Earth Observation Foundation Model PhilEO: Pretraining on the MajorTOM and FastTOM Datasets

Nikolaos Dionelis, Riccardo Musto, Jente Bosmans, Simone Sarti, Giancarlo Paoletti, Peter Naylor, Valerio Marsocci, Sébastien Lefèvre, Bertrand Le Saux, Nicolas Longépé

***Abstract*—Today, Earth Observation (EO) satellites generate massive volumes of data. To fully exploit this, it is essential to pretrain Foundation Models (FMs) on large unlabeled datasets, enabling efficient fine-tuning for downstream tasks with minimal labeled data. We study scaling-up FMs, where scaling refers to incorporating a larger pretraining dataset that includes both oceans and ice in addition to land (MajorTOM, 23TB), a larger model in terms of number of parameters (200M), and an architecture that achieves top performance, efficiency, and low computational complexity, being capable of handling images covering large areas. We develop various models using different architectures, including U-Net Convolutional Neural Networks (CNN), Vision Transformers (ViT), and Mamba State-Space Models (SSM), and different numbers of parameters. We evaluate the FLoating-point OPerations (FLOPs) needed by the models. ViT, due to attention, suffers from quadratic complexity, while Mamba achieves linear complexity, scaling-up better. We present the scaling of the model PhilEO on MajorTOM, covering the vast majority of Earth, as well as on the specialized subset FastTOM 2TB that does not include oceans and ice. We fine-tune the models on the PhilEO Bench for different downstream tasks: roads, buildings, and land cover. For all n-shots for roads, the Geo-Aware U-Net 44M-23T model outperforms U-Net 44M-0.5T. For most n-shots for roads and buildings, U-Net 200M-2T outperforms the other models. We show that using Mamba models, we achieve comparable results on the downstream tasks, with less computational expenses. We also compare with the recent FM TerraMind which we evaluate on the PhilEO Bench.**

***Index Terms*—Earth Observation, Foundation Models, remote sensing, Vision Transformer (ViT), Mamba state-space model.**

I. Introduction

GIVEN the vast amounts of satellite data coming from multiple constellations, such as Copernicus Sentinel-2, the focus has recently been shifted to Self-Supervised Learning (SSL). Today, Earth Observation (EO) satellites generate massive volumes of data, with the Copernicus Sentinel-2 constellation alone producing approximately 1.6TB per day. To fully exploit this information captured by remote sensing satellites, it is essential to pretrain EO Foundation Models (FMs) using SSL on *large* unlabeled datasets, enabling efficient fine-tuning for various downstream tasks with minimal labeled data.

SSL in EO and remote sensing. Masking Auto-Encoder (MAE) [1] and contrastive learning [2] to pretrain EO FMs using SSL aim at *reducing* the need for large labeled datasets and have led to a growing interest in FMs for EO [4], [5]. EO FMs are pretrained on satellite data to learn robust feature representations and, then, are fine-tuned on smaller labeled datasets for *various* downstream tasks. EO FMs use different architectures for the encoding (and the decoding) of the data and features, which leads to *different* performance results for the downstream tasks, e.g. for pixel-wise regression [3].

In this work, our main aim is to study the scaling-up of EO FMs. Here, scaling-up refers to: i) incorporating a larger pretraining dataset that includes both oceans and ice in addition to land, i.e. the MajorTOM dataset 23TB (Sentinel-2 L2A, Core-S2L2A) [15], ii) a larger model in terms of number of parameters, i.e. 200M, and iii) an architecture that is suitable for scaling-up in terms of performance, efficiency, speed, and computational complexity, being capable of handling high-resolution data and images that cover large areas. Regarding the latter, i.e. regarding the architecture we select to scale-up, we note that the Vision Transformer (ViT), because of *attention*, suffers from quadratic complexity with respect to the image token sequence length. On the contrary, the recent Mamba S6 State-Space Model (SSM) achieves linear complexity and scales more favorably to high-resolution data and images that cover *large* geographical areas [30]. In this way, *global* correlations and long-range dependencies within the EO image that covers a large area can be effectively and efficiently modelled. Moreover, U-Net Convolutional Neural Network (CNN) models also have *linear* complexity, but can not model global correlations and long-range dependencies.

We use and train our recently proposed model PhilEO [3]. MajorTOM covers the vast majority of the Earth's surface. In this paper, we also create and use the specialized subset FastTOM 2TB that does *not* include oceans and ice. FastTOM is a task-specific dataset for land only. In the literature, there exist FMs *only* for oceans or ice, for example HydroFM [25], but we are interested in *Earth* Observation and not only in Land Observation or Ocean Observation (or Ice Observation). To this end, in this work, we develop and study various models with different numbers of parameters, and also with different architectures. We evaluate and examine the FLoating-point OPerations (FLOPs) that the models need [65]. We fine-tune the models on the PhilEO Bench [3] for road density estimation, building density pixel-wise regression, and land cover semantic segmentation, and we evaluate their performance.

In light of all the above, this paper's main contributions are:

- We train our model on the pretraining dataset MajorTOM 23TB which includes all regions, and the performance is competitive versus models pretrained on more specialized datasets (i.e. substantially smaller datasets that only include land). The additional data of oceans and

N. Dionelis, P. Naylor, V. Marsocci, and N. Longépé are with the European Space Agency (ESA), Φ-lab. E-mail: Nikolaos.Dionelis@esa.int.
R. Musto, G. Paoletti, and S. Sarti are with Leonardo Labs, Italy.
J. Bosmans is with VITO and was with ESA, Φ-lab.
S. Lefèvre is with IRISA, Université Bretagne Sud.
B. Le Saux is with the European Commission (EC).

ice on average do not decrease the performance on land-focused downstream tasks. These results indicate that large GFMs trained on global datasets for a wider variety of downstream tasks can be useful for downstream applications that only require a subset of the information included in their training.

- The second contribution is the exploration of U-Net, ViT, and Mamba as FMs for EO. U-Net captures local correlations amongst pixels, while ViT and Mamba capture local and distant correlations amongst patches (or pixels).

Further contributions of this paper. In addition, in this work, because scaling-up also refers to including additional EO modalities, for example S-1 SAR in addition to S-2, we release and make publicly available the new S12-PhilEOBench dataset for fine-tuning and downstream tasks. S12-PhilEOBench is for evaluating EO Foundation Models that are multi-modal and use co-located S-2 and S-1 data.

Organization of this paper. The rest of the paper is organized as follows. Section II provides background on the PhilEO Geospatial Foundation Model (GFM) and the PhilEO Bench evaluation, as well as the datasets MajorTOM 23TB and FastTOM 2TB. Section III describes the proposed model[1]. Section IV presents the related work. We evaluate the models on the PhilEO Bench evaluation framework in Sec. V for road and building density estimation, and land cover mapping. We present the results of the PhilEO MajorTOM 23TB and FastTOM 2TB models, as well as the results of the PhilEO Geo-Aware U-Net, the PhilEO ViT UPerNet 100M MajorTOM and FastTOM, and the PhilEO 200M FastTOM.

II. Background

PhilEO GFM. The PhilEO model [3] employs a combination of masked reconstruction and geo-location estimation as pretext tasks to train on the PhilEO Globe 0.5TB S2L2A dataset, which contains only land, *excluding* oceans and ice. The architecture is a modified U-Net CNN model. PhilEO Globe also includes four time steps: 4 seasons from the same year. Regarding the four time steps in PhilEO Globe 0.5TB, we note that in this work, scaling-up also refers to including the time aspect in the pretraining, for example *seasonality* and the four seasons of summer, winter, autumn, and fall.

PhilEO Bench. To enable *fair* evaluation and comparison of GFMs, [3] introduced the PhilEO Bench, which includes a global labeled Sentinel-2 dataset for the standardized downstream tasks: road density estimation, building density regression, and land cover mapping. The bench also incorporates several existing GFMs, such as [6] and [7], allowing for consistent comparisons across models.

MajorTOM and FastTOM datasets. The MajorTOM dataset [15] comprises in total approximately 60TB of unlabeled global data, covering most of the Earth, including oceans and ice. In this work, we focus on the MajorTOM Core-S2L2A subset, 23TB. In addition, FastTOM is a 2TB specialized subset of MajorTOM Core-S2L2A[2], containing only land and excluding oceans and ice, making it more task-specific for terrestrial downstream tasks.

III. Proposed Model

A. PhilEO MajorTOM 23TB and PhilEO FastTOM 2TB

We extend our previously proposed PhilEO model by pretraining it on the MajorTOM Core-S2L2A dataset [15], which contains 23TB data covering the vast majority of the Earth's surface, including land, oceans, and ice. To the best of the authors' knowledge, this is the first instance where the full 23TB MajorTOM dataset is used to pretrain a GFM.

The PhilEO MajorTOM model was trained using the Leonardo Davinci-1 Supercomputer, using between four and eight compute nodes, each equipped with four NVIDIA A100 GPUs (40GB VRAM), resulting in a total of 16-32 GPUs. This HPC configuration provided approximately 2.5–5 petaFLOPS of compute performance, which is important. Without such infrastructure, training on a single GPU would have taken several months of continuous computation, thus highlighting the critical role of HPC resources in enabling this work.

To handle the scale of the dataset and model, we used PyTorch's Distributed Data Parallel (DDP) to accelerate training. For larger ViT-based models still ongoing training on MajorTOM, we are transitioning to Fully Sharded Data Parallel (FSDP) to improve memory efficiency and support even larger model scales. In addition, mixed-precision training (BF16/BF32) was used for the training, for improved throughput and reduced memory usage.

Alongside the MajorTOM pretraining, we also trained a model variant on the FastTOM 2TB subset, which contains only land. Here, we note that FastTOM serves as a *proxy* for MajorTOM: strong performance on FastTOM is expected to translate into good performance when *scaling-up* to the full MajorTOM 23TB dataset. This strategy allows for efficient experimentation and architecture selection before undertaking the expensive full-scale pretraining.

For the pretraining, the models we develop and train have three output heads [3], one for reconstruction with the Mean Squared Error (MSE) loss where we mask a percentage of the input image during training, one for estimating the geo-location longitude and latitude where cosine and sine values are used, and one for predicting the Köppen-Geiger climate zone classes [3], [67]. "Geo-Aware" (for example, in the PhilEO Geo-Aware U-Net model) refers to these *three* output heads, i.e. to the reconstruction objective, the geo-location estimation[3], and the Köppen-Geiger climate zone prediction.

In this subsection, we note that our main contribution is pretraining on the MajorTOM and FastTOM[4] datasets. We scale the pretraining from 0.5TB to 23TB, as well as to 2TB, where the 23TB are general-purpose EO data. We use different number of model parameters and architectures. We show that for road density regression, for the n-shot $n = 500$, larger pretraining data reduces the final downstream task's

[1]We release the code of our model in: http://github.com/ESA-PhiLab/PhilEO-MajorTOM.

[2]http://github.com/ESA-PhiLab/Major-TOM and also in http://huggingface.co/Major-TOM

[3]http://huggingface.co/datasets/NikolaosDionelis2023/data_statics/tree/main

[4]http://huggingface.co/datasets/NikolaosDionelis2023/Subset_FastTOM/tree/main

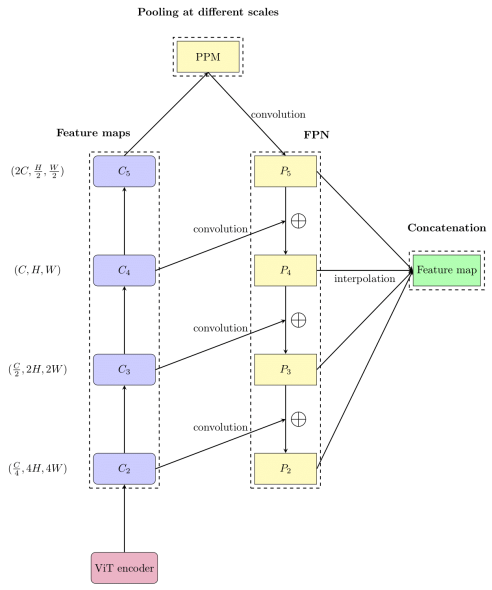

Fig. 1. High-level overview of ViT UPerNet. The dimensions of the feature map C_4 are maintained. The other feature maps are upsampled, respectively downsampled. The resulting feature map is used for downstream processing.

