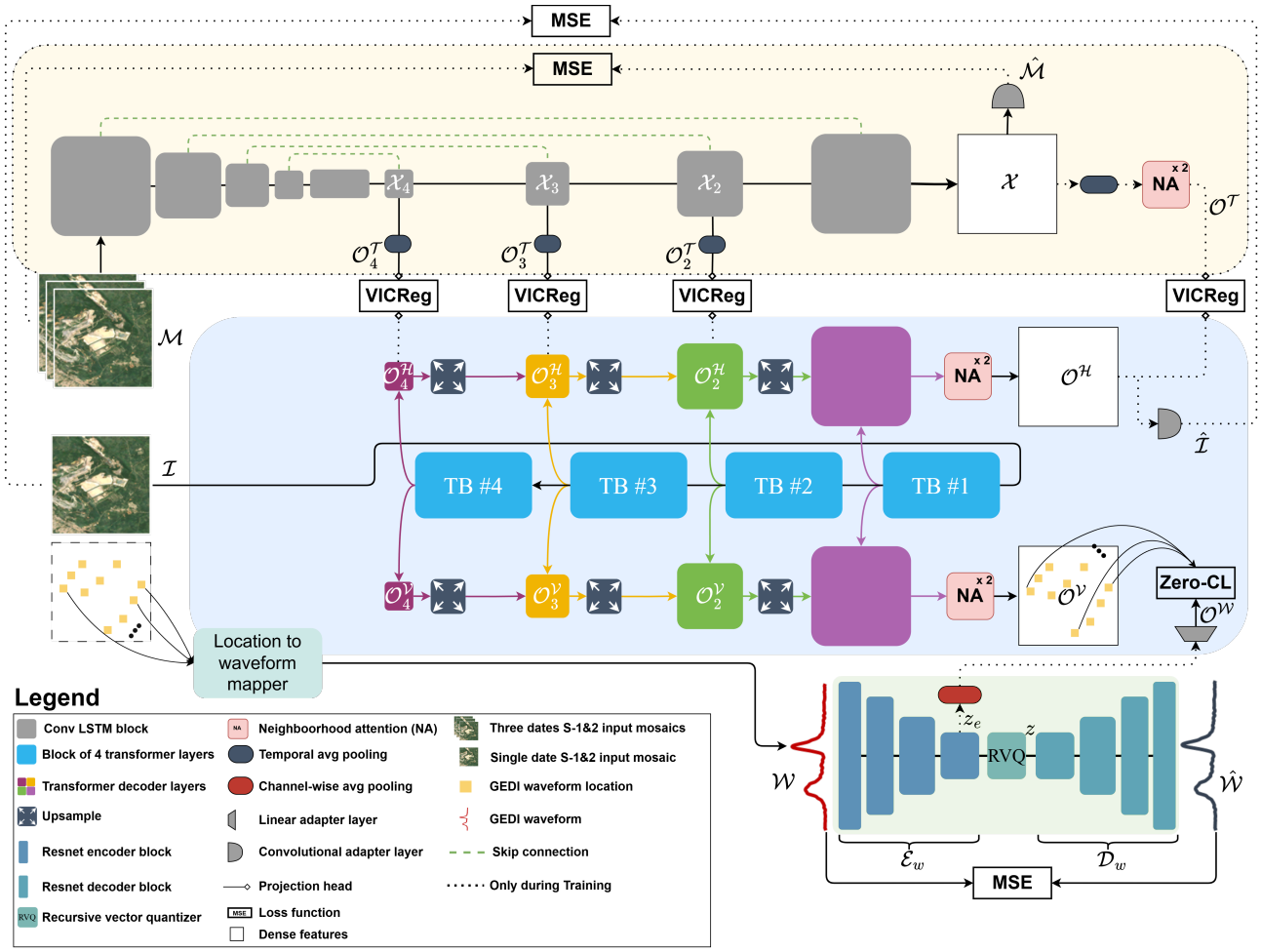

Figure A1. An overview of our framework showcasing the main blocks and connections. The pre-trained model, which take as input a single date S-1 & 2 image mosaic, is shaded in blue. The multi-temporal image AE, which take as input a multi-date S-1& 2 image mosaic, is shaded in yellow. The vertical structure AE, which take as input the GEDI waveforms, is shaded in green.

A.1. Pre-Trained Model

Patch Embedding. To generate patch embeddings, the 2D composite image $\mathcal{I} \in \mathbb{R}^{H\times W\times C}$ is processed using a convolutional layer with kernel size (P, P) and stride P. This operation partitions $\mathcal{I}$ into $N = HW/P^2$ non-overlapping patches, which are reshaped into a sequence of embeddings in $\mathbb{R}^{N\times D}$. Here, C represents the number of input channels, (H, W) denotes the resolution of the input image, P is the patch size, and D is the patch embedding dimension.

Unlike tokenization strategies that rely on fixed-size input grids, the use of convolutions allows the model to adapt to varying input resolutions. This is particularly useful in EO applications, where spatial resolutions can vary across datasets. While transformer-based architectures address sequence length variability through techniques like rotary embeddings (Su et al., 2021), ALiBi (Press et al., 2022), and relative position encodings (Shaw et al., 2018), convolutional patch embedding provides a structured inductive bias that benefits spatial feature extraction while maintaining translational invariance.

Decoder Architecture. The decoding process begins by reshaping the encoded token sequence from $\mathbb{R}^{N\times D}$ into an image-like feature representation at different resolutions. At each decoder stage $d \in 1, 2, 3, 4$, tokens are rescaled to $\frac{H}{2^{d-1}} \times \frac{W}{2^{d-1}} \times D_{p_d}$, where $D_{p_d} = 2^{d-1}D_p$ represents the channel dimension at each stage. This transformation involves an initial 1×1 convolution to adjust channel dimensions, followed by strided 3×3 convolutions for downsampling when $2^{d-1} \geq P$, or transpose convolutions when $2^{d-1} < P$.

To reconstruct high-resolution embeddings, feature maps from deeper layers ($d = 4$) are progressively upsampled using bilinear interpolation and refined through two 3×3 Conv-BatchNorm-ReLU operations before being concatenated with corresponding features from earlier transformer layers. A final 1×1 convolution ensures that the final output dimension matches D_p when required. This architecture is instantiated as two separate decoders, outputting $\mathcal{O}^{\mathcal{V}}$ and $\mathcal{O}^{\mathcal{H}}$.

Projection Head. For the pixel-pixel alignment objective, we apply a lightweight two-layer 1×1 convolutional projection head to each output $\mathcal{O}^{\mathcal{H}}_d$. The first layer expands the feature dimension by a factor of two, followed by a second layer that projects it back to D_p. This design enhances feature expressiveness during training but, as observed in prior work (Xue et al., 2024), projection heads tend to overfit to the pretraining task, reducing their transferability to downstream applications. Consequently, we discard the projection head after training. In contrast, the pixel-waveform alignment output $\mathcal{O}^{\mathcal{V}}$ is directly used without an additional projection head, as its representation remains well-aligned with downstream tasks such as waveform retrieval and generation.

A.2. Waveform Model

Waveform Encoder. The waveform encoder consists of an input stem and three stages designed to progressively reduce the 1D waveform of length L (i.e., $\mathcal{W} \in \mathbb{R}^{L\times 1}$) while increasing its feature representation capacity. First, the input stem processes the waveform using a 3×1 convolutional layer with a stride of 2. This initial operation halves the length of the waveform to $L/2$ and increases the channel dimension to 2. Following the input stem, the encoder comprises three stages, each containing three ResNet blocks. Each ResNet block is designed with two 3×1 convolutional layers, batch normalization, ReLU activations, and a skip connection. At the beginning of each stage, the first ResNet block incorporates a strided convolution in its initial layer to halve the waveform's length and double the channel dimension. The subsequent blocks within the same stage operate with a stride of 1. By the end of the three stages, the waveform encoder reduces the input waveform's length to $L/16$ while progressively increasing the channel dimension to 16. Thus, the waveform encoder $\mathcal{E}_w$ transforms a waveform of shape $\mathbb{R}^{L\times 1}$ into a latent representation $z \in \mathbb{R}^{L/16\times 16}$.

The alignment of the waveforms and the pixels is performed on this latent representation $z_e = \mathcal{E}_w(\mathcal{W})$. However, to adapt them to the outputs of $\mathcal{O}^{\mathcal{V}}$, we first apply an average pooling along the channel dimension, followed by a projection to the $\mathcal{O}^{\mathcal{V}}$ projection dimension (i.e., D_p) using a linear layer. In essence, this process transforms z_e into $\mathcal{O}^{\mathcal{W}} \in \mathbb{R}^{D_p}$.

Residual Vector Quantizer (RVQ). The RVQ layer takes a segment from the latent waveform representation $z^i_e \in \mathbb{R}^{1/16}$, where z^i_e is the i^{th} row of the waveform latent representation, and iteratively quantizes it. It operates by first mapping each z^i_e to the closest codebook vector in the first quantizer, then computing the residual (i.e., the difference between the input vector and the quantized approximation). This residual is passed to the next quantizer in the sequence, and the process is repeated for all Q quantizers. At the end of the process, the latent waveform representation z_e is converted into a discrete series of quantized residual vectors, which are summed to produce the final quantized waveform representation z.

Waveform Decoder. The waveform decoder ($\mathcal{D}_w$) mirrors the structure of the encoder but operates in reverse order to reconstruct the waveform from its quantized latent representation z. Similar to the encoder, the decoder consists of three stages and an output stem. However, instead of using strided convolutions for downsampling, the decoder replaces them with transpose convolutions to progressively upsample the waveform. Each stage in the decoder increases the length of the waveform while halving the channel dimension, reversing the transformations applied by the encoder. By the end of the decoding process, the latent representation is transformed back into a waveform $\hat{\mathcal{W}}$ with length L.

A.3. Multitemporal Image Processing Model

Multitemporal Image AE. The AE has four processing and downsampling stages. In each stage we replace the conventional double Conv2D-BatchNorm-ReLU layers in UNets with a two-layer ConvLSTM followed by a BatchNorm layer and ReLU activation layer. This design choice is to explicitly model temporal correlations across time steps as mentioned earlier. Downsampling is performed using a strided Conv3D layer applied on the spatial dimension, while

upsampling is performed using a trilinear upsampling operation also on the spatial dimension, followed by a Conv3D layer. As such, the model takes a multi-temporal input image $\mathcal{M} \in \mathbb{R}^{T\times H\times W\times C}$, where T is the temporal dimension, and produces a feature map $\mathcal{X} \in \mathbb{R}^{T\times H\times W\times D_p}$. Finally, as this is an autoencoder, the original input shape is recovered using a Conv3D layer producing $\hat{\mathcal{M}} \in \mathbb{R}^{T\times H\times W\times C}$. All Conv3D layers in the AE have a kernel of 3×3 on the spatial dimension.

B. Objective Functions.

Zero-CL Loss. Let $Z^A \in \mathbb{R}^{G\times D_p}$ represent a L_2 normalized pixel embeddings from $\mathcal{O}^{\mathcal{V}}$, where G is the number of waveforms in a minibatch and D_p is the feature dimension, and $Z^B \in \mathbb{R}^{G\times D_p}$ denote their corresponding L_2 normalized waveform embeddings from $\mathcal{O}^{\mathcal{W}}$. The instance-wise contrastive loss is defined as:

$$\mathcal{L}_{\text{Ins}} = \sum_{i=1}^{G}\left(1 - \sum_{d=1}^{D_p} H_{i,d}^{A,\text{Ins}} \cdot H_{i,d}^{B,\text{Ins}}\right)^2 \tag{3}$$

$H_{i,d}$ represents the d^{th} feature value of the i^{th} instance. H^{Ins} is a zero-phase component analysis (ZCA) whitened embedding matrix defined as:

$$H^{\text{Ins}} = W^{\text{Ins}} Z, \quad W^{\text{Ins}} = E^{\text{Ins}} \Lambda_S^{-1/2} E^\top \tag{4}$$

where $E^{\text{Ins}} \in \mathbb{R}^{G,G}$ and Λ_S are the eigenmatrix and diagonal of the eigenvalue matrix of the *affinity matrix* $S = \frac{1}{D_p} ZZ^\top$, respectively. Similar to the instance-wise contrastive objective, the feature-wise contrastive loss is formulated as:

$$\mathcal{L}_{\text{Fea}} = \sum_{d}^{D_p}\left(1 - \sum_{i}^{G} H_{i,d}^{A,\text{Fea}} \cdot H_{i,d}^{B,\text{Fea}}\right)^2 \tag{5}$$

Where H^{Fea} is defined as:

$$H^{\text{Fea}} = W^{\text{Fea}} Z^\top, \quad W^{\text{Fea}} = E \Lambda_C^{-1/2} E^\top \tag{6}$$

where $E \in \mathbb{R}^{D_p,D_p}$ and Λ_C are the eigenmatrix and diagonal of the eigenvalue matrix of the *covariance matrix* $C = \frac{1}{G} Z^\top Z$, respectively.

The overall loss $\mathcal{L}_{Zero-CL}$ is $\mathcal{L}_{\text{Fea}} + \mathcal{L}_{\text{Ins}}$.

VICReg Loss. Let $Z^{\mathcal{H}} \in \mathbb{R}^{M\times D_p}$ represent an L_2 normalized pixel embedding from $\mathcal{O}^{\mathcal{H}}$ and $Z^{\mathcal{T}} \in \mathbb{R}^{M\times D_p}$ its corresponding L_2 normalized pixel embeddings from $\mathcal{O}^{\mathcal{T}}$, with $M = B \times H \times W$ the number of pixels in the mini-batch of size B. VICReg is expressed as a sum of three losses: a variance loss, an invariance loss, and a covariance loss. The variance loss is formulated as:

$$\begin{aligned}
&\mathcal{L}_{\text{var}} = \frac{1}{2}(v(Z^{\mathcal{H}}) + v(Z^{\mathcal{T}})),\\
&v(Z) = \frac{1}{D_p}\sum_{d}^{D_p} \max\left(0, 1 - \sqrt{\frac{1}{M}\sum_{i}^{M}(Z_{i,d} - \overline{Z_d})^2 + \epsilon}\right)
\end{aligned} \tag{7}$$

where $Z_{i,d}$ represents the d^{th} feature value of the i^{th} instance.

The invariance loss is formulated as:

$$\mathcal{L}_{\text{inv}} = \frac{1}{M}\sum_{i=1}^{M} \|Z_i^H - Z_i^T\|_2^2 \tag{8}$$

The covariance loss is formulated as:

$$\begin{aligned} \mathcal{L}_{\text{cov}} &= c(Z^{\mathcal{H}}) + c(Z^{\mathcal{T}}), \\ c(Z) &= \frac{1}{D_p} \sum_{k \neq l} [cov(Z)]_{k,l}^2 \\ cov(Z) &= \frac{1}{M-1} \sum_{i}^{M} (Z_i - \overline{Z})(Z_i - \overline{Z})^\top \end{aligned} \tag{9}$$

The overall loss term $\mathcal{L}_{\text{VICReg}}$ is then:

$$\mathcal{L}_{\text{VICReg}} = \alpha_v \mathcal{L}_{\text{var}} + \beta_i \mathcal{L}_{\text{inv}} + \gamma_c \mathcal{L}_{\text{cov}} \tag{10}$$

Following Bardes (2021) we set α_v and β_i to 25.0 and γ_c to 1.0.

C. Waveform Generation

Denoising diffusion models (DM) (Sohl-Dickstein et al., 2015) are generative models that learn a data distribution p_{data} by gradually denoising a noisy variable following a predefined noise schedule. Let ϵ_θ be a neural network. Its goal is to predict the original data x from a noisy version x_t generated by $x_t = \alpha_t x_0 + \sigma_t \epsilon$, where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ is Gaussian noise, α_t and σ_t are parameters of a noise scheduling function, and t is a time step from $\{1, ..., T\}$.

The corresponding objective of a DM can then be written as:

$$L_{DM} = \mathbb{E}_{x_0 \sim p_{data}, \epsilon \sim \mathcal{N}(0,\mathbf{I}), t} \left[\|y - \epsilon_\theta(x_t, t)\|_2^2 \right] \tag{11}$$

where p_{data} denotes the data distribution over the clean inputs x_0, and the target y can be the input noise ϵ, the original input x variable, or the velocity $v = \alpha_t \epsilon - \sigma_t x$. DMs can also be parametrized by a condition $c \in \mathbb{R}^D$ in addition to the diffusion time t, and the corresponding objective becomes:

$$L_{CDM} = \mathbb{E}_{x_0 \sim p_{data}, \epsilon \sim \mathcal{N}(0,1), t, c} \left[\|y - \epsilon_\theta(x_t, t, c)\|_2^2 \right] \tag{12}$$

Another class of diffusion models, known as Latent Diffusion Models (LDMs) (Rombach et al., 2022; Shen et al., 2023; Ramesh et al., 2022), operate in a latent space rather than directly in the high-dimensional data space. By operating in a lower-dimensional latent space, LDMs significantly reduce the computational cost of training and inference compared to DMs, which operate in the original data space. LDMs leverage the latent representation of data obtained by training an autoencoder, defined by an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$. The autoencoder is trained such that $\hat{x} = \mathcal{D}(\mathcal{E}(x))$ is a reconstruction of the original data x. After training the autoencoder, a diffusion model is trained directly on the encoded data samples $z = \mathcal{E}(x)$. A new sample is then obtained by first sampling a representation z, and then $\mathcal{D}(z)$ yields x.