Root Mean Squared Error (RMSE), resulting in a percentage improvement 12.0%. In addition to pretraining dataset scaling, we also study the scaling-up of the model parameters and the architecture, comparing CNN- and ViT-based models, as well as Mamba. We evaluate the effectiveness of the scaling-up on the PhilEO Bench, covering both road and building density estimation, as well as land cover mapping.

We also develop and compare different architectures for the EO Foundation Models in order to select which *one* to scale-up. More specifically, for road density regression, we show that the ViT UPerNet model outperforms the ViT with a convolutional decoder in RMSE, with a percentage improvement of 34.2% for the n-shot $n = 100$. On the same task, the U-Net CNN Geo-Aware model achieves the same performance as the ViT UPerNet.

B. PhilEO ViT UPerNet

PhilEO Bench [3] standardizes model comparison by using a common convolutional decoder based on a U-Net-like architecture [8]. In this work, we introduce an alternative decoder strategy by implementing the UPerNet decoder [10], shown in Fig. 1, in the PhilEO Bench, in order to compare models with a ViT backbone. The UPerNet design [10] draws inspiration from human visual perception, combining *hierarchical* feature extraction through a Feature Pyramid Network (FPN) [21] with global context aggregation via a Pyramid Pooling Module (PPM). The FPN aggregates features across *multiple* scales through a top-down pathway with lateral connections, while the PPM captures context at various spatial resolutions.

ViT UPerNet architectures have been shown to achieve state-of-the-art performance in various computer vision tasks [11]-[14], including segmentation and classification. For models that use a ViT backbone, we extract intermediate feature maps, $\{C_2, C_3, C_4, C_5\}$ in Fig. 1, across different layers. These feature maps are then appropriately resized (i.e. upsampled or downsampled) to align spatial dimensions.

The highest-level feature map C_5 is processed by the PPM, which applies pooling operations at *multiple* scales to capture contextual information at different resolutions. The outputs of the PPM and the FPN are then *fused* through a series of 1×1 convolutions and element-wise additions, producing the final set of *multi-scale* feature maps, $\{P_2, P_3, P_4, P_5\}$ in Fig. 1. These aggregated features are then used for downstream pixel-wise prediction tasks, including regression and segmentation. Finally, in this subsection, we note that our main contributions are: i) including the ViT UPerNet in the PhilEO Bench (i.e. see Sec. V-A, as well as Figs. 4-6), and ii) studying the scaling-up of the ViT UPerNet model (see Sec. V-E, as well as Figs. 15-17, and also Sec. V-C, as well as Figs. 7-11).

C. Mamba 2D Models

In this paper, our main aim is to study scaling-up and we include in our analysis Mamba S6 2D SSM models [36]. We develop, train, and evaluate models based on Mamba and we include this type of models (i.e. Mamba) in the PhilEO Bench, and this is important. Mamba models are able to handle long sequences and have demonstrated strong performance in tasks that require *long*-range correlations and dependencies [38]. We note that this type of models has *not* been tested before in the evaluation framework PhilEO Bench, and also not in other evaluation testbeds such as Geo-Bench, Copernicus-Bench, FoMo-Bench [49], PANGAEA, EarthNets, and SustainBench.

We develop, train, and evaluate Mamba models that use input-dependent matrices (i.e. the S6 model compared to, for example, the S4 SSM) [33]. The Mamba models we train and test are for images, i.e. 2D, so that there are limited spatial discrepancies introduced due to the Mamba model being sensitive to the image token order. More specifically, we do not use 1D Mamba, which is for 1D *causal* signals, because this would introduce spatial discrepancies between neighbouring pixels in an image based on the scan traversal strategy that is used (i.e. based on how the image is *unrolled*), and this is important. To use 1D Mamba models in images, a reverse scan can be used [30], [55], that is, because we focus on spatial dependencies and correlation (and not, for example, on temporal correlation), the Mamba models for images, i.e. 2D, are bidirectional. Notably, regarding an example of *not* a 2D Mamba model, 1D Mamba would not have spatial continuity (and also not *as strong* correlations and dependencies) for the top and left adjacent pixels (i.e. and also for the *entire* 8-pixel neighbourhood of each pixel) in an EO image [37], [36].

For the Mamba 2D models, which we will also examine in Sec. V, they are based on continuous-time dynamics [30], [33]. We denote the state by h, and for the linear dynamics (i.e. similar to the Kalman filter), we have:

$$h'(t) = \mathbf{A}h(t) + \mathbf{B}x(t) \tag{1}$$

$$y(t) = \mathbf{C}h(t) + \mathbf{Z}x(t) \tag{2}$$

where the input signal is denoted by x and the output by y. The matrices $\mathbf{A}$, $\mathbf{C}$, $\mathbf{B}$, and $\mathbf{Z}$ are the *state* transition matrix,

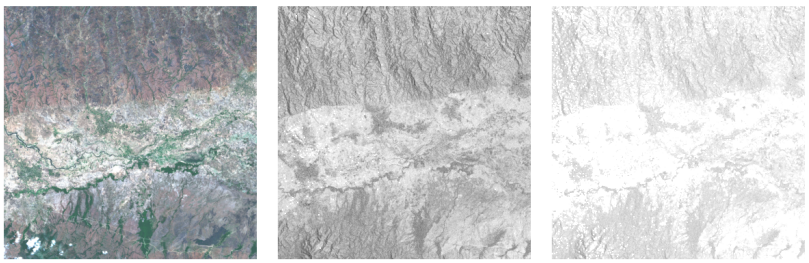

Fig. 2. Example images from S12-PhilEOBench from East Africa where we have, from the left, S-2, co-located S-1 SAR GRD VV, and S-1 GRD VH.

Fig. 3. Further example images from S12-PhilEOBench where we have, starting from the left, S-2 multi-spectral and co-located S-1 SAR GRD VV and VH.

the output matrix, the input matrix, and the input affecting the output matrix, respectively [54]. Here, we note that the input affecting the output matrix, $\mathbf{Z}$, is the residual connections in the network. The state, h, is of dimension N, i.e. $h \in \mathbb{R}^N$. In addition, $x \in \mathbb{R}$ and $y \in \mathbb{R}$. Moreover, for the matrices, we also have: $\mathbf{A} \in \mathbb{R}^{N \times N}$, $\mathbf{C} \in \mathbb{R}^{1 \times N}$, $\mathbf{B} \in \mathbb{R}^{N \times 1}$, and $\mathbf{Z} \in \mathbb{R}$.

The next steps are discretization [31], [30], making the matrices input-dependent (i.e. S6, Mamba), and also ensuring spatial *continuity* for the 2D images (i.e. for EO data) rather than introducing spatial discrepancies due to the traversal and unrolling of the image to a vector [38]. For the discretization of the continuous-time dynamics in (1) and (2), we have:

$$h_k = \bar{\mathbf{A}} h_{k-1} + \bar{\mathbf{B}} x_k \tag{3}$$

$$y_k = \mathbf{C} h_k \tag{4}$$

where because of using residual connections, we omit the input affecting the output matrix. The main *advantage* of the Mamba S6 SSM model is the linear complexity with respect to the input sequence length (in contrast, for example, to the ViT which has quadratic complexity), and the models which we will also examine in Sec. V are for images, i.e. 2D [36]-[38].

Mamba models, instead of attention and the Query, Key, Value multi-head mechanism [9], use selectivity. For the linear dynamics in (3) and (4), the input-dependent matrices are:

$$\bar{\mathbf{A}} \; = \; \exp(\mathbf{S}_\Delta \mathbf{A}) \tag{5}$$

$$\bar{\mathbf{B}} \; = \; (\mathbf{S}_\Delta \mathbf{A})^{-1} \left(\exp(\mathbf{S}_\Delta \mathbf{A}) - \mathbf{I}\right) \mathbf{S}_\Delta \mathbf{S}_\mathbf{B} \tag{6}$$

where $\mathbf{S_B}$ is the output of the learnable linear projection for making the matrix $\mathbf{B}$ input-dependent, i.e. $\mathbf{S_B} = \mathbf{x}\mathbf{W}_\mathbf{B}^T$, where $\mathbf{x} \in \mathbb{R}^{B \times L \times D}$ is the input, B is the batch size, L is the image token sequence length, and D is the embedding dimension [30], [36]. In addition, the learnable linear projection is denoted by $\mathbf{W_B} \in \mathbb{R}^{N \times D}$. Here, $\mathbf{S_B} \in \mathbb{R}^{B \times L \times N}$. Similarly, for making the matrix $\mathbf{C}$ input-dependent, we have: $\mathbf{C} = \mathbf{x}\mathbf{W}_\mathbf{C}^T$, where we denote the learnable linear projection by $\mathbf{W_C} \in \mathbb{R}^{N \times D}$. Also, $\mathbf{C} \in \mathbb{R}^{B \times L \times N}$ [30], [31]. We note that for ViT, attention is also based on *learnable* linear projections for the Query, Key, Value matrices that are input-dependent.

Furthermore, in (5) and (6), Δ is the sampling interval of the continuous parameters and $\mathbf{S}_\Delta \in \mathbb{R}^{B \times L \times D}$. Δ is an interval, i.e. the step size for the continuous dynamics [33], [51]. We note that using Mamba S6 SSM models, we are able to use, leverage, *train*, and fit the learnable model parameter Δ which controls the sampling rate for the continuous parameters, and this is important. The parameter Δ of Mamba [52], [53] is like a *forgetting* mechanism that is able to control how much a specific sample is forgotten or remembered. In particular, if $\Delta \to \infty$, then the model focuses more on the *input*, while if $\Delta \to 0$, the focus is more on the previous state. For the input-dependent Δ and its learnable linear projection, we have:

$$\mathbf{S}_\Delta \; = \; f(\mathbf{x}\mathbf{W}_\Delta^T), \tag{7}$$

$$\text{where the function } f(.) \text{ is: } f(x) = \tau_\Delta \, \text{Broadcast}_D(x) \tag{8}$$

where $\mathbf{W}_\Delta \in \mathbb{R}^{1 \times D}$ is the learnable linear projection (i.e., fully-connected (FC) layer) for making the time step Δ input-dependent [30], [63]. In addition, $\mathbf{S}_\Delta$ is a function of the output of the learnable linear projection, $\mathbf{x}\mathbf{W}_\Delta^T$. Moreover, in (7) and (8), τ_Δ is the softplus function. The $\text{Broadcast}_D(.)$ operation broadcasts the result to all the D dimensions, i.e. from 1 to D, as $\mathbf{S}_\Delta \in \mathbb{R}^{B \times L \times D}$. Using the parameter Δ as well as the equations (1)-(8), Mamba SSM models also use the Zero-Order Hold (ZOH) condition so as to treat each discrete input sample as a constant value that is held for a specific time interval until the next data sample is introduced [30], [69].

In *contrast* to Mamba SSM models, instead of the selective scan mechanism, Transformers (including ViT) use attention.

TABLE I
COMPARISON OF THE PHILEO MAJORTOM MODEL TO OTHER GFMS WITH RESPECT TO THE FEATURES OF THE PRETRAINING DATASET.

MODEL	DATASET	DATA SIZE	DATA FEATURES
PRITHVI-EO-2.0	GLOBAL HLS	4-5TB	LAND [23]
HYDROFM	HYDRO	1TB	OCEAN [25]
SATMAE++	FMOW	3.5TB	LAND [26]
CROMA	SSL4EO-S12	1.5TB	LAND [27]
COPERNICUS FM	C.-PRETRAIN	10TB	LAND [28]
TERRAMIND [29]	TERRAMESH	14TB	LAND, OCEAN, ICE
PHILEO GLOBE [3]	PH. GLOBE	0.5TB	LAND
PHILEO MAJORTOM	MAJORTOM	23TB	LAND, OCEAN, ICE

HLS = Harmonised Landsat S-2; fMoW = Functional Map of the World. Here, the MajorTOM dataset includes oceans and ice, while FastTOM and PhilEO Globe do not.

For Query, Key, Value (Q, K, V) multi-head attention [30]:

$$\mathbf{S}_Q = \mathbf{x}\mathbf{W}_Q^T,\ \mathbf{S}_K = \mathbf{x}\mathbf{W}_K^T,\ \mathbf{S}_V = \mathbf{x}\mathbf{W}_V^T \tag{9}$$

where we observe that we have trainable linear projections (like, for example, in (5)-(8) for the input-dependent matrices $\mathbf{S_B}$, $\mathbf{S_C}$, and $\mathbf{S}_\Delta$). In (9), $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$ have dimensions $\mathbb{R}^{N\times D}$. Next, for ViT, the attention weights are computed; these are given by: $S = \text{Softmax}(\mathbf{S}_Q\mathbf{S}_K^T/\sqrt{d_k})\,\mathbf{S}_V$. Here, the scaled dot product uses the dimension of the Key vectors, d_k. Multi-head attention [9] involves also using concatenation of the m attention weights and a final learnable linear transformation to project the multi-head attention scores to the outputs.

ViT models have significantly less inductive bias compared to CNN U-Net models. This is because ViT attention correlates everything with everything else. Moreover, in self-attention, image token order does not matter [9], [30]. In addition, Mamba models have more image-inductive bias compared to ViT models, as in Mamba S6 SSM models, the image token order is important and matters. In addition, we also note that our main contribution in this subsection is including Mamba SSM 2D models in the PhilEO Bench (i.e. see Sec. V-D), examining their performance as they can capture global correlations and have linear (and not quadratic) complexity.

D. The S12-PhilEOBench evaluation framework

In addition, in this paper, scaling-up also refers to including additional modalities, for example Sentinel-1 (S-1) SAR data in addition to Sentinel-2 (S-2) multi-spectral images. To this end, in this work, we also extend the PhilEO Bench dataset [3] to include co-located S-1 SAR data in addition to S-2 multi-spectral images. In Figs. 2 and 3, we show example images[5] from the global dataset S12-PhilEOBench (for example, Fig. 2 is from East Africa). For the *same* S-2 and S-1 images, we have three types of labels: road density pixel-wise regression (continuous real-valued output), building density regression, and land cover pixel-wise classification (with 11 classes).

[5]The S-1 SAR PhilEO Bench data can be found in http://huggingface.co/datasets/NikolaosDionelis2023/s1-phileobench/tree/main. In addition, we note that the S-2 multi-spectral PhilEO Bench data can be found in http://huggingface.co/datasets/NikolaosDionelis2023/s2-phileobench/tree/main.

IV. RELATED WORK

Recent studies on GFMs, summarized in Table I, highlight the increasing potential of learning robust feature representations from large-scale EO data, later used for fine-tuning on various downstream tasks. EO FMs can be trained, for example, using MAE [1], [56], [57], contrastive learning [2], or Joint Embedding Predictive Architecture (JEPA) [39]. MAE refers to masking (e.g., a large percentage of the image such as 75%), using the reconstruction loss, and the MSE [23], [26]. Also, the ViT [9] (or a *variant* of the ViT like the Swin Transformer [12]) is usually used. Here, we also note that the Spectral Angle Mapper (SAM), which takes into account the spectral signatures of the observations[6], can also be used in addition to the MSE, or even on its own as the loss function that is minimized during training. In addition, contrastive learning refers to performing data augmentation and bringing *closer* together in the latent feature space the data-augmented samples, repelling the dissimilar samples.

Furthermore, JEPA [40] (or I-JEPA [41]) refers to performing reconstruction in the latent feature space, estimating and predicting in the feature space the representation of a *part* of the image based on another part of the image (e.g., similar to masking), and operating in the *latent* space rather than the pixel image space. The feature space is closer to encoding semantics and is more *robust* than pixels because, for example, in S-2, the image space may have clouds or missing data.

For a recent study on the scope of geo-spatial FMs, we refer the reader to [60]. However, most (but certainly not all) existing studies have been limited by relatively modest dataset sizes and have not explored training GFMs at the scale of *more* than 20TB. This motivates our work, where we develop and pretrain the PhilEO MajorTOM model on the full 23TB MajorTOM Core-S2L2A dataset. Furthermore, during the ESA-NASA Workshop on AI Foundation Models for Earth Observation [61] (in May 2025), it was highlighted that the performance of FMs is not uniform across tasks; moreover, in certain cases, conventional models surpass state-of-the-art EO FMs. This observation prompts a critical inquiry: should the field pursue a *single* universal model, develop application-specific models, or adopt a Mixture of Experts (MoE) strategy?

The recent introduction of Meta's improved and upgraded DINOv3 framework [62] provides an additional rationale for enlarging the dataset scale. Although the large-scale proprietary geo-spatial dataset SAT-493M—comprising 493 million images at 0.6m resolution—has not been publicly released, its storage footprint can be reasonably estimated to substantially exceed 20TB, and this is important. It has been demonstrated that the $7B$-parameter variant of the DINOv3 model, pretrained on the SAT-493M dataset, attains state-of-the-art performance across 12 of 15 benchmark tasks, including classification, segmentation, and horizontal object detection. Evaluated on the Geo-Bench suite, DINOv3 surpasses prior leading approaches such as Prithvi-EO-2.0. Furthermore, under a linear probing paradigm—where the backbone (EO FM) is *frozen*—DINOv3 establishes superior results on three

[6]http://step.esa.int/main/wp-content/help/versions/9.0.0/snap-toolboxes/org.esa.s2tbx.s2tbx.spectral.angle.mapper.ui/sam/SAMProcessor.html

TABLE II
THE MAIN MODELS WE DEVELOP, IMPLEMENT, TRAIN, AND EVALUATE IN THIS WORK. WE PRESENT AND CATEGORIZE THE MODELS IN THIS TABLE. THE MODELS ARE LISTED IN ORDER OF APPEARANCE IN THE FIGURES.

NAME OF MODELS, INCLUDING SIZES	FIGURES IN	SUBSECTION(S) IN
VIT WITH CNN DECODER, 300M-0.5T	4-6	V-A
VIT WITH UPERNET DECODER, 300M-0.5T, 100M-2T, 100M-23T	4-11, 15-17	V-A-V-C, V-E
GEO-AWARE U-NET, 44M-0.5T, 200M-2T, 44M-23T, 44M-2T	4-11	V-A-V-C
RS3MAMBA 168M-2T	12-14	V-D
MAMBA UPERNET P16 100M [36]	12-14	V-D
TERRAMIND, V1.0-B, V1.0-L [29]	18-20	V-E
PRITHVI-EO-2.0 600M TL [23]	18-20	V-E

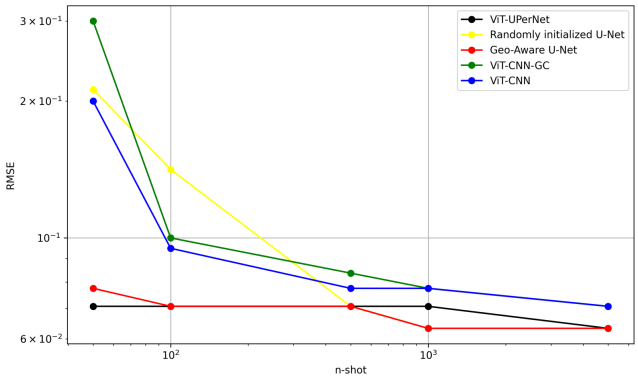

Fig. 4. Road density estimation: Evaluation in RMSE at different n-shots, where the best performing model is PhilEO Geo-Aware U-Net. The PhilEO ViT UPerNet black line is below the Geo-Aware U-Net red line. ViT UPerNet outperforms ViT CNN and ViT CNN Grouped Channels (GC) [3], [7].

previously unsaturated classification tasks and on *five* out of six segmentation tasks, further underscoring its robustness and generalization capacity. This indicates that, for specific important downstream applications, general-purpose self-supervised learning can serve as a competitive alternative to domain-specialized fine-tuning in the context of geo-spatial data.

The main shortcomings of existing methods, also in continuation of the discussion in Secs. II and I, as well as in continuation of Table I, is the quadratic complexity of ViT and attention with respect to the image token sequence length [34], [35], the potential performance benefits of the Mamba S6 2D SSM model (i.e. see Sec. V), seeing how much *faster* is for EO data Mamba compared to ViT, the effect of scaling-up the pretraining dataset size to 23TB (as well as to 2TB), and the impact of increasing the model size (i.e. number of parameters) on the model's final performance on the downstream tasks.

We note that because many EO FMs have been proposed recently, about the choice of models in the summary Table I, for the model proposed in [4], the code is *not* available. In addition, to this day, U-Net models are still considered the default benchmark for semantic segmentation, often outperforming new architectures. Moreover, [16] benchmarked the performance of Swin-UPerNet against a U-Net on semantic segmentation use cases. It was shown that the Swin-UPerNet model is a good competitor for the U-Net.

Furthermore, few-shot learning has gained attention within the EO community, as it enables models to generalize effectively from *limited* labeled data. Prior work [17], [18] demonstrated the advantages of few-shot learning for land cover classification, showing that models can maintain strong performance even with a small number of examples. The synergy between Foundation Models and few-shot learning was further explored in [19], showcasing improved segmentation performance through *combined* training strategies. Additionally, frameworks such as [20] have proposed systematic approaches to low- and few-shot adaptation in the context of Foundation Models.

V. EVALUATION AND RESULTS

The experiments in this paper are as follows. We have three datasets, 0.5TB, 2TB, and 23TB, and we also have three types of models, U-Net CNN, ViT UPerNet, and Mamba. In addition, we have three downstream tasks. We first present the 0.5TB experiments in Sec. V-A, i.e. Figs. 4-6. Next, we present the U-Net CNN and ViT experiments, i.e. in Figs. 7-11 (Sec. V-C). Here, the U-Net 23TB scaling-up experiments are presented. We then present the Mamba results (i.e. Figs. 12-14). These are in Sec. V-D. Subsequently, we present the ViT UPerNet 23TB scaling-up experiments, in Figs. 15-17 (i.e. Sec. V-E). Next, we present the comparison with TerraMind [29] and Prithvi-EO-2.0 [23]. This is in Figs. 18-20. We are doing these experiments in this order because in the beginning, for the 0.5TB results, we examine which architecture/ model to scale-up based on the results on the small dataset. Next, for the U-Net CNN versus ViT experiments, this is an interesting comparison because U-Net CNNs capture local correlations while ViTs capture *global*, including local, correlations. Then, for the ViT versus Mamba experiments, this is an interesting comparison because they both capture global correlations and Mamba is more computationally efficient. For the ViT 23TB scaling-up experiments, this is interesting because we increase the pretraining dataset size and study what happens when oceans and ice data are included in the pretraining. Also, for the comparison with TerraMind and Prithvi experiments, this is interesting because they are recent state-of-the-art models. The main findings (as we will also see in the sections below) are that the Geo-Aware U-Net 200M model is effective for pixel-wise regression tasks and outperforms the other models. In addition, the FLOPs analysis indicates that the smaller model U-Net 44M can be effective for pixel-wise regression downstream tasks. Furthermore, the comparison with TerraMind shows that for pixel-wise regression tasks, for some n-shot samples, the proposed models can outperform TerraMind.

In this paper, we have developed and implemented several models, including for example the PhilEO Geo-Aware U-Net 200M model, as well as the ViT UPerNet 100M-2T model, the RS3Mamba 168M-2T (e.g., in Fig. 14), and the patches-based Mamba UPerNet model which we will examine and evaluate in Sec. V-D. We list and summarize these main

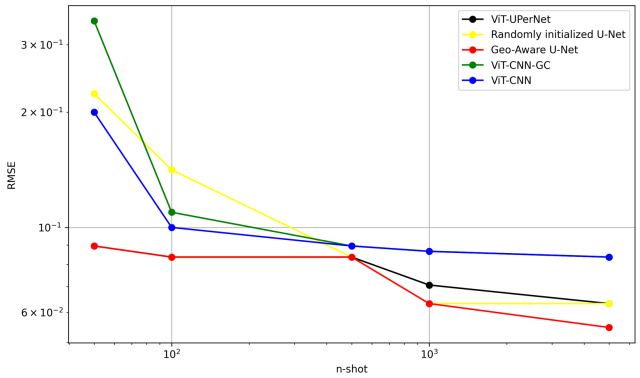

Fig. 5. Building density regression: Evaluation in RMSE at various n-shots. The PhilEO ViT UPerNet black line is below the Geo-Aware U-Net red line.

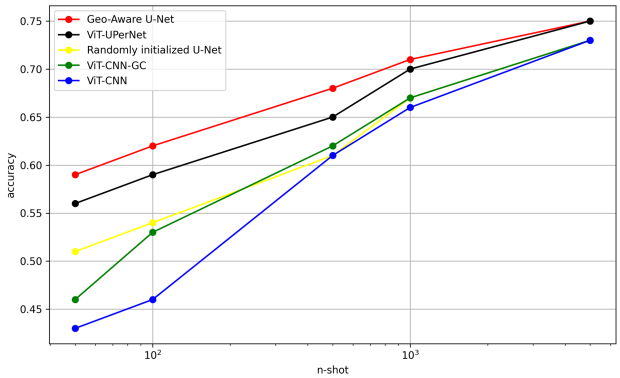

Fig. 6. Land cover mapping: n-shot evaluation of models in accuracy.

models of this paper in Table II, where we also summarize the main subsection in which the models are evaluated in. In this section, we evaluate the proposed models, comparing different overall model sizes and architectures (e.g., ViT vs. Mamba), analyzing the impact of scaling-up pretraining (i.e. Sec. III-A). The results are presented, examined, and discussed in the next subsections, i.e. in Secs. V-A-V-F.