Following the LDM formulation, we learn a network ϵ_θ conditioned on $\mathcal{O}^V_{\phi,\lambda}$, where ϕ and λ are the waveform coordinates, that directly predicts a denoised waveform latent $z_0 \in \mathbb{R}^{\frac{L}{16} \times 16}$, where $z_0 = \mathcal{E}(\mathcal{W})$, by minimizing:

$$L_{LDM} = \mathbb{E}_{z_0 \sim p(z_0), \epsilon \sim \mathcal{N}(0,1), t \sim U[1,T], \mathcal{O}^V_{\phi,\lambda}} \left[\left\| \epsilon_\theta(z_t, t, \mathcal{O}^V_{\phi,\lambda}) - z_0 \right\|_2^2 \right] \tag{13}$$

In our formulation, the network $\epsilon_\theta(z_t, t, \mathcal{O}^V_{\phi,\lambda})$ is a 1D UNet model which takes the current noisy latent z_t, the time step t as input, and the condition $\mathcal{O}^V_{\phi,\lambda}$ is integrated into the UNet via cross-attention at each layer. To sample z_0 from the diffusion model, we start from a Gaussian noise z_T and gradually denoise it into samples $z_{T-1}, ..., z_0$ using the ordinary differential equation (ODE) solver proposed by Karras et al. (2022), as we found that it produces high-quality samples with fewer denoising steps. To make the sampled latent waveform z_0 better match its conditioning we use classifier-free guidance. Specifically, during training we randomly drop the conditioning for 50% of the examples to make the model capable of conditional and unconditional denoising. In practice, we replace the conditioning signal with a random vector. Within the classifier-free guidance formulation, model predictions can be written as:

$$\hat{\epsilon_\theta}(z_t, t, \mathcal{O}^V_{\phi,\lambda}) = \epsilon_\theta(z_t, t) + s \cdot (\epsilon_\theta(z_t, t, \mathcal{O^V}_{\phi,\lambda}) - \epsilon_\theta(z_t, t)) \tag{14}$$

where s is the guidance scale. Setting s to 0 is equivalent to unconditional sampling, while setting it to ≥ 1 has shown to produce more coherent but less diverse results. Here we set s to 3. Finally, to synthesize a waveform at any location (ϕ, λ) given a pixel embedding $\mathcal{O}^V_{\phi,\lambda}$ as a condition, we use eq. 14 to get $\hat{z_0}$, and leverage the frozen waveform decoder $\mathcal{D}_w$, where $\mathcal{D}_w(\hat{z_0})$ is a generated waveform.

D. Visual Comparison: Pixel vs. Patch Embeddings

To highlight the difference in spatial granularity between our proposed pixel-level embedding model and a patch-based baseline (e.g., CROMA), we present a side-by-side qualitative visualization in Figure D1. Both models are applied to the same S-1 & 2 input, and their embeddings are projected to RGB using t-SNE. The pixel-based model captures finer spatial structures and preserves object boundaries more effectively than the patch-based baseline, yielding lower-resolution outputs.

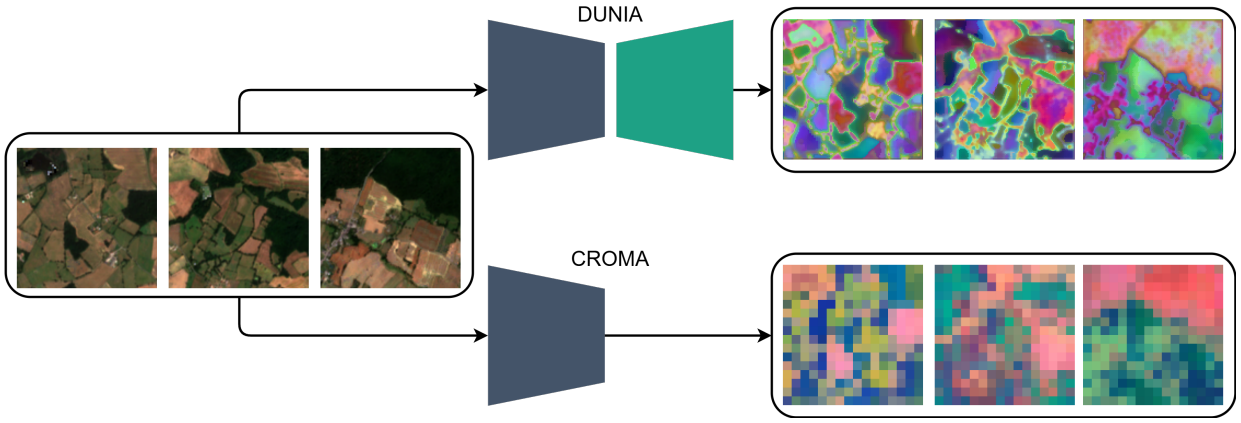

Figure D1. Comparison between pixel-level embeddings from our proposed model (top row) and patch-based embeddings from one of the baseline models (bottom row), both generated from the same S-1 & 2 inputs (left). Embeddings are visualized using t-SNE.

E. Model Configuration and Optimization

E.1. Model Configuration

The image model is pre-trained on 64×64 pixel sized images with 14 bands, a 16-layer transformer with 8 attention heads, a patch size of eight, 512 embedding dimensions, a feedforward layer with two linear layers with a hidden dimension of 2048, GeGLU activation, and 0.1 dropout and attention dropout rates. The two image decoders are identical and project the decoded image feature map to 64 embedding dimensions (i.e. $D_p = 64$). Finally, we set the window size w of the NA layers to 19 pixels. The waveform AE is trained on 256 bin-long waveforms, with RVQ layers composed of eight quantizers, and each quantizer containing 512 codebook entries. As such, the waveform encoder encodes the waveforms into a 16×16 latent representation which is average pooled along the channel dimension and projected to 64 embeddings dimension for the pixel-waveform alignment. The multi-temporal image AE is trained on three 64×64 pixel sized input images, each with 14 bands, and produces a similarly shaped feature map on the temporal and spatial dimensions, and 64 embeddings dimension. This output is then average-pooled on the temporal dimension for the pixel-pixel alignment. Finally, the diffusion model is trained on the 16×16 quantized waveform latents, which are first projected to 128 channels before being passed to the UNet model.

E.2. Training, Inference, and Generation

DUNIA was pre-trained on a single NVIDIA A6000 48GB GPU with a batch size of 60 for 250K steps using the Lion optimizer (Chen et al., 2024), a learning rate of $5\text{e}{-5}$, weight decay of 0.4, 5K warmup steps, and a cosine annealing schedule. For finetuning, we used AdamW (Loshchilov, 2017) with a $2\text{e}{-4}$ learning rate, a cosine annealing schedule, a batch size of 20, and trained until plateau. The diffusion model was trained with a batch size of 4096 for 100K steps using AdamW, a $1\text{e}{-4}$ learning rate, 5K warmup steps, and a cosine annealing schedule. During inference, diffusion steps were set to 30. Both DUNIA and the diffusion model were regularized with the Switch EMA (SEMA) technique (Li et al., 2024),

maintaining an EMA with a decay rate of 0.9, updating every 5 steps, and replacing online model parameters every 1K steps with a SEMA coefficient of 0.9.

F. Experimental Settings

Sentinel-2 Data. We used Level-2A surface reflectance data (S2-L2A) from Google Earth Engine (GEE). The dataset includes bands at 10m (Blue, Green, Red, and NIR) and 20m (Red Edge 1-4, SWIR1, and SWIR2) spatial resolutions, all upscaled to 10m ground sampling distance (GSD) using cubic interpolation. We created two sets of mosaics: a single mosaic during the leaf-on season using the median of all available images from April to September 2020. This mosaic is used as input for the pre-trained model. The other set is used as input for the multitemporal AE and is comprised of three mosaics, each representing the median acquisitions of all images in a four-month span from October 2019 until September 2020. Cloudy pixels were filtered out using the S-2 Cloud Probability dataset provided by SentinelHub in GEE.

Sentinel-1 Data. Sentinel-1 data were obtained from the Sentinel-1A and Sentinel-1B satellites, operating at the C-band (6 cm wavelength). Images were collected in Interferometric Wide swath mode (IW) with VV and VH polarizations, derived from the high-resolution Level-1 ground range detected (GRD) product. The original 20 m × 22 m resolution was resampled to 10 m × 10 m GSD. Similar to S-2, we also created two sets of mosaics. The S-1 data were calibrated using the Sentinel SNAP toolbox, converting pixel values to backscattering coefficients (σ^0) in linear units and applying geometric correction using the 30m Shuttle Radar Topography Mission (SRTM) Digital Elevation Model (DEM). Finally, the backscattering coefficients σ^0_θ acquired at difference incidence angles θ were normalized to a common reference incidence angle set at $\sigma_{ref} = 40^\circ$ using the cosine correction equation (N. Baghdadi & Neeson (2001); Topouzelis et al. (2016)):

$$\sigma_ref^0 = \frac{\sigma^0_\theta cos^2(\theta_{ref})}{cos^2(\theta)} \tag{15}$$

GEDI Data. GEDI measures the vertical structure of objects (e.g., vegetation) using three lasers emitting near-infrared light (1064 nm), one of the laser beams is split in two, and after applying optical dithering, this results in eight ground tracks spaced 600 m apart, with each shot having a 25 m footprint. For this study, we used GEDI Level 1B (L1B), Level 2A (L2A), and Level 2B (L2B) data from April 2019 to December 2021. From the L1B product we extracted the waveforms and their geolocation (i.e., latitude, longitude, and elevation), and the geolocation of the GEDI instrument for each shot. Finally, as the waveforms are stored as vectors of 1420 bins (1 bin = 0.15 m), we also extracted for each waveform the bin count, the signal start ($\mathcal{W}_{start}$) and signal end ($\mathcal{W}_{end}$). The latter two indicate the location of the canopy top and the end position of the waveform before the noise, respectively. From the L2A data product, we extracted the RH_{98} (referred to $\mathcal{W}_{rh}$), which represents the estimated height of the tallest object within the waveform footprint, the canopy cover ($\mathcal{W}_c$). Finally, from the L2B data product, we extracted the Plant Area Index ($\mathcal{W}_{pai}$).