Before training the models directly on MajorTOM, we first train them on the 0.5TB PhilEO Globe dataset [3] in order to select the *best* architecture to scale-up. Here, the main reason for this, is that it is expensive, requiring a lot of compute power, GPU machines, and GPU hours (i.e. real-world and pragmatic realistic scenario), as well as it is time-consuming, to train the models on the 23TB MajorTOM dataset. For these reasons, in the next subsection, we first present and examine the results of the models that we developed, implemented, and trained on the 0.5TB PhilEO Globe dataset.

A. PhilEO ViT UPerNet pretrained on PhilEO Globe 0.5TB

As shown in Fig. 4, the PhilEO ViT UPerNet model consistently outperforms the PhilEO ViT model with a convolutional decoder across all the n-shot settings for road density estimation. Notably, the PhilEO ViT UPerNet achieves comparable performance to the PhilEO Geo-Aware U-Net model [3], highlighting the effectiveness of the UPerNet decoder for pixel-wise regression tasks. We test the models in the figures using the n-shot evaluation protocol, where we note that the horizontal x-axis is the n-shot number of samples per country, i.e. 6 countries, 50 times 6 (i.e. 300) data samples for the first point in the horizontal axis.

In Fig. 5, the PhilEO ViT UPerNet demonstrates improved RMSE performance for building density estimation compared to the convolutional decoder ViT baseline. For instance, in Fig. 5, at $n = 50$, the PhilEO ViT UPerNet achieves a RMSE of 0.08944, compared to 0.2 for the convolutional decoder ViT model, corresponding to an improvement of 55.28%. At $n = 100$, the ViT UPerNet model further improves to a RMSE of 0.08367, versus 0.1 for the convolutional decoder ViT baseline, leading to a 16.33% relative gain.

In Fig. 6, the PhilEO ViT UPerNet also outperforms the ViT CNN model across all n-shot experiments for land cover mapping, in accuracy. Moreover, both the PhilEO ViT UPerNet and the Geo-Aware U-Net outperform a fully-supervised U-Net model (without pretraining), particularly in low-shot regimes. Overall, the PhilEO Geo-Aware U-Net achieves slightly better performance, on average, than the PhilEO ViT UPerNet across the three downstream tasks, as shown in Figs. 4-6.

B. Scaling-up: Evaluation of PhilEO MajorTOM 23TB GFM

In this section, we present the evaluation and results of the models when we *scale-up* FMs in terms of larger pretraining dataset (MajorTOM Core-S2L2A 23TB dataset), larger model, and architecture. According to the results, evaluation, and experiments, FMs pretrained on general-purpose data, when scaled-up appropriately, can eventually outperform models pretrained on specialized data. We are interested in Earth Observation, and *not only* in Land Observation or Ocean Observation (or Ice Observation). In the literature, there exist FMs only for oceans or ice, for example HydroFM [25]. When we pretrain on the entire MajorTOM 23TB dataset, we do not use any prior information, while when we pretrain on specialized datasets that have only land and exclude oceans, ice, and deserts, i.e. the FastTOM 2TB dataset, then we use prior information. The downstream tasks we are interested in are on land, i.e. the PhilEO Bench [3]: road and building density estimation, and land cover mapping[7]. We focus on land downstream tasks[8] and not on oceans or ice (or deserts), and this is why *specialized* datasets have prior knowledge.

For clarity purposes, we plot and examine the performance of the models and our evaluation results in separate figures for: i) the Geo-Aware U-Net models (see Sec. III-A) and the ViT models (i.e. Sec. III-B), and also for ii) the best ViT model and the Mamba models. We focus on comparing no more than 5 or 6 models per plot, for clearness and clarity. By

[7]The PhilEO Bench downstream tasks labeled dataset (0.4TB) was presented in [3], and we also note that a small and more manageable subset can be found in http://huggingface.co/datasets/JenteBosmans/tiny_phileo.

[8]http://phileo-bench.github.io/

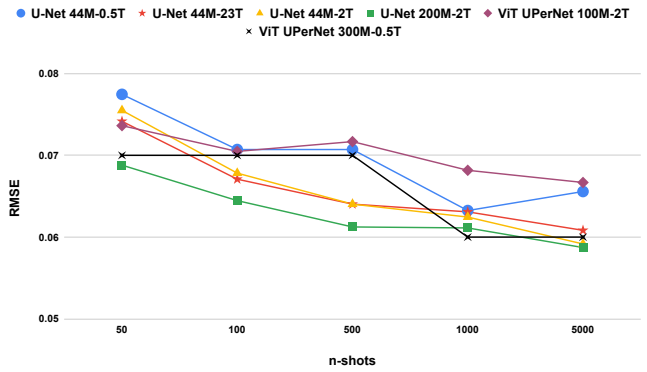

Fig. 7. Road density estimation in RMSE, comparing the models: PhilEO Globe 44M, PhilEO MajorTOM and FastTOM, PhilEO 200M FastTOM, ViT UPerNet 100M FastTOM and PhilEO Globe ViT UPerNet 300M. The models PhilEO MajorTOM and FastTOM outperform at most n-shots PhilEO Globe.

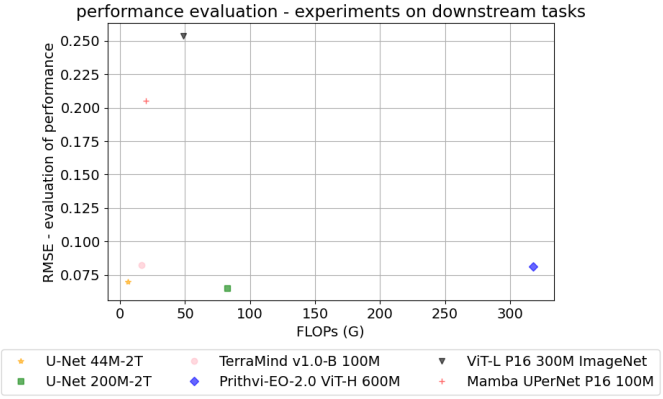

Fig. 8. Road and building density estimation: Evaluation in RMSE and also average over the different n-shots, examining the FLOPs needed by the models [65], where the best performing model is the PhilEO Geo-Aware U-Net. We also note that the two EO FM models TerraMind [29] and Prithvi-EO-2.0 [23] are also evaluated in Sec. V-E for roads, buildings, and land cover.

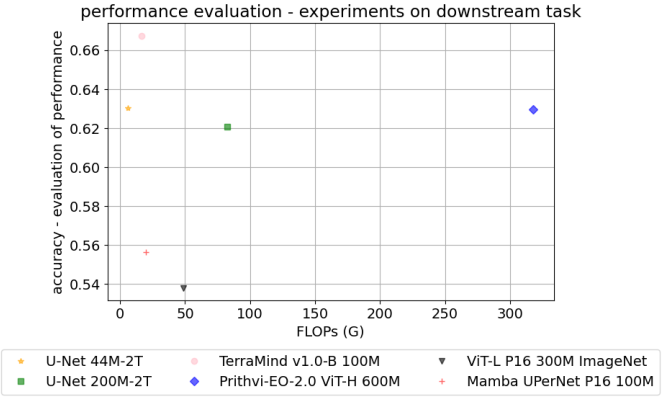

Fig. 9. Land cover mapping: Evaluation in accuracy and average over the n-shots, examining the FLOPs of the models [65], where the best is TerraMind.

separating the plots in this way, we compare CNN-based U-Net models with ViT, which is an interesting comparison, and we also *compare* ViT with recent 2D Mamba S6 SSM models (see Sec. III-C), which is also a very interesting comparison and engenders cogent discussions mainly around the trade-off between computational complexity (i.e. ViT, attention, and quadratic complexity vs. Mamba, SSM selective scan [32], and *linear* complexity) and performance (i.e. lower RMSE or higher accuracy) in the final downstream task, for example for global S-2 road density estimation pixel-wise regression. In addition, U-Net CNN models also achieve linear complexity.

Regarding (i) above, the comparison of the models that are based on Geo-Aware U-Net and those that use ViT is for example in Fig. 7. In the following subsections, we will present both (i) and (ii), for example the results in Figs. 7-11 for (i), for the Geo-Aware U-Net models and the ViT models. In addition, we will also present and examine the results for (ii) above, for the best ViT model and the Mamba models (i.e. Sec. III-C), for example the results in Fig. 13.

In addition to the performance of the models on downstream tasks (e.g., in terms of lower RMSE and higher accuracy, for example Figs. 7-11), we are also interested in how much time do the models need to train. This is important, as we are interested in *how much* compute we use (i.e. GPU power and hours), and for example, to address the problem of *quadratic* complexity of the ViT (i.e. self-attention), many different methods have been proposed in the recent literature [42], including for example FlashAttention [43], linear attention [45], [46], partial attention [42], Swin Transformer [12], [9], MetaFormer which uses spatial *pooling* operations instead of attention [44], Mamba [36], [38], the Fast Fourier Transform (FFT) spectral block [47], [50], [59], and using attention at later stages and not at the initial model blocks [36].

For the evaluation of the models, continuing the discussion in the preceding paragraph, we count the FLOPs that are needed by the different models [65]. The lower the FLOPs, the faster the model, i.e. the better (*less* operations). We measure the FLOPs in Python using the DeepSpeed library. Here, we also note that FLOPs are different from Floating-point operations Per Second (FPS). To be machine GPU agnostic, we use FLOPs and we do not measure time (i.e. seconds). In our FLOPs profiling analysis, we also include the models TerraMind [29] and Prithvi-EO-2.0 [23] (see Sec. V-E).

Sentinel-2, L2A data. For the models we develop, implement, train, and evaluate that are based on the EO FM PhilEO [3], we use images of size 128x128, and also 10 spectral bands (channels), i.e. multi-spectral satellite Sentinel-2 data. More specifically, as in [3], we use the 10 and 20 metre spatial resolution spectral bands from the satellite Sentinel-2[9].

Image patches, and working with patches rather than pixels. In this paper, we also develop, train, and evaluate a Mamba S6 2D SSM model that is based on image patches, learnable linear projection, tokenization, and positional embeddings like in ViT. We use patches of size 16x16 (and we also experiment with the patch sizes of 8x8 and 4x4).

C. Comparison of scaling-up for U-Net and ViT models

We study scaling-up and the behaviour and performance of the different models, and this is important. When we increase the

[9]http://sentiwiki.copernicus.eu/web/s2-mission#S2Mission-SpatialResolutionS2-Mission-Spatial-Resolutiontrue

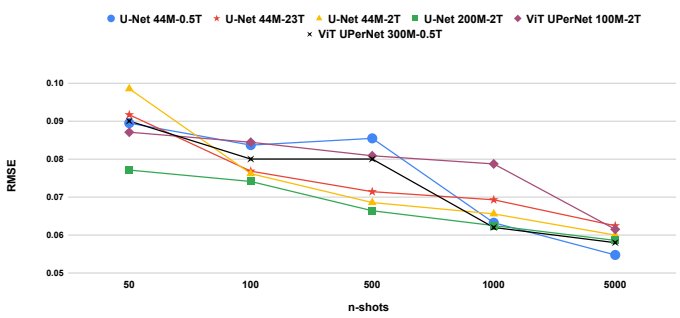

Fig. 10. Building density in RMSE for PhilEO Globe, PhilEO MajorTOM and FastTOM, PhilEO 200M FastTOM, ViT FastTOM and PhilEO Globe ViT.

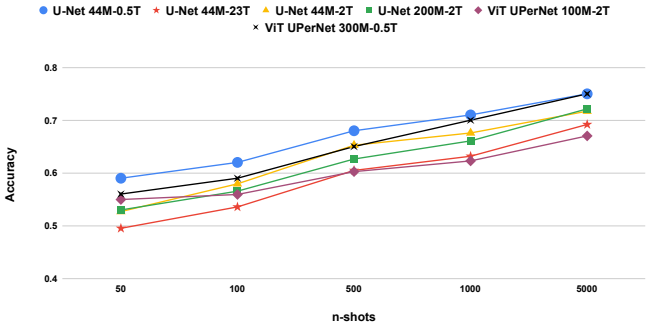

Fig. 11. Land cover in accuracy for the different models, at various n-shots.

size of the pretraining S-2 dataset to 23TB (i.e. Sec. III-A), we add ocean and ice data samples, and we effectively increase the aleatoric uncertainty, where here we note that aleatoric uncertainty is statistical and related to randomness, and the sample *not* being a typical example of the class and quantity we observe. Here, at this point, we also note that, in contrast to *aleatoric* uncertainty, epistemic uncertainty is systematic, caused by lack of knowledge (e.g., induced by the lack of detail in the measurements, for example by the lack of resolution or spectral information), and can be reduced by learning the specific characteristics of the quantity using additional information (for example, in-situ measurements).

Experiments and motivation. In this evaluation section, we present and examine the results of many models on the PhilEO Bench. The two recent models TerraMind [29] and Prithvi-EO-2.0 [23] are also evaluated on the PhilEO Bench in Sec. V-E. According to the results in Section V-A, we scale-up the best-performing architecture, the PhilEO Geo-Aware U-Net, and we pretrain it on the *full* 23TB MajorTOM dataset, as well as on the FastTOM 2TB specialized subset. We study the scaling-up to 23TB (and 2TB) data, as well as the number of parameters 200M (versus 44M), and the architecture—ViT UPerNet, i.e. see Sec. III-B— (vs. Geo-Aware U-Net [3]).

Models of size 44M and 200M, and different architectures. In this paper, in the beginning, we first develop and examine models using different architectures pretrained on the PhilEO Globe 0.5TB dataset. Next, we scale-up the best of the examined architectures by increasing the number of parameters and pretraining the models with bigger datasets, i.e. MajorTOM and FastTOM. We evaluate the performance of these models on the examined downstream tasks, and we find the best models. Also, we *compare* the models with the RS3Mamba model [48] which comprises a Mamba encoder, a CNN auxiliary encoder, and a custom multi-scale decoder. Our results demonstrate that for most n-shots for road density estimation and building density regression, Geo-Aware U-Net 200M-2TB outperforms all the other models we examine. We also show that for all n-shots for the task of road density regression, the PhilEO Geo-Aware U-Net 44M-23TB model outperforms Geo-Aware U-Net 44M-0.5TB. The effectiveness of both dataset and model scaling of the different EO FMs is validated using the PhilEO Bench [3]. We also study the impact of architecture scaling, transitioning from U-Net CNN to ViT, where the UPerNet decoder brings significant performance gains, as well as to 2D Mamba models [31].

Jointly increasing the pretraining dataset size and the model size. In this paper, we focus *not only* on scaling-up the volume of the pretraining data (i.e. to 23TB) [15], but also on scaling the number of parameters and the architecture, also studying the transition from CNN-based U-Net models to ViT backbones combined with the UPerNet decoder, as well as to Mamba for images. Here, the recent Mamba S6 SSM is a potential effective paradigm-shift solution [30], as it combines global dependencies modeling and linear computational scaling. The self-attention mechanism of the ViT is *not* sensitive to the token order, while on the contrary, the selectivity mechanism of Mamba (i.e. the selective scan) is sensitive to the token order. Notably, it needs a large hidden state dimension as it has an exponentially decaying recall accuracy based on the distance from the current token [36], [68]. In addition, ViT has quadratic complexity with respect to the image token sequence length, due to the attention Query, Key, Value mechanism, which is based on the scaled dot product similarity measure, while the 2D Mamba S6 SSM model is faster and more efficient as it achieves linear complexity with respect to the image token sequence length [30], [31].

Results for road density estimation. Fig. 7 shows the RMSE for road density estimation across different n-shot settings. For all n-shots, the scaled-up PhilEO MajorTOM 23TB model outperforms the Geo-Aware U-Net model pretrained on the PhilEO Globe 0.5TB dataset. At $n = 100$, the PhilEO MajorTOM 23TB model achieves a RMSE of 0.06724, compared to 0.07070 for the model pretrained on the PhilEO Globe 0.5TB dataset. This corresponds to an *improvement* of 4.89%. In the figure, we denote the Geo-Aware U-Net by U-Net for clarity purposes, where the Geo-Aware U-Net that we train and use is a modified U-Net pretrained on S-2 global data. For $n = 50$ and 100, the PhilEO 44M MajorTOM model outperforms the FastTOM and PhilEO Globe pretrained baselines, demonstrating the benefits of large-scale pretraining even when fine-tuning with *limited* labeled data.

Additionally, we develop and evaluate two *larger* models, as shown in Fig. 7:

- PhilEO 200M FastTOM: a 200M-parameter Geo-Aware U-Net variant pretrained on FastTOM,
- ViT 100M FastTOM: a Vision Transformer model (ap-

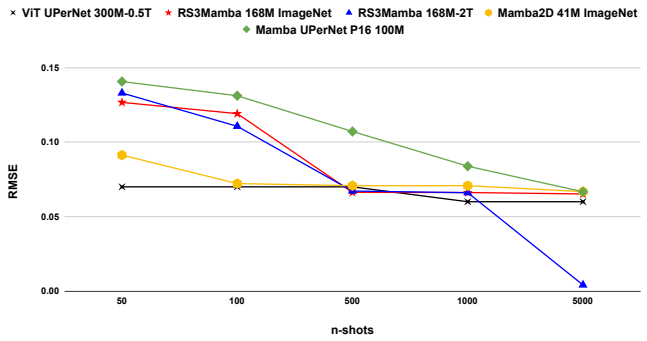

Fig. 12. Road density estimation in RMSE, comparing the models: the best ViT model with the Mamba models.

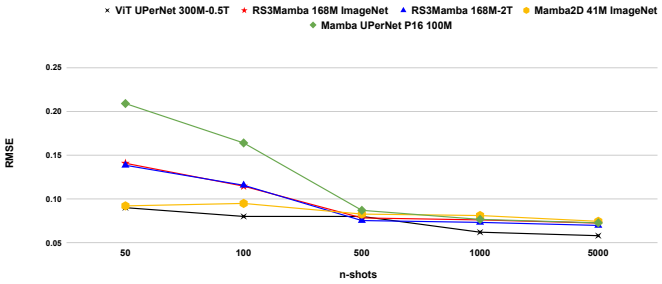

Fig. 13. Building density in RMSE for the best ViT and the Mamba models.

proximately 100M parameters, with depth 32, number of heads 16, and embedding dimension 512) pretrained on FastTOM with a UPerNet decoder utilizing the intermediate layers 5, 15, 23, and 31.

Both larger models show strong performance, with PhilEO 200M FastTOM outperforming the others at several n-shots. In particular, for n-shot 500, for roads in Fig. 7, the Geo-Aware U-Net 200M (i.e. U-Net 200M-2T) has better performance than U-Net 44M (U-Net 44M-2T), and the percentage improvement is 4.31%, i.e. 0.061 compared to 0.064. Here, at this point, we also note that because jointly increasing the pretraining dataset size and the model size (i.e. scaling-up *both* the volume of the pretraining data (23TB [15]) and also scaling the number of parameters) is important and also because pretraining on the entire MajorTOM dataset 23TB is expensive, we defer as future work the training of the Geo-Aware U-Net 500M 23TB model (i.e. it is ongoing work).

In Figs. 7-11, according to our FLOPs profiling analysis, the model Geo-Aware U-Net that is of size 44M parameters has 5.87G FLOPs. This is *low* computational complexity (for example, see TerraMind and Prithvi-EO-2.0 in Sec. V-E). In addition, the Geo-Aware U-Net 200M has 82.23G FLOPs. In Fig. 8, we compute, present, and examine the FLOPs needed by the different models, where the average over the two pixel-wise regression downstream tasks is calculated, as well as the mean over the different n-shots. The results show that, when taking into account the FLOPs (i.e. in Giga (G) FLOPs) that are required for the downstream tasks, the Geo-Aware U-Net model 200M (and 44M) have top/ very good performance.

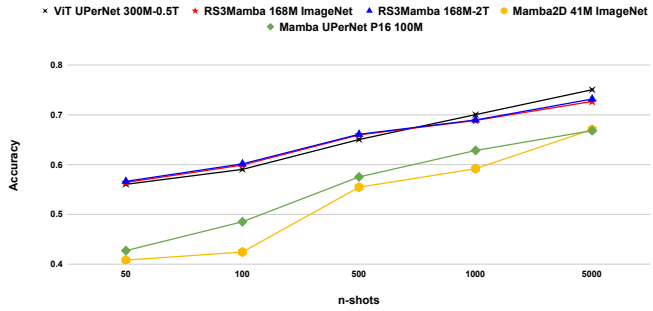

Fig. 14. Land cover mapping results in accuracy for the different models, i.e. for the best ViT model and the Mamba models, at various n-shots.

The model Geo-Aware U-Net 44M has a small number of FLOPs, approximately 6G (Fig. 8), and this is one of the key takeaways of this paper. We observe that the *small* model wins in this case. About scaling laws for the models, we are interested in the improvement in performance when we increase the number of FLOPs. For example, Geo-Aware U-Net 44M to 200M in Fig. 8: 5.87 to 82.23 G FLOPs, RMSE 0.0698 to 0.0653. A relevant question that is important is which is the best model for a given cost, and whether *saturation* is observed in performance accuracy when increasing the number of FLOPs. In Fig. 8, FLOPs are for the downstream tasks (i.e. for the forward operation of the network for a single image), and for Mamba, see Sec. V-D (as well as Sec. III-C).

We examine the performance vs. FLOPs plot for the models, where in this plot the vertical axis is the *performance* of the model (e.g., RMSE for the pixel-wise regression EO task in Fig. 8, or accuracy for the semantic segmentation classification task in Fig. 9), and the horizontal axis is the number of FLOPs needed by the model for the downstream task. In Fig. 8, the results are for road and building density pixel-wise regression, and we also examine the performance and FLOPs for land cover semantic segmentation in Fig. 9. For the latter, we observe that for this specific task, the best models are TerraMind (i.e. see Sec. V-E) and Geo-Aware U-Net 44M.

For reference, at this point, we also provide the FLOPs for the usually used ResNet models (i.e. these are common classification models and not EO FMs): ResNet-18 1.19G FLOPs, 12M parameters, ResNet-50 2.68G FLOPs, 26M model parameters, and ResNet-152 7.54G FLOPs, 60M parameters. We also note that the analysis of the computational complexity using FLOPs in Figs. 8 and 9 also indicates that sometimes smaller models can be better than larger models, i.e. a claim that, for example, has also recently been made in [64].

Performance of ViT vs. U-Net CNN. Taking into account all the three downstream tasks, i.e. Figs. 7-11, for n-shots larger than 1000, i.e. for a large n-shot samples number, ViT UPerNet is better than the U-Net CNN model, and this is important. This is mainly because of the performance of the ViT on the semantic segmentation land cover mapping downstream task (i.e. the model ViT UPerNet 300M-0.5T, black colour in the figure vs. the model U-Net 200M-2T, green colour), as well as because of the performance of the ViT on the pixel-wise regression road and building density estimation.

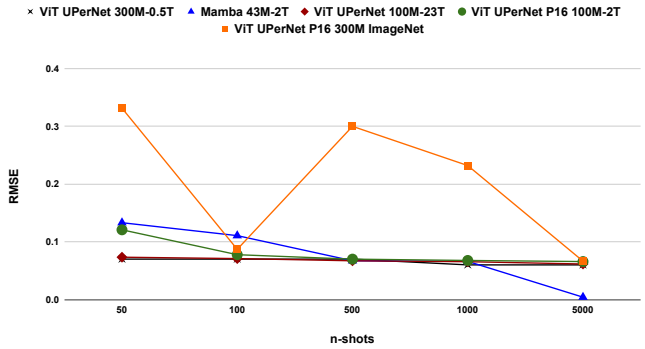

Fig. 15. Road density estimation in RMSE stydying the scaling-up of the ViT 100M model to the 23TB pretraining dataset MajorTOM, and also comparing the models: ViT and Mamba.