As the waveforms are affected by the atmospheric conditions at acquisition time, we followed the filtering scheme of Fayad et al. (2024) to remove low-quality waveforms. Finally, due to the different waveform shapes between the leaf-off and leaf-on seasons, especially for those acquired over deciduous forests, we only considered waveforms acquired during the leaf-on season. This resulted in a GEDI dataset of ≈19 million waveforms and their associated metrics.

PureForest (PF). The PureForest dataset was designed as a benchmark and provides ground truth patches for the classification of mono-specific forests in France (Gaydon & Roche, 2024). It includes high-resolution imagery and corresponding annotations for more than 13K 50×50 m forest patches for 13 tree species.

CLC+Backbone (CLS_+). The CLS_+ A pan-European wall-to-wall land cover inventory for the 2021 reference year. The product is based on Sentinel 2 (i.e., optical) time series from 2020 to 2022 and a temporal convolutional neural network (TempCNN) (Pelletier et al., 2019) serving as a classifier. The product is available as a 10 m raster and shows for each pixel the dominant land cover among the 11 basic land cover classes.

PASTIS dataset. A crop mapping dataset by Garnot (2021) for 18 crop classes and 1 background class from the French Land Parcel Information System. The dataset contains 2433 128×128 pixels densely annotated patches at 10 m resolution.

Vertical Structure dataset. For vertical structure evaluation, we evaluated our model on its ability to map at 10 m resolution: (1) forest heights ($\mathcal{W}_{rh}$), forest fractional canopy cover ($\mathcal{W}_c$), plant area index ($\mathcal{W}_{pai}$), and complete waveform ($\mathcal{W}$) retrieval or generation. For these products, we rely on the products derived from the GEDI dataset presented earlier.

G. Impact of Label Quantity on Performance

Table G1. Top-1 retrieval-based zero-shot classification performance of DUNIA for different database sizes. and are DUNIA query embeddings from $\mathcal{O}^{\mathcal{V}}$ and $\mathcal{O}^{\mathcal{H}}$ respectively. S represents the number of samples in the retrieval database, with *im* meaning a 64×64 pixels fully annotated image, and *l* meaning a single annotated pixel. KNN_b represents the best KNN value for a given dataset. $\mathcal{W}^{**}$ represents performance results for vertical structures higher than 5 m.

DATASET	METRIC	S	KNN_b	100% S	10% S	5% S
$\mathcal{W}_{rh}$	RMSE (r)	50K *l* *	50	2.0 (.93)	2.1 (.92)	2.1 (.92)
$\mathcal{W}_c$	RMSE (r)	50K *l*	50	11.7 (.89)	12.4 (.86)	12.0 (.84)
$\mathcal{W}_{pai}$	RMSE (r)	50K *l*	50	0.71 (.75)	0.72 (.75)	0.72 (.74)
CLC_+	*w*F1	500 *im*	50	80.1	75.2	70.2
$PASTIS$	OA	500 *im*	50	56.2	52.4	48.3
PF	*w*F1	50K *l*	5	76.0	73.5	70.9
$\mathcal{W}^{**}$	r	50K *l*	1	.70	.67	.66

Table G2. Fine-tuning performance of DUNIA and the five competing models. and are DUNIA's embeddings from $\mathcal{O}^{\mathcal{V}}$ and $\mathcal{O}^{\mathcal{H}}$ respectively. S represents the number of samples used for the fine-tuning, with *im* meaning a 64×64 pixels fully annotated image, and *l* meaning a single annotated pixel. $\mathcal{W}^{**}$ represents performance results for vertical structures higher than 5 m. Best scores are in **bold**.

DATASET	METRIC	SAMPLES (S)	DUNIA	ANYSAT	CROMA	DOFA	DECUR	SATMAE
$\mathcal{W}_{rh}$	RMSE (r)	1K *im*	**1.4** (.93)	2.8 (.89)	3.6 (.76)	11.2 (.50)	11.1 (.52)	10.5 (.52)
$\mathcal{W}_c$	RMSE (r)	1K *im*	**10.0** (.83)	12.4 (.79)	14.5 (.72)	30.1 (.48)	29.4 (.47)	30.7 (.47)
$\mathcal{W}_{pai}$	RMSE (r)	1K *im*	**0.61** (.70)	0.94 (.67)	1.5 (.60)	1.7 (.35)	1.7 (.36)	1.9 (.37)
CLC_+	*w*F1	1K *im*	89.4	**89.5**	85.9	71.1	74.6	73.8
$PASTIS$	*w*F1	300K *im*	75.3	**80.2**	71.0	52.2	56.4	54.8
PF	*w*F1	10K *l*	80.1	80.0	**80.2**	79.8	78.5	78.7
$\mathcal{W}^{**}$	r	≈3.8M *l*	.75	—	—	—	—	—

H. Ablation Studies

H.1. Loss Choice

We used two different loss functions for the Pixel-Pixel alignment and Pixel-Waveform alignment. Table H1 shows that the cosine similarity (CS) using the VICReg loss for the pixel-waveform alignment only reaches a maximum of 0.56 for the positive pairs and 0.10 for the negative pairs, indicating poor alignment. On the other hand, using the ZERO-CL loss, the CS between the positive pairs reaches 0.86, and 0.35 between the negative pairs. We attribute the poor results for Pixel-Waveform alignment using the VICReg loss to the low number of available waveforms within a mini-batch. In contrast, for the Pixel-Pixel alignment, the CS for the positive pairs is 0.99, and 0.04 for the negative pairs using the VICReg loss. It decreases to 0.98 for the positive pairs and 0.45 for the negative pairs with ZERO-CL. Nevertheless, using ZERO-CL for both modalities increased training times exponentially due to the whitening transformation and the high number of available pixels within the mini-batch.

Table H1. Cosine similarity between positive pairs (+) and negative pairs (-) during pre-training. P represents a pixel embedding, W represents a waveform embedding. Positive Pairs consist of a pixel embedding and its corresponding pixel/waveform embedding, while negative pairs are computed between a pixel embedding and other pixel/waveform embeddings within the mini-batch.

Loss	+P↔P	-P↔P	+P↔W	-P↔W
VICReg	0.99	0.04	0.56	0.10
ZERO-CL	0.98	0.45	0.86	0.35

H.2. Using a Shared Decoder

In our formulation, we used two different decoders for horizontal and vertical structural understanding. We hypothesized that a single real-valued embedding cannot simultaneously encode contrasting information, such as trees of the same species with different vertical structures. Table H2 Shows that in the retrieval case, training a single decoder produces pixel embeddings with no semantic overlap with the waveforms with an average CS of -0.42, while the retrieved pixel embeddings given a pixel input are still highly similar. In the case of training separate decoders, retrieval performance is drastically different, with high correlations (0.99) between pixels and the retrieved waveforms.

Table H2. Average cosine similarity ($\overline{CS}$) between retrieved pixels and pixel queries (P←P) and between retrieved waveforms and pixel queries (P←W) for KNN=1.

	P←P $\overline{CS}$	P←W $\overline{CS}$
Shared decoder	0.97	-0.42
Separate decoders	0.98	0.99

H.3. Hierarchical VICReg Loss

The original VICReg loss proposed by Bardes et al. (2021) was applied between two views of the same input image. They later extended their work and applied it to local image features (Bardes et al., 2022). Their results showed that better results were obtained when applying the VICReg loss on local features and at the instance level. Here, we further extend this work and apply it hierarchically to the decoder outputs between the pre-trained model and the multi-temporal AE. Results in Table H3 show significant improvements for both datasets when using this formulation. These results are also in line with the findings in Bardes et al. (2022).

Table H3. Zero-shot classification performance (wF1) on the PF and CLC_+ datasets using a model pre-trained with multiple VICReg losses at different decoder levels (VICReg$_h$) against a single loss applied on the embeddings from the last decoder layer (VICReg$_s$).

	PF	CLC_+
VICReg$_h$	76.0	80.1
VICReg$_s$	67.6	74.2

H.4. Neighborhood Attention

The output of our pre-trained model is composed of two neighborhood attention layers per output head, instead of the standard convolutional layers. This design choice allows each embedding to be modeled based on its local neighborhood instead of relying on small and fixed receptive fields. Table H4 shows an increase of 2.1% (wF1) for the CLC_+ dataset and an RMSE decrease of 0.4 m for the $\mathcal{W}_{rh}$ dataset using this new configuration.

Table H4. Performance differences between a convolutional output layer (CNN) vs. a NA layer on the CLC_+ and $\mathcal{W}_{rh}$ datasets.

	CLC_+	$\mathcal{W}_{rh}$
CNN	72.1 (wF1)	2.4 (RMSE)
NA	74.2 (wF1)	2.0 (RMSE)

H.5. Embedding sensitivity

Our results, either in the zero-shot setting or the fine-tuned setting used embeddings from $\mathcal{O}^{\mathcal{H}}$ for horizontal structure-related products (e.g., tree species identification, land cover mapping) and embeddings from $\mathcal{O}^{\mathcal{V}}$ for vertical structure-related products (e.g., canopy height mapping, waveform retrieval). We perform the same tests in the zero-shot setting, but reversing the query and retrieval embeddings for the other structure direction. Table H5 shows that relying on vertical structure data for products requiring horizontal understanding and vice versa performs poorly for the six tested products. This validates our design choice of cross-modal alignment with vertical structure data for EO products relying on this type of information.

Table H5. Top-1 retrieval-based zero-shot classification performance of DUNIA with opposite direction query and retrieval embeddings. OG represents the original results from Table 1 for KNN=50. and are DUNIA query embeddings from $\mathcal{O}^{\mathcal{V}}$ and $\mathcal{O}^{\mathcal{H}}$ respectively.

DATASET	METRIC	OG	DUNIA
$\mathcal{W}_{rh}$	RMSE (r)	2.0 (.93)	4.3 (.63)
$\mathcal{W}_{c}$	RMSE (r)	11.7 (.89)	18.9 (.53)
$\mathcal{W}_{pai}$	RMSE (r)	0.71 (.75)	1.6 (.42)
CLC_{+}	*w*F1	80.1	35.1
$PASTIS$	OA	56.2	0.0
PF	*w*F1	73.8	56.1

H.6. Inference Runtime (Zero-Shot)

Generating maps in the zero-shot setting requires: 1) a forward pass through the pre-trained model (encoder-dual decoders) to obtain the embeddings, 2) for each pixel, retrieve the k nearest neighbors (KNN) from the retrieval database (DB), and 3) classify the queried pixel embedding based on distance-weighted voting. As such, generating a map over e.g. a 20.48×20.48 Km area at 10 m resolution, requires querying and classifying 4,194,304 pixels. However, Table H6 shows that the querying and classification times only add a small overhead. Moreover, the time required for the three operations (forward pass, retrieval, and classification) is still faster than a similarly capable model like AnySat.

Table H6. Inference runtime (in seconds) of DUNIA in the zero-shot setting, compared to AnySat. DB denotes the retrieval database of N key/value pairs (embedding/label), and KNN is the number of nearest neighbors retrieved per pixel. All times are in seconds.

MODEL	DB	KNN	FORWARD PASS	RETRIEVAL	CLASSIFICATION	TOTAL
DUNIA	256K	100	2.52	0.36	1.34	4.22
DUNIA	512K	200	2.52	0.40	1.88	4.80
ANYSAT	—	—	177.37	—	—	177.37

H.7. Impact of Input Size

To assess the sensitivity of DUNIA to the input image resolution, we conducted experiments using three different input sizes: 128×128, 256×256, and 512×512. As shown in Table H7, the results across datasets and metrics remain unchanged. This confirms that image size has a negligible impact on the final performance of the model in the zero-shot setting.

Table H7. Zero-shot performance of DUNIA at different input image sizes for several datasets.

DATASET	METRIC	128×128	256×256	512×512
$\mathcal{W}_{rh}$	RMSE (r)	2.2	2.0	2.1
$\mathcal{W}_{c}$	RMSE (r)	11.6	11.7	11.7
$CLC+$	*w*F1	80.2	80.1	80.2

H.8. Temporal Stability

To evaluate the temporal stability of DUNIA, we conducted two experiments on several vertical structural products across the years 2019, 2020, and 2021. Other products were excluded due to the unavailability of labels for different years.

1. Fine-tuning an MLP head on top of a model pre-trained in 2020, and evaluating on 2019 and 2021.
2. Pre-training DUNIA using combined data from 2019–2021 and evaluating it in the zero-shot setting.

Table H8 shows that fine-tuning on individual years yields consistent performance across time. Similarly, Table H9 demonstrates that zero-shot performance is also stable over several years.

Table H8. Fine-tuned performance (RMSE) on vertical structure variables estimation across several years.

DATASET	METRIC	2019	2020	2021
$\mathcal{W}_{rh}$	RMSE	1.35	1.34	1.40
$\mathcal{W}_{c}$	RMSE	9.8	9.8	10.1
$\mathcal{W}_{pai}$	RMSE	0.65	0.62	0.63

Table H9. Zero-shot performance (RMSE) on vertical structure variables retrieval across years.

DATASET	METRIC	2019	2020	2021
$\mathcal{W}_{rh}$	RMSE	2.4	2.0	2.1
$\mathcal{W}_{c}$	RMSE	11.6	11.7	11.9
$\mathcal{W}_{pai}$	RMSE	0.75	0.71	0.74

H.9. Impact of S-2 and S-2 Input Images

Table H10 shows that combining both modalities yields the best performance for height estimation and tree species identification, while land cover classification is relatively robust to the choice of modality.

Table H10. Performance of DUNIA when using S-1 only, S-2 only, or both modalities. Results are reported for height prediction (RMSE), land cover classification (wF1) on the CLC_+ dataset, and tree species identification (wF1) on the PF dataset. Performance results obtained in the fine-tuned setting.

DATASET	METRIC	S-1 ONLY	S-2 ONLY	S-2 & 1
$\mathcal{W}_{rh}$	RMSE	3.80	2.80	1.34
PF	wF1	65.5%	81.2%	82.2%
CLC_+	wF1	90.2%	90.0%	90.3%

H.10. Median Composites vs. Single-Date Images

Results on height estimation and crop classification ($PASTIS$) (Table H11) show that median composites — even without full time series — significantly outperform single-date imagery by capturing more stable and representative reflectance patterns over time.

Table H11. Comparison between single-date and median composite inputs for products where temporal information is critical. Performance results obtained in the fine-tuned setting.

DATASET	METRIC	SINGLE-DATE IMAGE	MEDIAN COMPOSITE
$\mathcal{W}_{rh}$	RMSE	1.9	1.3
$PASTIS$	wF1	42.3	77.0

I. Further Results

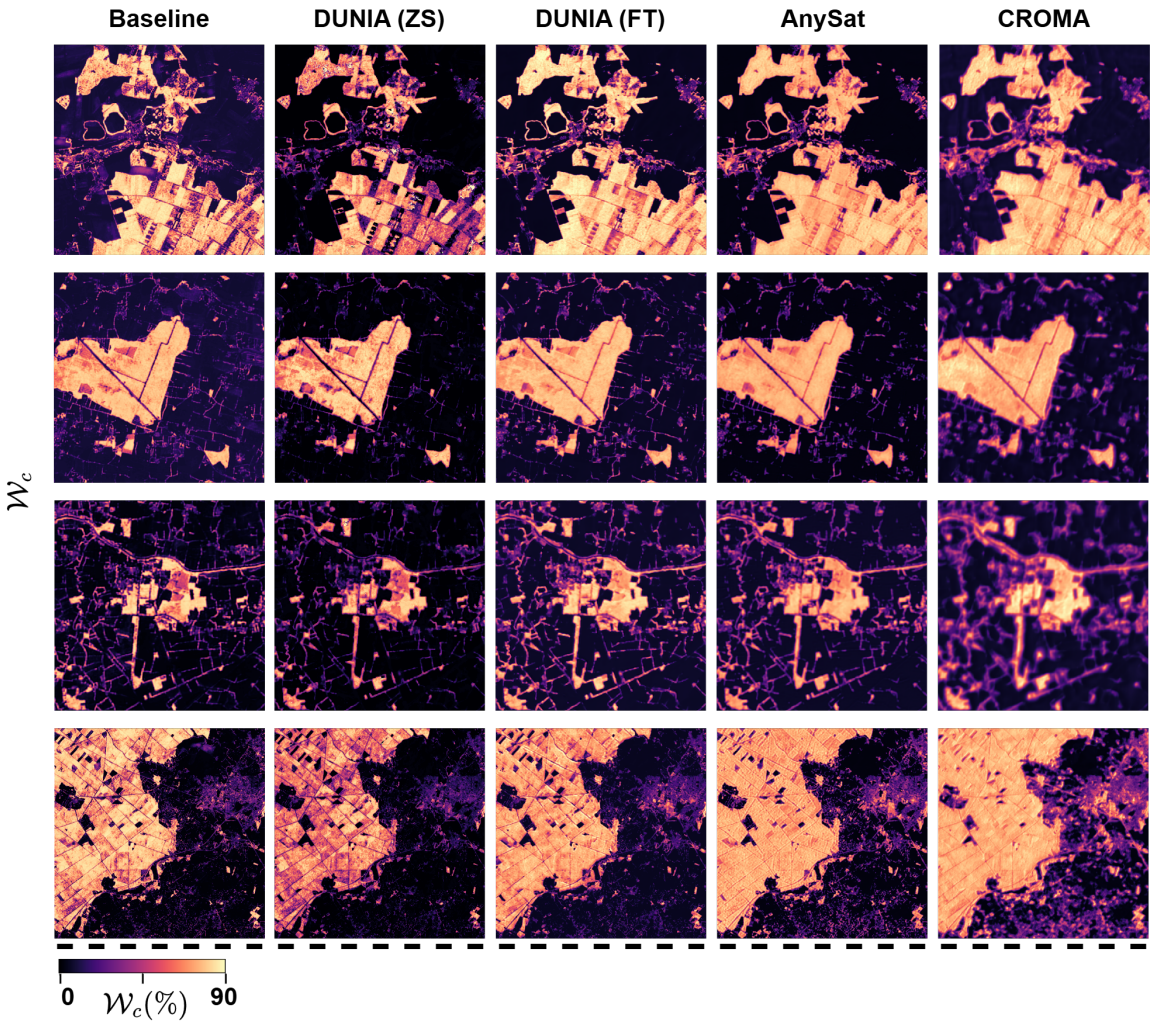

Figure I1. Fractional canopy cover ($\mathcal{W}_c$) maps produced with different models. Baseline represents reference maps from adapted FORMS (Schwartz et al., 2023). DUNIA (ZS) represents maps obtained in the zero-shot setting, while DUNIA (FT) represents maps obtained in the fined-tuned setting. Best viewed zoomed-in (300+%).