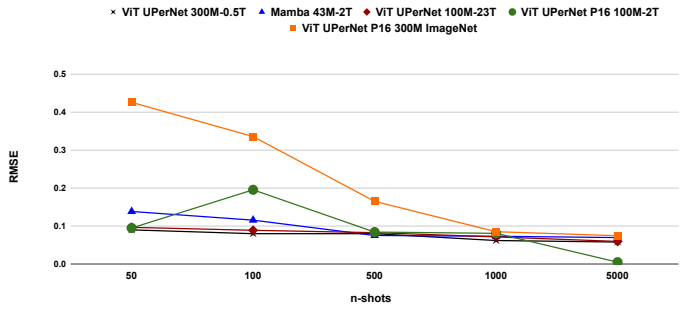

Fig. 16. Building density in RMSE for the scaled-up ViT 100M model to the 23TB pretraining dataset MajorTOM. Comparison of the models ViT, Mamba.

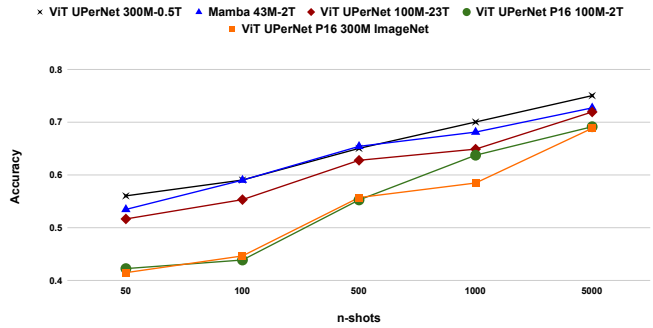

Fig. 17. Land cover in accuracy for the scaled-up ViT 100M model to the 23TB pretraining dataset MajorTOM, at various n-shots, as well as for the different models ViT and Mamba. Also, comparison with patch size 16 for the ViT, rather than 4.

The effect of increasing the model size. In Figs. 7 and 10, we also examine the results when increasing the model size together with increasing the size of the pretraining dataset. This is for the 200M parameters model, the Geo-Aware U-Net 200M-2T model in the figures. EO FMs pretrained on general-purpose data, when scaled appropriately (i.e. *joint* increase of pretraining dataset size and model size), can eventually outperform models pretrained on specialized EO data.

Building density regression. As shown in Fig. 10, similar trends are observed for building density estimation. In particular, at $n = 100$ and 500, both the PhilEO 44M MajorTOM and FastTOM models outperform in RMSE the PhilEO Globe 44M baseline. In addition, the PhilEO 200M FastTOM model again performs best across most n-shot settings.

Land cover mapping. Fig. 11 presents accuracy results for land cover semantic segmentation. Across all n-shot settings, the PhilEO Geo-Aware U-Net, pretrained on the smaller PhilEO Globe 0.5TB dataset, consistently outperforms models pretrained on the PhilEO MajorTOM and FastTOM datasets. A likely explanation for the superiority of PhilEO Globe in land cover is that it includes seasonal variations, i.e. 4 seasons, 3 months separation over 1 year, 4 time steps. For land cover, *seasonality* is beneficial, i.e. seasonality is more correlated with land cover than with roads or buildings: PhilEO Globe includes 4 time steps (i.e. 1 year), rather than MajorTOM, one *random* time step [15]. Also, it is important to note that PhilEO Globe and FastTOM contain only land, while MajorTOM includes a *broader* variety of scenes, such as land, oceans, ice, forests, and deserts. Since our downstream tasks focus solely on land, the models pretrained on PhilEO Globe and FastTOM benefit from prior knowledge that *aligns* with the evaluation setting, while MajorTOM-pretrained models do not.

Evaluation results, and land cover being correlated with seasonality. The dataset PhilEO Globe 0.5TB is multi-temporal and includes four time steps corresponding to the four seasons of the year. In Fig. 11, we observe much more correlation between land cover and including the four seasons of the year in the training set, compared to between road density and including the four seasons in the training dataset, as well as between building density and including seasonality in the pretraining dataset, in Figs. 7 and Fig. 10 respectively.

The main question of which is the best model. In this work, we are also interested in which is the best model. This depends on the task (as well as on the n-shot samples number—e.g., see Fig. 7—), and more specifically on whether we focus on pixel-wise regression or semantic segmentation. For the *former*, the best model for road and building density estimation is the model Geo-Aware U-Net 200M (i.e. Figs. 7 and 10, as well as Fig. 8). In addition, for the latter, for the task of land cover mapping, TerraMind is the best model (see Sec. V-E, as well as Fig. 9), and the main reason for this is because in its modalities in the pretraining (i.e. in its *correlation learning* process), it includes land cover classes, geo-location, DEM, S-1 SAR, and NDVI data in addition to S-2 multi-spectral images. Here, the next best model for land cover classification is the model Geo-Aware U-Net 44M.

To better understand the impact of general-purpose versus specialized EO pretraining, we compare the performance of models trained on specialized land-only datasets (i.e. PhilEO Globe 0.5TB and FastTOM 2TB) with the performance of models trained on a *large* general-purpose dataset (MajorTOM 23TB) which includes oceans and ice. Given that all models share the same Geo-Aware U-Net architecture (approximately 44M parameters) and that the smaller datasets use a land mask to exclude oceans and ice, we expect models trained on PhilEO Globe and FastTOM to perform *better* in downstream land-focused tasks. Indeed, in most cases, this expectation holds. However, for the road density estimation task under

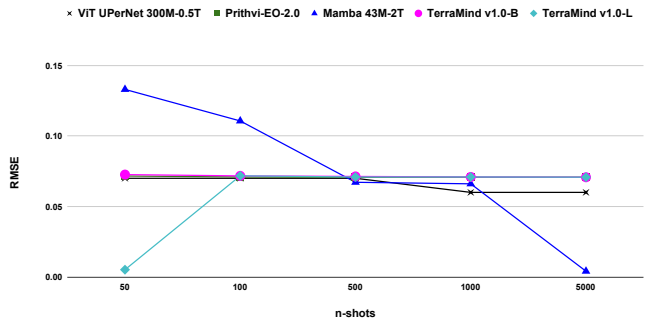

Fig. 18. Road density pixel-wise regression in RMSE for the models ViT and Mamba. Comparison to the TerraMind [29] and Prithvi-EO-2.0 [23] models. In the plot, the green line (Prithvi-EO-2.0) is below the pink line (TerraMind).

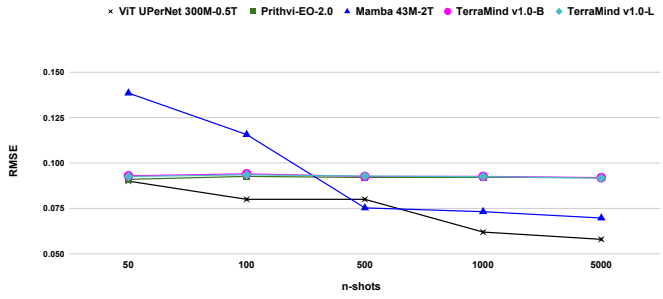

Fig. 19. Building density estimation in RMSE for the models ViT and Mamba, and comparison to the TerraMind 100M and 300M and Prithvi-EO-2.0 600M TL models. The pink TerraMind 100M line is below cyan TerraMind 300M.

low-shot (few-shot) learning settings, the model pretrained on MajorTOM surprisingly outperforms the specialized models in terms of RMSE. This suggests that despite the *broader* and potentially noisier pretraining data, large-scale general pretraining can be advantageous under certain conditions.

This study also allows us to analyze two key factors:

- Specialization effect: Does pretraining on land-only images give an advantage when the downstream tasks are also land-focused?
- Scaling effect: How does increasing the pretraining dataset size from 0.5TB and 2TB (specialized) to 23TB (general-purpose) impact downstream performance?

We study how GFMs scale and we explore how the PhilEO models behave, i.e. their scaling behavior. Pretraining on the MajorTOM dataset represents a truly general-purpose approach, and this is important, aligning with the concept of a GFM for Earth observation, rather than a Land-only model (or an Ocean-only model). GFMs pretrained on general-purpose satellite data, and not only on land, are Earth Observation models and not only Land Observation models (or Ocean, Ice, or Desert Observation models). In contrast, models like SeCo [24] focus on *land* observations only, while others, such as HydroFM [25] (i.e. Table I), specialize only in oceans. Our evaluation and results in this section show that although specialized pretraining offers an initial advantage, scaling-up the pretraining on diverse general-purpose datasets, when scaling-up is performed appropriately, can eventually lead to

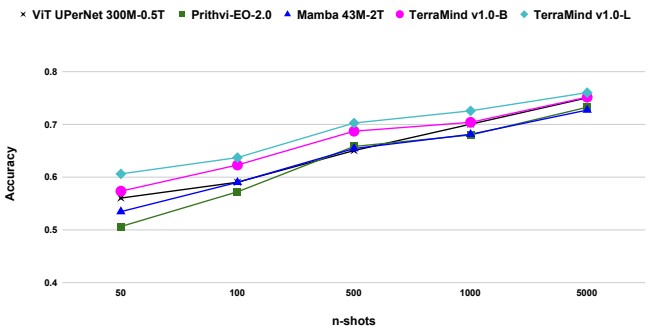

Fig. 20. Land cover in accuracy for the models ViT and Mamba, and comparison to the models TerraMind and Prithvi-EO-2.0, at various n-shots.

GFMs that outperform *specialized* models, even in tasks where prior domain-specific knowledge initially seemed crucial.

D. Comparison of scaling-up for Mamba and ViT models

In this section, we focus on the evaluation and results of the Mamba models (i.e. Sec. III-C). We examine and compare, also in continuation of the discussion in Sec. V-B, the best ViT model and the Mamba 2D S6 SSM models. We note that in the figures, for consistency, we present the results for the three PhilEO Bench fine-tuning downstream tasks, which are global S-2, in the order: road density pixel-wise regression, building density estimation, and land cover pixel-wise classification. Moreover, for clarity and comprehension, for the same model, we use the same identical symbol (e.g., star) and *colour* (for example, dark orange for the model U-Net 44M-23TB).

In Fig. 12, as well as in Figs. 13 and 14, we also evaluate the model Mamba2D [36]. We observe that Mamba2D achieves good performance (*low* RMSE) for road and building density estimation in Figs. 12 and 13, i.e. light orange line (circle symbol). We train, use, and test the Mamba-only variant of Mamba2D, i.e. *not* a hybrid model, not using attention at the end stages [36], and using only "2D_local" at the four blocks. The model Mamba2D has 41M parameters and 4.62G FLOPs, and it is pretrained on ImageNet (130GB, *labeled* dataset). For reference, here, we also note that the model 2DMamba [37] has 10M parameters and 18.2G FLOPs when the embedding dimension is 512 and the 2D Mamba SSM blocks are 4.

In Fig. 12, we examine the results of the best ViT model and the Mamba models for road density estimation. We observe that for low n-shots, the ViT model achieves a *lower* RMSE score compared to the Mamba 2D S6 SSM models. We train, use, and evaluate the model RS3Mamba [48] pretrained on ImageNet which has approximately 168M parameters and 18.64G FLOPs, and we note that a smaller version of RS3Mamba is also possible, i.e. 52M model parameters and 6.71G FLOPs. For the *larger* model, we use ResNet-50 [48], while for the smaller version, it is based on ResNet-18.

In Figs. 12-14, we evaluate and compare with the recent model RS3Mamba [48]. For *building* density pixel-wise regression in Fig. 13, we present and examine the RMSE results of the best ViT model and the Mamba models. For low n-shots, we observe that the ViT model achieves a lower RMSE score

compared to the Mamba models. In addition, for the task of land cover mapping in Fig. 14, for low-shot samples, the ViT and the pretrained Mamba 168M on S-2 multi-spectral data have approximately comparable performance in accuracy.

For the Mamba S6 models in Figs. 12-14, we note that the size of ImageNet is 130GB and it is *labeled*, i.e. supervised learning is performed (rather than self-supervised learning).

Mamba using patches and UPerNet. In Figs. 12-14, we also develop, train, and evaluate the model Mamba UPerNet P16 100M which uses patches of size 16 (i.e. P16)—relatively large patches— and the decoder UPerNet (see Fig. 1, as well as Sec. III-B). It has 100M parameters and 12.68G FLOPs, and this is relatively low. The Mamba 2D block, which is used instead of *attention*, is M2DBlock (Mamba 2D block) [36]. According to the results in Figs. 12 and 13, for pixel-wise regression tasks, for a large number of labeled data samples, i.e. for n-shot 5000, the model achieves good performance.

Finally, because Mamba has benefits in efficiency, in addition to Figs. 8 and 9, we further examine the number of FLOPs of the Mamba SSM models. The Mamba UPerNet P16 model (i.e. from the preceding paragraph) has 100M parameters and 12.68G FLOPs. RS3Mamba has 168M parameters and 18.64G FLOPs. These two are significantly lower FLOPs than ViT UPerNet 100M (i.e. see Sec. V-E, as well as Figs. 15-17) which has 388.51G FLOPs, i.e. approximately 30 times lower and this is because of the patch size 16 (instead of 4, i.e. larger patch sizes) and also because of Mamba selective scan linear complexity rather than ViT self-attention quadratic complexity.

E. ViT scaling-up to 23TB, and TerraMind and Prithvi-EO-2.0

In this subsection, we will first present and examine the ViT scaling-up to the MajorTOM dataset, 23TB. Next, we will evaluate, compare with, and examine the use of larger patch sizes in the ViT and the *effect* of this in terms of performance. Further, in this subsection, we will also evaluate and compare with the models TerraMind [29] and Prithvi-EO-2.0 [23].

Evaluation of ViT scaling-up to MajorTOM, 23TB. In Figs. 15-17, we evaluate and examine the performance of the ViT *scaling-up* to the MajorTOM dataset. We also perform our FLOPs profiling analysis and the model ViT UPerNet 100M, which has patch size 4 (i.e. relatively *small* patch size) and is pretrained on S-2 (MajorTOM, 23TB), has 388.51G FLOPs.

The performance of ViT UPerNet 100M in Figs. 15 and 16 for road and building density estimation, respectively, is good. We evaluate ViT UPerNet 100M (i.e. see Sec. III-B, as well as Sec. III) in Figs. 15-17, and we note that scaling-up the pretraining dataset size to 23TB (MajorTOM Core-S2L2A) required significant HPC GPU model training resources.

Evaluation of using larger patch size, and efficiency-performance trade-off. In Fig. 17, as well as in Figs. 15 and 16, we also evaluate, plot, and examine the performance of the model when we use patches of *larger* size, i.e. of more pixels. Using different patch sizes, we study the trade-off between efficiency (i.e. larger patches, more coarse-grained and faster) and performance. We examine ViT UPerNet P16 100M-2T and ViT UPerNet P16 300M ImageNet. Regarding the latter, we examine the model size *Large* which has a bigger number of model parameters (300M), i.e. an interesting comparison. In addition to comparisons and discussion about the patch size, we also have comparisons and discussion about the pretraining dataset, i.e. EO-specific or *ImageNet*, as well as about the model size. The ViT UPerNet P16 100M model has 19.12G FLOPs, while ViT UPerNet P16 300M ImageNet has 48.48G FLOPs. We observe that pretraining on satellite S-2 data (i.e. rather than on natural images, ImageNet) is beneficial for low n-shot samples. For example, this is also true for patches of size 8, i.e. for ViT UPerNet P8 100M, where for land cover mapping, for the n-shot 500, the model pretrained on S-2 data has an accuracy of 0.5713, while the model pretrained on ImageNet 0.5343 (i.e. percentage improvement 6.92%).

Evaluation and comparison to TerraMind. We evaluate and compare with the recently proposed EO FM TerraMind [29]. TerraMind v1.0-B, i.e. the model of size Base, has approximately 100M parameters, and according to our FLOPs profiling analysis, it has 16.62G FLOPs. Also, TerraMind v1.0-L, i.e. the variant of size *Large*, has approximately 300M parameters; according to our analysis, it has 45.72G FLOPs.

We evaluate the recently proposed model TerraMind, performing fine-tuning, which *differs* from the freezing the backbone procedure that is used in [29]. We evaluate and compare with TerraMind, which in the pretraining dataset contains oceans (here, see Table I). In Fig. 20, for land cover mapping, the main reason that on this downstream task, the best performance is achieved by TerraMind, is that it includes in the pretraining many modalities, including Land Use Land Cover (LULC), geo-location, DEM, S-1, and NDVI (i.e. Figure 1 in [29]). For the model TerraMind, the LULC classes in the pretraining are *Built*, Trees, Grass, Crops, Bare ground, Water, Flooded vegetation, Shrub and Scrub, and Snow and Ice.

In Figs. 18-20, we evaluate and compare with TerraMind both 100M and 300M. Here, TerraMind v1.0-B [29], during pretraining for all the modalities, has approximately 300M parameters. Also, the *same* model, during fine-tuning for a single modality, has approximately 100M parameters. We use the framework TerraTorch and we perform fine-tuning. We note that both using TerraTorch, i.e. not only the encoder but also the suggested decoder network U-Net, and performing fine-tuning (and *not* freezing the backbone), as well as evaluating on the PhilEO Bench, differ from the evaluation in [29].

In Figs. 18 and 19, the performance of TerraMind is not constant, i.e. it varies with the number of n-shots and it looks this way due to the zoom-out. TerraMind has this behaviour for road and building density estimation in Figs. 18 and 19 likely because it includes in the pretraining LULC classification, in particular the class Built that includes roads and buildings.

Evaluation and comparison to Prithvi-EO-2.0. We now compare the performance of our models with the Prithvi-EO-2.0 model [23]. We also are interested in the computational complexity, in addition to the final performance evaluation results. According to our FLOPs profiling analysis, the model Prithvi-EO-2.0 ViT-L, i.e. the variant of size Large, has approximately 300M model parameters (i.e. 334.8M) and 159.72G FLOPs. In addition, Prithvi-EO-2.0 ViT-H, i.e. the model of size *Huge*, has approximately 600M parameters and 317.99G FLOPs. Moreover, for reference, the

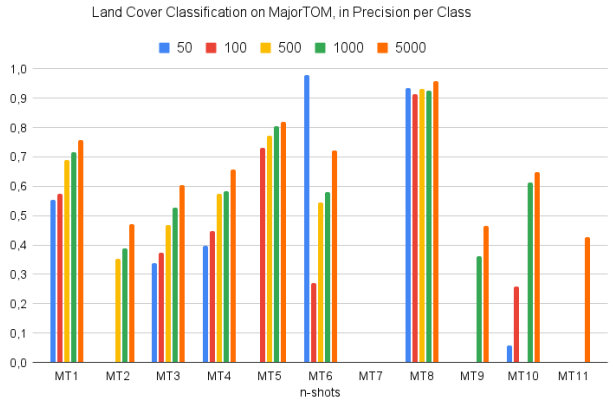

Fig. 21. Land cover mapping: PhilEO MajorTOM model precision per class.

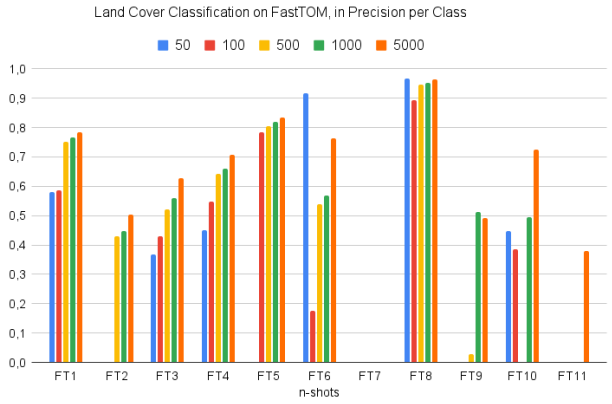

Fig. 22. Land cover mapping: PhilEO FastTOM model precision per class.

model CROMA ViT-B 100M (i.e. 116.3M) [27] has 43.83G FLOPs, while CROMA ViT-L 300M has 155.13G FLOPs. Furthermore, compared to TerraMind v1.0-L 300M (i.e. during inference), which has 45.72G FLOPs, Prithvi-EO-2.0 ViT-L 300M—where both have 300M parameters and also are *Transformer*-based—, has 159.72G FLOPs, and here this is approximately 3.5 times larger number of FLOPs [29], [23].

We evaluate and compare with Prithvi-EO-2.0 600M TL in Figs. 18-20 on the PhilEO Bench downstream tasks (S-2, 10 spectral bands): road and building density, and land cover.

F. Discussion of main findings, and per-class results

Main observations and findings from the results, as well as from our scaling-up study. According to the results in this section, the performance of the models, as well as the important decision to select *one* of the models, i.e. the best one, depends on the downstream task and also on the n-shot samples number, and this is important. This holds for example in Fig. 15 for the n-shot 5000 for the Mamba model, as well as in Figs. 7 and 10 for the low-shot samples for the Geo-Aware U-Net model. One of the main conclusions of this paper is that the U-Net CNN 200M model achieves the *best* results for few- and low-shot samples in road and building density estimation in Figs. 7 and 10, but *not* for the land cover mapping task. An additional main finding of this paper is that Mamba 2D models are effective and can achieve good results, and in addition, they are also efficient (i.e. compared to ViT), *fast* and scalable.

ViT models in general are good for *long*-range dependencies and correlations. However, in EO images, we have such global correlation for roads, rivers, time, and *global* understanding of the image (for example, if urban or rural area, or forest or *desert*). In EO, we mainly have local correlations, thus CNN U-Net models and not ViT models lead to top/ very good performance. We note that the CNN U-Net has better performance than ViT also in [29] (e.g. *Table 6* in this paper).

Per-class results for land cover mapping. We also evaluate and examine the precision for each class across different n-shot settings in Fig. 21, focusing on the PhilEO 44M MajorTOM 23TB model's performance on the PhilEO Bench land cover semantic segmentation task, which includes 11 classes based on the ESA WorldCover classification. Similarly, Fig. 22 presents the class-wise precision metrics for the PhilEO 44M FastTOM 2TB model on the same land cover semantic segmentation task. The results from the figures demonstrate that precision *improves* across most of the classes as the number of n-shots increases. Here, we also note that the 11 classes in the WorldCover semantic segmentation dataset in Figs. 21 and 22 are: Tree cover, Shrubland, Grassland, Cropland, Built-up, Bare/sparse vegetation, Snow and ice, Permanent water bodies, Herbaceous wetland, Mangrove, and Moss and lichen.

Summary and final discussion. We have examined in this paper the effect of scaling models and data: i) Model size and architecture, and ii) Data size. The main reason for wide scaling is to include oceans and ice in addition to land, and we note that we defer for future work ocean downstream tasks, i.e. MADOS dataset [66] for marine litter detection. In this work, the main research questions (which we answer affirmatively to both) are: "Is it feasible to have a model tackling a wider variety of downstream tasks (i.e. oceans and ice) compared to smaller models?", and "Does the extra information in MajorTOM 23TB (oceans, ice) make it possible to use the model?". For the *former* research question, we note that the larger model is able to tackle a wider variety of downstream tasks. Our focus is on examining if the downstream tasks can be performed at the same level (or better) with a larger model.

VI. Conclusion

In this work, we studied scaling-up the pretraining data for GFMs, as well as using a larger model (i.e. more model parameters) and employing a different architecture (e.g., Mamba SSM). The results in this paper underline the critical role of high-performance computing infrastructure and optimized training strategies in scaling-up EO AI systems (i.e. we used the Leonardo supercomputer davinci-1), paving the way for future work on even larger models and more complex downstream applications. We also computed, studied, and examined the FLOPs needed by the various different models (including in Sec. V), and this is also a main finding of this paper. Our study confirms that pretraining on large, diverse datasets—combined with appropriate architectural choices—enables the development of more robust and generalizable GFMs. By scaling the pretraining dataset from

0.5TB (PhilEO Globe) to 23TB (MajorTOM), as well as to 2TB (FastTOM), we achieved improvements in road and building density regression when labeled data was scarce. For land cover mapping, models pretrained on the specialized land-only dataset PhilEO Globe performed better than the MajorTOM-pretrained model, due to the prior exposure to both seasonality and land-specific imagery.

In this paper, we also compared with Mamba models (see Sec. III-C, as well as Sec. V-D), and this also one of our main contributions. Furthermore, we think our FLOPs profiling analysis, comparison, and benchmarking (i.e. in Sec. V, as well as in Figs. 8 and 9) opens the road to fairly evaluating different models (including GFMs) and will inspire others to adopt and use this real-world setting. Finally, it was a major achievement to process the 23TB of MajorTom on 32 cores of the Leonardo supercomputer davinci-1, in many days. This would have taken *many* more days, and even more months, on a single GPU. As future work, we plan to use JEPA (see Sec. IV), as well as to add multi-temporal capability (i.e. multiple time steps) [58], and multi-modality. Regarding the latter, we will add S-1 in addition to S-2 in the pretraining and not only in the S12-PhilEOBench (i.e. see Sec. III-D). As future work, we will also use Mamba 2D models with larger images, for example 512x512 (or even 1024x1024), as well as perform knowledge distillation of models to the same number of FLOPs (e.g., for On-Board AI) and compare and benchmark. Finally, as future work, we plan to also examine, compare with, and try to include in the model pipeline the recent AlphaEarth Foundations Satellite Embeddings[10]. These are geo-located embeddings at 10m resolution, and they also need the longitude and latitude information to be used in the downstream tasks.