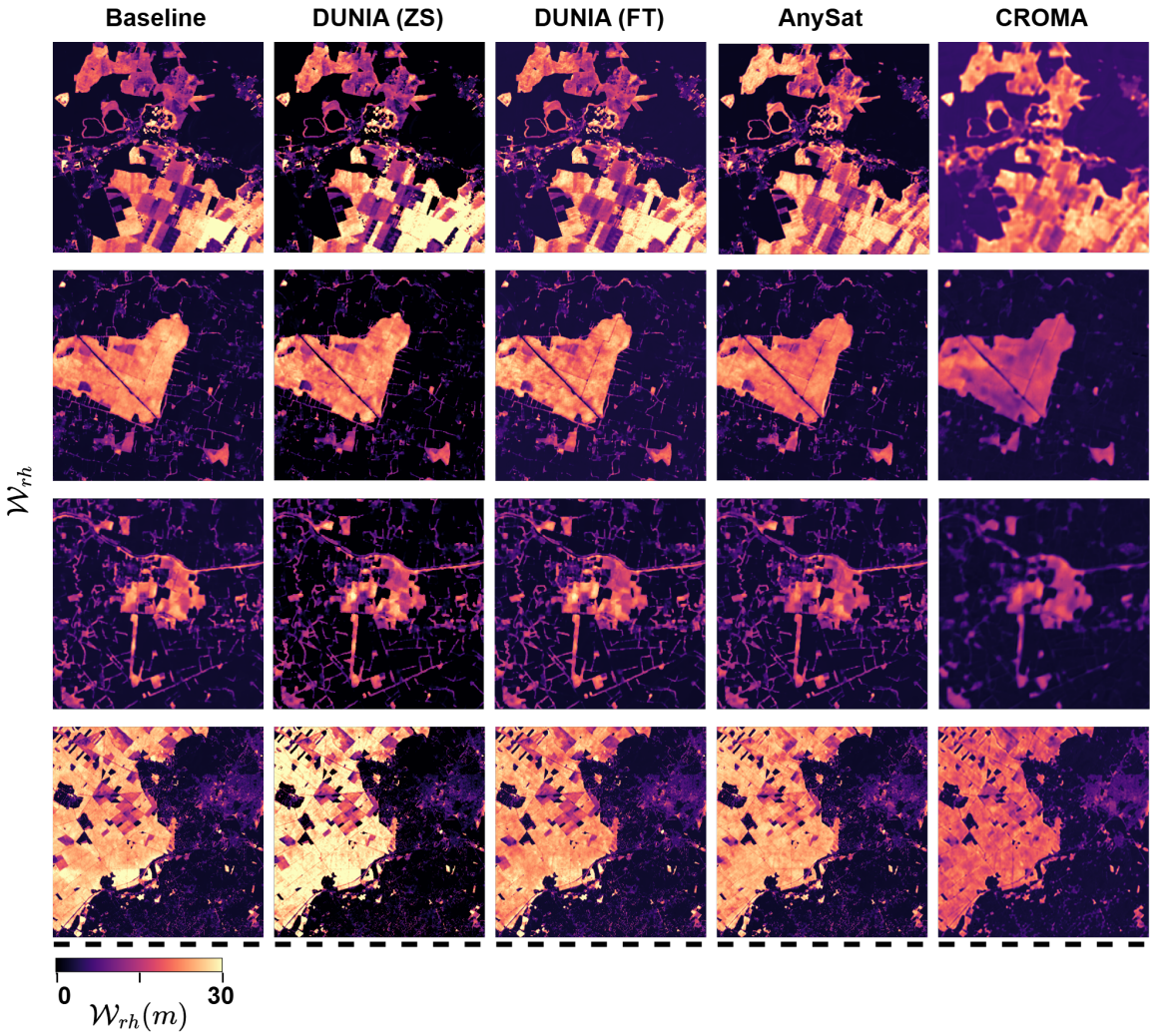

Figure I2. Canopy height ($\mathcal{W}_{rh}$) maps produced with different models. Baseline represents reference maps from FORMS (Schwartz et al., 2023). DUNIA (ZS) represents maps obtained in the zero-shot setting, while DUNIA (FT) represents maps obtained in the fined-tuned setting. Best viewed zoomed-in (300+%).

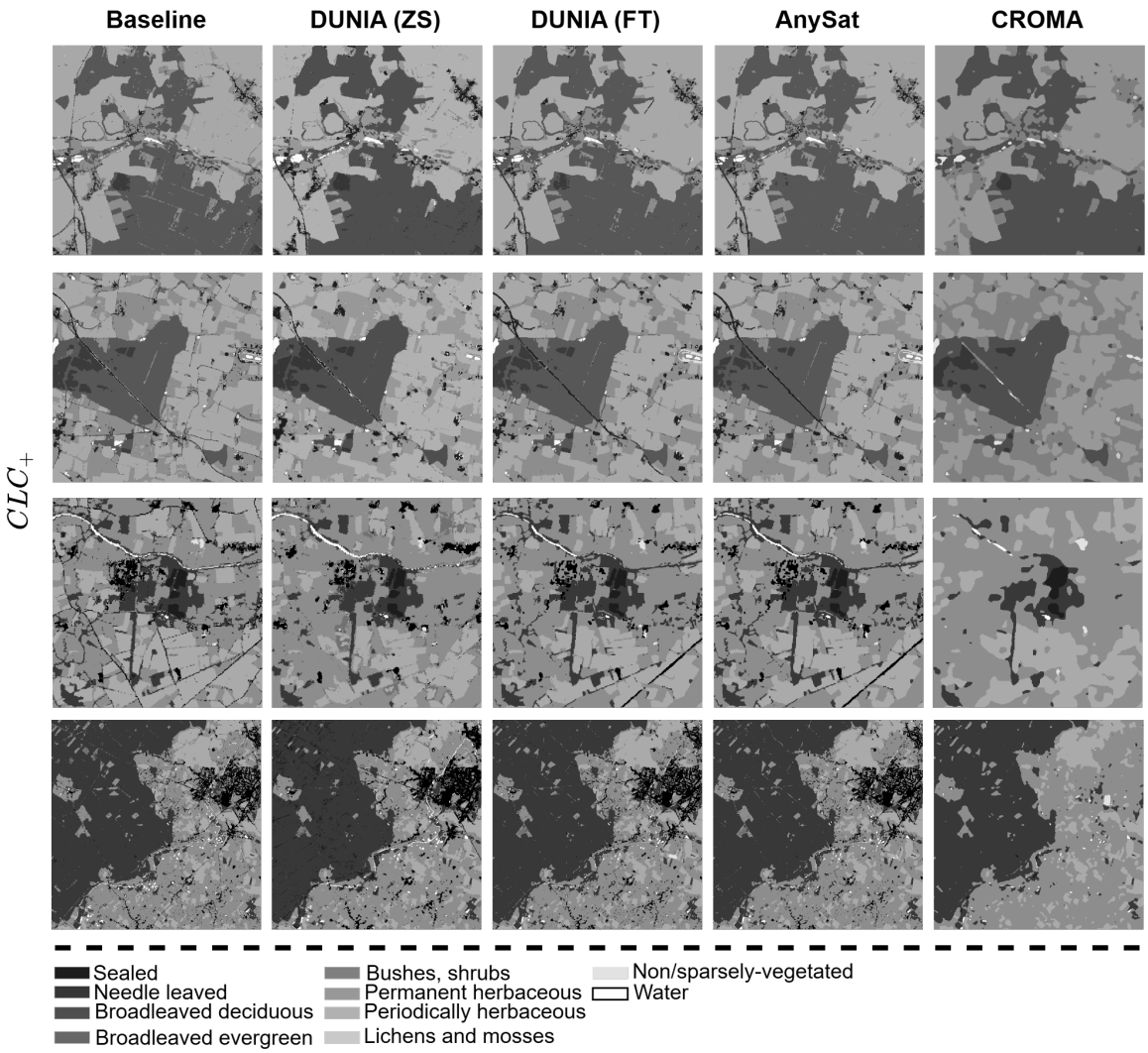

Figure I3. Land cover classes (CLC_+) maps produced with different models. Baseline represents reference maps from the European Environment Agency (2024). DUNIA (ZS) represents maps obtained in the zero-shot setting, while DUNIA (FT) represents maps obtained in the fined-tuned setting. Best viewed zoomed-in (300+%).

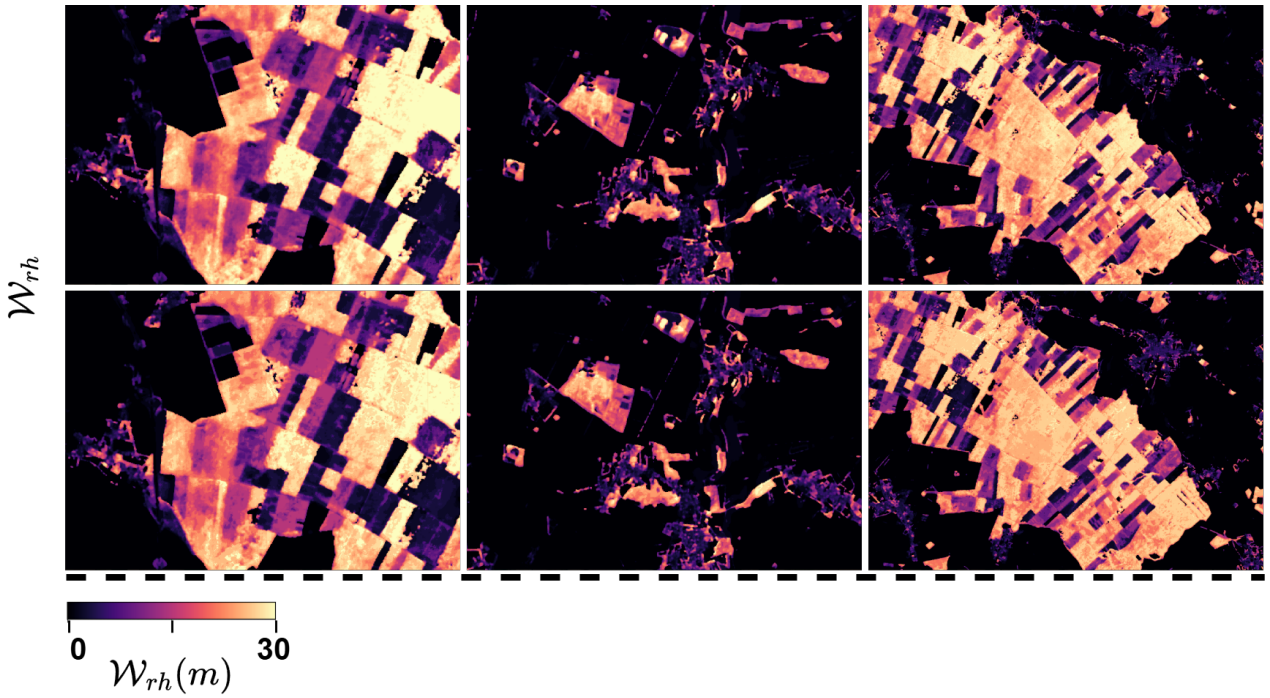

Figure I4. Comparison of map qualities for the task of estimating the canopy height in the zero-shot setting using 50K l GEDI samples as queries (top row) vs. only 10K l (bottom row).

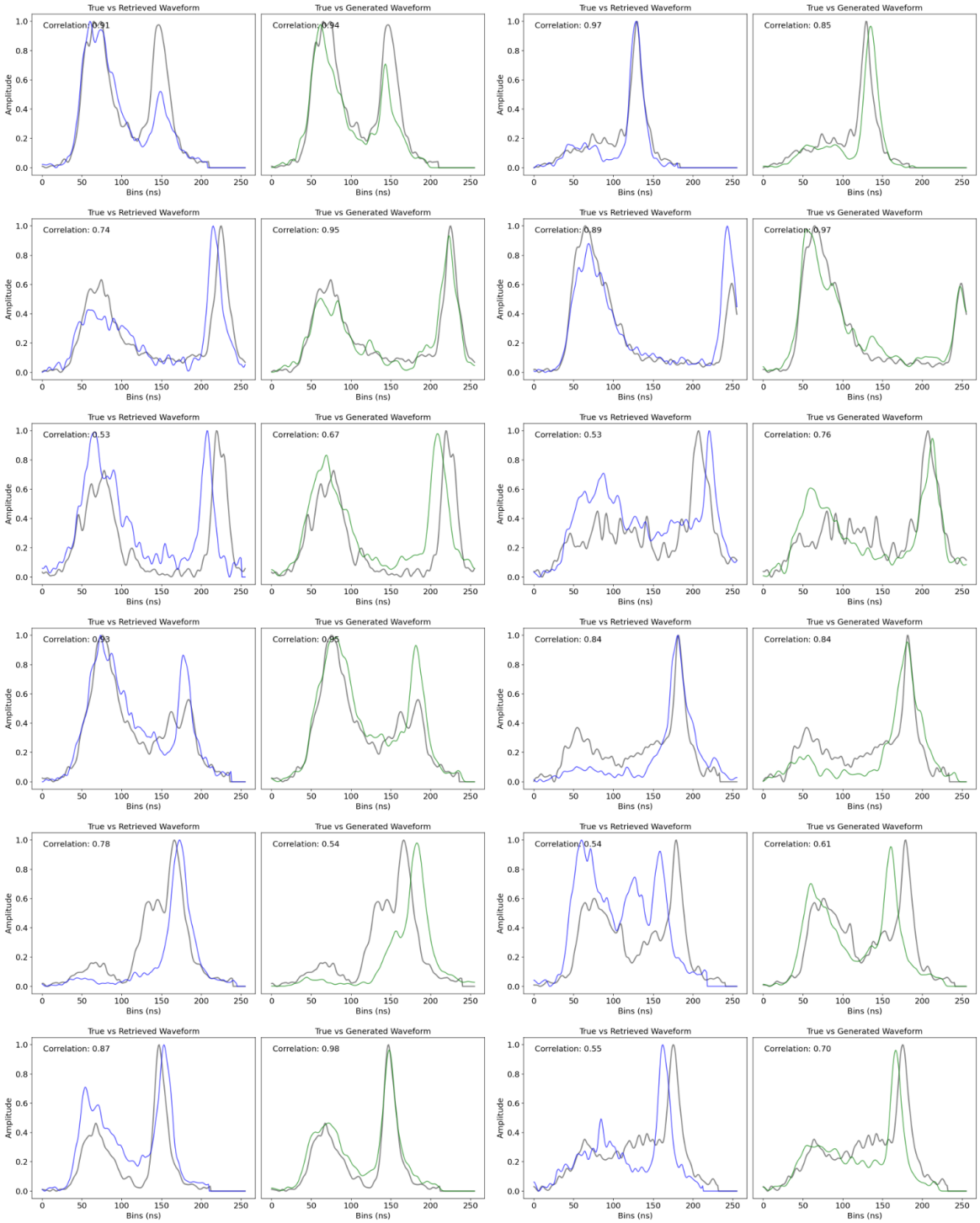

Figure I5. Uncurated list of retrieved ( ) and generated waveforms ( ) overlayed on a reference waveform ( ). 'Correlation' is Pearson's correlation coefficient (r) between the reference and the retrieved/generated waveforms. Best viewed zoomed-in (200+%).

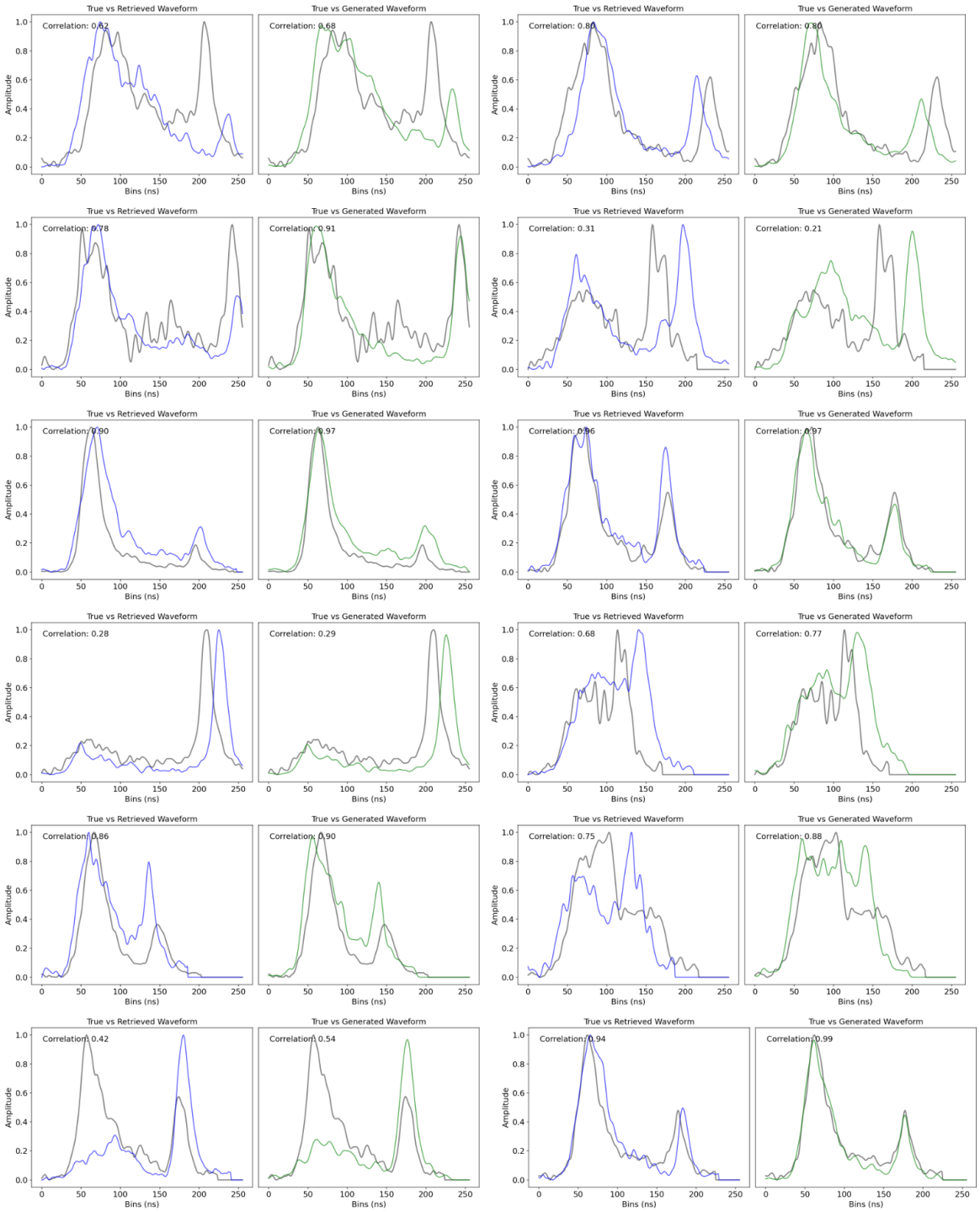

Figure I6. Uncurated list of retrieved ( ) and generated waveforms ( ) overlayed on a reference waveform ( ). 'Correlation' is Pearson's correlation coefficient (r) between the reference and the retrieved/generated waveforms. Best viewed zoomed-in (200+%).