REFERENCES

[1] Kaiming He, et al., "Masked Autoencoders Are Scalable Vision Learners," *IEEE/CVF Conference CVPR*, p. 15979–15988, 2023.

[2] Phúc H. Lê Khac, et al., "Contrastive Representation Learning: A Framework and Review," *IEEE Access*, vol. 8, p. 193907–193934, 2020.

[3] C. Fibaek, et al., "PhilEO Bench: Evaluating Geo-Spatial Foundation Models," *in Proc. IEEE Int Geosc. Rem. Sens. Sympos. (IGARSS)*, 2024.

[4] Xin Guo, et al., "SkySense: A Multi-Modal Remote Sensing Foundation Model Towards Universal Interpretation for EO," in Proc. CVPR, 2024.

[5] Xian Sun, et al., "RingMo: A Remote Sensing Foundation Model With Masked Image Modeling," *IEEE Trans Geosc Rem Sens*, vol. 61, 2023.

[6] Johannes Jakubik, Sujit Roy, et al., "Foundation Models for Generalist Geospatial Artificial Intelligence," *Arxiv*, abs/2310.18660, 2023.

[7] Yezhen Cong, et al., "SatMAE: Pre-training Transformers for Temporal and Multi-Spectral Satellite Imagery," *Arxiv*, abs/2207.08051, 2022.

[8] Olaf Ronneberger, et al., "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Arxiv*, abs/1505.04597, 2015.

[9] A. Dosovitskiy, et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in Proc. ICLR, arXiv:2010.11929, 2021.

[10] T. Xiao, et al., "Unified Perceptual Parsing," *arXiv:1807.10221*, 2018.

[11] Jianjian Yin, et al., "Swin-TransUper: Swin Transformer-based UperNet for medical image segmentation," *Multim Tools and Applications*, 2024.

[12] Ze Liu, et al., "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," in Proc. ICCV, 2021.

[13] Ruiping Yang, et al., "ViT-UperNet: A hybrid vision transformer with unified-perceptual-parsing network," *Complex & Intellig. Systems*, 2024.

[14] Liyuan Geng, Jinhong Xia, and Yuanhe Guo, "Applying ViT in Generalized Few-shot Semantic Segmentation," arXiv:2408.14957, 2024.

[15] A. Francis and M. Czerkawski, "Major TOM: Expandable datasets for Earth observation," in Proceedings IEEE Conf IGARSS, 2024.

[16] I. Tsiporenko, et al., "Going Beyond U-Net: Assessing ViTs for Semantic Segmentation in Microscopy Image Analysis," *ArXiv*, abs/2409.16940.

[17] Marc Rußwurm, et al., "Meta-Learning for Few-Shot Land Cover Classification," *IEEE/CVF Conf Comp. Vision P. Rec. W (CVPRW)*, 2020.

[18] Marc Rußwurm, et al., "Humans are Poor Few-Shot Classifiers for Sentinel-2 Land Cover," in Proc. Conf IGARSS, p. 4859-4862, 2022.

[19] Tianyi Gao, et al., "Enrich, Distill and Fuse: Generalized Few-Shot Semantic Segmentation in Remote Sensing," *IEEE/CVF CVPRW*, 2024.

[20] Zhuoyan Xu, et al., "Towards Few-Shot Adaptation of Foundation Models via Multitask Finetuning," *ArXiv*, abs/2402.15017, 2024.

[21] Tsung-Yi Lin, et al., "Feature Pyramid Networks for Object Detection," *IEEE Conf Computer Vision and Pattern Recognition (CVPR)*, 2017.

[22] Hengshuang Zhao, et al., "Pyramid Scene Parsing Network," *IEEE Conf Computer Vision and Pattern Recognition (CVPR)*, p. 6230-6239, 2017.

[23] Daniela Szwarcman, et al., "Prithvi-EO-2.0: A Versatile Multi-Temporal Foundation Model for EO Applications," arXiv:2412.02732, 2025.

[24] Oscar Manas, et al., "Seasonal contrast: Unsupervised pre-training from uncurated remote sensing data," in Proc. ICCV, 2021.

[25] Isaac Corley and Caleb Robinson, "Hydro Foundation Model," GitHub, AGU, 2024. http://github.com/isaaccorley/hydro-foundation-model

[26] M. Noman, et al., "Rethinking Transformers Pre-training for Multi-Spectral Satellite Imagery," in Proc. CVPR, 2024.

[27] A. Fuller, et al., "CROMA: Remote Sensing Representations with Contrastive Radar-Optical Masked Autoencoders," Proc. NeurIPS, 2023.

[28] Yi Wang, Zhitong Xiong, et al., "Towards a Unified Copernicus Foundation Model for Earth Vision," arXiv:2503.11849, 2025.

[29] J. Jakubik, F. Yang, B. Blumenstiel, et al., "TerraMind: Large-Scale Generative Multimodality for Earth Observation," arXiv:2504.11171, 2025.

[30] Haohao Qu, Liangbo Ning, et al., "A Survey of Mamba," arXiv:2408.01129, 2025.

[31] Shriyank Somvanshi, et al., "From S4 to Mamba: A Comprehensive Survey on Structured State Space Models," arXiv:2503.18970, 2025.

[32] MD Maklachur Rahman, et al., "Mamba in Vision: A Comprehensive Survey of Techniques and Applications," arXiv:2410.03105, 2024.

[33] Xiao Liu, Chenxu Zhang, and Lei Zhang, "Vision Mamba: A Comprehensive Survey and Taxonomy," arXiv:2405.04404, 2024.

[34] Lianghui Zhu, et al., "Vision Mamba: Efficient Visual Representation Learning with Bidirectional State Space Model," in Proc. ICML, arXiv:2401.09417, 2024.

[35] Yue Liu, et al., "VMamba: Visual State Space Model," in Proc. NeurIPs, arXiv:2401.10166, 2024.

[36] Enis Baty, et al., "Mamba2D: A Natively Multi-Dimensional State-Space Model for Vision Tasks," arXiv:2412.16146, 2025.

[37] Jingwei Zhang, et al., "2DMamba: Efficient State Space Model for Image Representation with Applications on Giga-Pixel Whole Slide Image Classification," in Proc. CVPR, arXiv:2412.00678, 2025.

[38] Chengkun Wang, et al., "V2M: Visual 2-Dimensional Mamba for Image Representation Learning," arXiv:2410.10382, 2024.

[39] Shabnam Choudhury, et al., "REJEPA: A Novel Joint-Embedding Predictive Architecture for Efficient Remote Sensing Image Retrieval," arXiv:2504.03169, 2025.

[40] Guillaume Astruc, et al., "AnySat: One Earth Observation Model for Many Resolutions, Scales, and Modalities," in Proc. CVPR, arXiv:2412.14123, 2025.

[41] Mahmoud Assran, et al., "Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture," in Proc. CVPR, arXiv:2301.08243, 2023.

[42] Xuan-Thuy Vo, et al., "Efficient Vision Transformers with Partial Attention," in Proc. European Conference on Computer Vision (ECCV), p. 298-317, 2024.

[43] Tri Dao, et al., "FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness," in Proc. Neural Information Processing Systems (NeurIPS), arXiv:2205.14135, 2022.

[44] Weihao Yu, et al., "MetaFormer Is Actually What You Need for Vision," in Proc. CVPR, arXiv:2111.11418, 2022.

[45] Sinong Wang, et al., "Linformer: Self-Attention with Linear Complexity," arXiv:2006.04768, 2020.

[46] Dongchen Han, et al., "FLatten Transformer: Vision Transformer using Focused Linear Attention," in Proc. ICCV, arXiv:2308.00442, 2023.

[47] Badri Narayana Patro, et al., "SpectFormer: Frequency and Attention is what you need in a Vision Transformer," in Proc. WACV, arXiv:2304.06446, p. 9525-9536, 2025.

[10]http://deepmind.google/discover/blog/alphaearth-foundations-helps-map-our-planet-in-unprecedented-detail/

[48] Xianping Ma, et al., "RS3Mamba: Visual State Space Model for Remote Sensing Image Semantic Segmentation," IEEE Geoscience and Remote Sensing Letters, vol. 21, n. 6011405, 2024. DOI: 10.1109/LGRS.2024.3414293.
[49] Nikolaos Ioannis Bountos, et al., "FoMo: Multi-Modal, Multi-Scale and Multi-Task Remote Sensing Foundation Models for Forest Monitoring," in Proc. 39th Annual AAAI Conference on Artificial Intelligence, AI for Social Impact track, arXiv:2312.10114, n. 3104, p. 27858 - 27868, 2025.
[50] Sujit Roy, et al., "AI Foundation Model for Heliophysics: Applications, Design, and Implementation," arXiv:2410.10841, 2024.
[51] Fengxiang Wang, et al., "RoMA: Scaling up Mamba-based Foundation Models for Remote Sensing," arXiv:2503.10392, 2025.
[52] Badri Narayana Patro and Vijay Srinivas Agneeswaran, "Mamba-360: Survey of State Space Models as Transformer Alternative for Long Sequence Modelling: Methods, Applications, and Challenges," arXiv:2404.16112, 2024.
[53] Xiaochuan Tang, et al., "Mamba for landslide detection: A lightweight model for mapping landslides with very high-resolution images," IEEE Transactions on Geoscience and Remote Sensing, 2025.
[54] Feng Wang, et al., "Mamba-Reg: Vision Mamba Also Needs Registers," in Proc. CVPR, p. 14944-14953, 2025.
[55] Leiye Liu, et al., "DefMamba: Deformable Visual State Space Model," in Proc. CVPR, p. 8838-8847, arXiv:2504.05794, 2025.
[56] Chuc Man Duc and Hiromichi Fukui, "SatMamba: Development of Foundation Models for Remote Sensing Imagery Using State Space Models," arXiv:2502.00435, 2025.
[57] Yunze Liu and Li Yi, "MAP: Unleashing Hybrid Mamba-Transformer Vision Backbone's Potential with Masked Autoregressive Pretraining," in Proc. CVPR, p. 9676-9685, arXiv:2410.00871, 2025.
[58] Zhengpeng Feng, et al., "TESSERA: Temporal Embeddings of Surface Spectra for Earth Representation and Analysis," arXiv:2506.20380, 2025.
[59] Sujit Roy, Johannes Schmude, et al., "Surya: Foundation Model for Heliophysics," arXiv:2508.14112, 2025.
[60] Siqi Lu, Junlin Guo, et al., "Vision foundation models in remote sensing: A survey," IEEE Geoscience and Remote Sensing Magazine, 2025.
[61] N. Longépé, H. Alemohammad, et al., "Earth Action in Transition: Highlights from the 2025 ESA-NASA International Workshop on AI Foundation Models for EO," 10.22541/au.175346055.53428479/v1, 2025.
[62] Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, et al., "DINOv3," arXiv:2508.10104, 2025.
[63] Tanzhe Li, et al., "DAMamba: Vision State Space Model with Dynamic Adaptive Scan," arXiv:2502.12627, 2025.
[64] Gencer Sumbul, et al., "SMARTIES: Spectrum-Aware Multi-Sensor Auto-Encoder for Remote Sensing Images," in Proc. ICCV, arXiv:2506.19585, 2025.
[65] Thanh-Dung Le, et al., "Onboard Satellite Image Classification for Earth Observation: A Comparative Study of ViT Models," arXiv:2409.03901, 2025.
[66] K. Kikaki, et al., "Detecting Marine Pollutants and Sea Surface Features with Deep Learning in Sentinel-2 Imagery," ISPRS Journal of Photogrammetry and Remote Sensing, 2024.
[67] Mojgan Madadikhaljan and Michael Schmitt, "Geolocation-Aware Land Cover Classification from Sentinel-2 Images," in Proc. IGARSS, 2024. IEEE. DOI: 10.1109/IGARSS53475.2024.10640576.
[68] Cong Ma and Kayvan Najarian, "Rethinking the Long-Range Dependency in Mamba/SSM and Transformer Models," arXiv:2509.04226, 2025.
[69] Yujie Zhu, et al., "First-order State Space Model for Lightweight Image Super-resolution," arXiv:2509.08458, 2025.