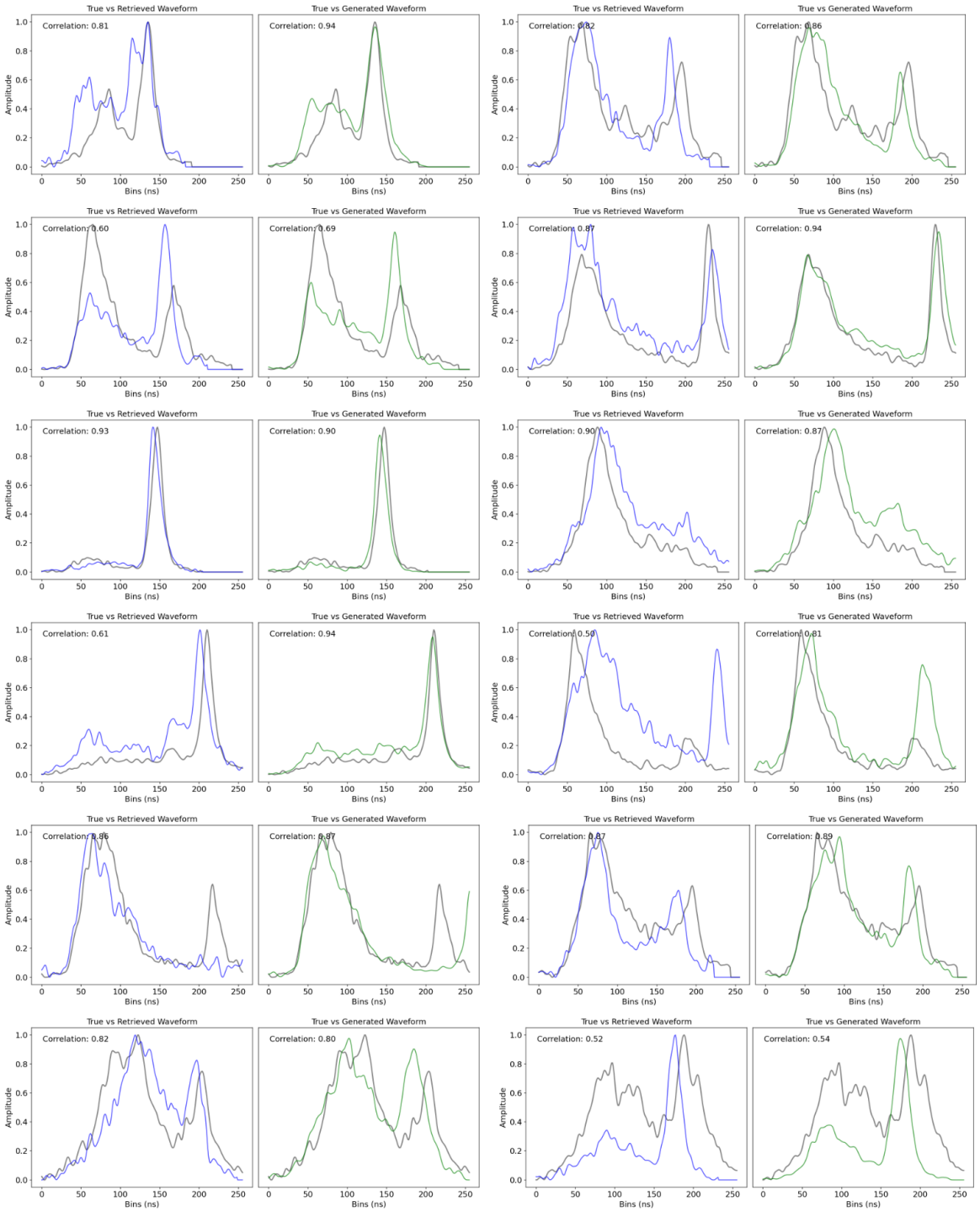

Figure I7. Uncurated list of retrieved ( ) and generated waveforms ( ) overlayed on a reference waveform ( ). 'Correlation' is Pearson's correlation coefficient (r) between the reference and the retrieved/generated waveforms. Best viewed zoomed-in (200+%).

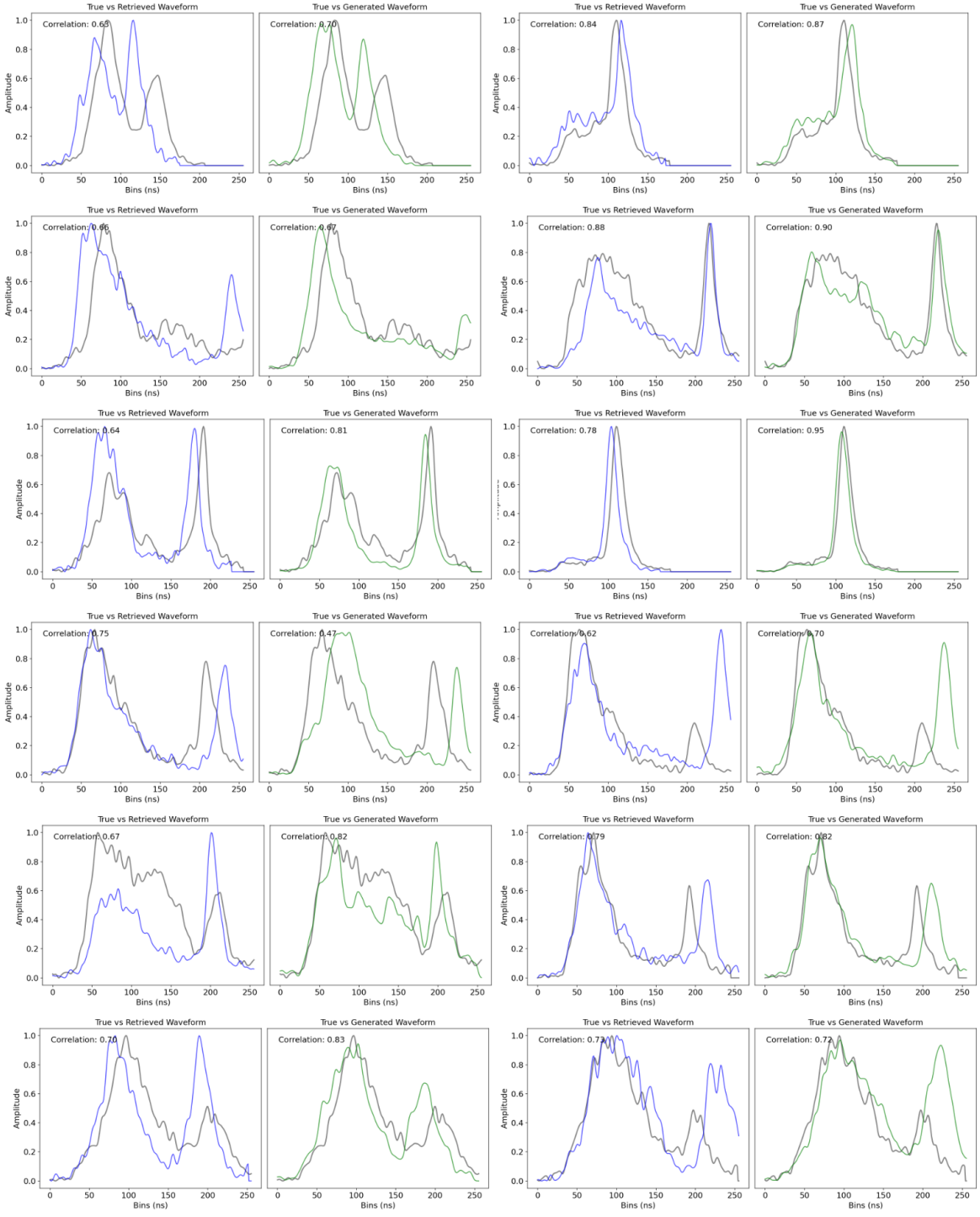

Figure I8. Uncurated list of retrieved ( ) and generated waveforms ( ) overlayed on a reference waveform ( ). 'Correlation' is Pearson's correlation coefficient (r) between the reference and the retrieved/generated waveforms. Best viewed zoomed-in (200+%).