How Much of a Model Do We Need? Redundancy and Slimmability in Remote Sensing Foundation Models

Leonard Hackel*
Big Data Analytics for Earth Observation
BIFOLD / TU Berlin
10587 Berlin, Germany
l.hackel@tu-berlin.de

Tom Burgert
Big Data Analytics for Earth Observation
BIFOLD / TU Berlin
10587 Berlin, Germany
t.burgert@tu-berlin.de

Begüm Demir
Big Data Analytics for Earth Observation
BIFOLD / TU Berlin
10587 Berlin, Germany
demir@tu-berlin.de

Abstract

Large-scale foundation models (FMs) in remote sensing (RS) (denoted as RS FMs) are developed following paradigms established in computer vision (CV), yet the validity of transferring CV scaling laws to RS has not been systematically examined. We hypothesize that RS FMs enter an overparameterized regime at substantially smaller scales than their CV counterparts, with task-relevant information encoded redundantly across model dimensions. To test this hypothesis, we apply post-hoc slimmability, uniform width reduction of pretrained encoder transformer blocks, as a tool to measure representational redundancy across ten state-of-the-art RS FMs on classification, segmentation, and change detection tasks. RS FMs retain on average over $85\,\%$ relative accuracy on RS classification datasets under aggressive width reduction, while masked autoencoder (MAE) and DINOv2 pretrained on natural images (denoted as CV MAE and CV DINOv2) degrade sharply on ImageNet subsets of matched class count over the same range of computational requirements. A CV MAE evaluated directly on the same RS dataset narrows but does not close the gap, indicating that both dataset characteristics and domain-specific pretraining contribute to the differences between the models. Analyses of the representation space of the models such as feature correlation, explained variance, and effective dimensionality indicate that task-relevant variance concentrates in few principal components and is redundantly encoded across model dimensions. We further show that learned slimmable training improves over post-hoc slimmability for contrastive objectives, while reconstruction-based objectives do not benefit consistently from current slimmable training protocols. Our findings establish post-hoc slimming as a practical deployment strategy for resource-constrained RS applications and as a diagnostic tool for representational redundancy in RS FMs. Code is available at `https://github.com/hackelle/slimmable-remote-sensing-foundation-models`.

*Corresponding author

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

1 Introduction

Remote sensing (RS) images acquired by satellite or airborne systems are a rich source of information to monitor the Earth surface, e.g., for climate change analysis, urban area studies, risk and damage assessment, and crop monitoring [29, 31, 42, 43]. With the rapid emergence of large-scale foundation models (FMs) trained on extensive RS image archives (denoted as RS FMs), practitioners have shifted from training single-purpose models to finetuning pretrained RS FMs [22, 28, 40]. Many of these RS FMs follow the paradigms established in computer vision (CV), most notably self-supervised learning (SSL) paradigms such as contrastive learning (Momentum Contrast (MoCo) [14]), self-distillation (DINO [6], DINOv2 [32]), and masked image modeling (MIM) (masked autoencoder (MAE) [15]), as well as multi-modal pretraining strategies inspired by CLIP [33]. In line with prevailing practices in CV, RS FMs are typically scaled by increasing parameter count, with the implicit assumption that larger models yield more universal and transferable representations. However, no prior work has tested whether the benefits of model scaling observed in CV equally transfer to RS. While both domains rely on visual data, they differ in data characteristics such as spatial and spectral resolution [35]. Recent analyses show that models trained on RS data depend predominantly on low- and mid-level features, whereas those trained on natural images rely more on higher-level semantic abstractions [3]. This suggests that RS FMs may not require the same representational capacity as their CV counterparts.

Building on these findings, we hypothesize that many RS downstream tasks require less representational capacity than CV downstream tasks, and that RS FMs distribute task-relevant information redundantly across their parameters. According to our hypothesis, uniform, task-agnostic parameter removal preserves substantial downstream performance. We therefore use uniform width reduction of pretrained encoder transformer blocks, which we call *post-hoc slimmability*, as a tool to quantify representational redundancy across ten state-of-the-art RS FMs.

Beyond its diagnostic value, post-hoc slimmability has direct practical relevance for RS. Many downstream users operate under limited computational budgets, which makes deploying or finetuning increasingly large FMs difficult. These constraints are most severe for edge and on-board satellite inference, where memory, power, and latency are limited, and for time-critical applications such as disaster response or search and rescue. By adapting pretrained models to available hardware without requiring retraining or access to the original pretraining data, post-hoc slimmability provides a practical mechanism for making FMs accessible across a broad range of deployment scenarios.

To test our hypothesis and quantify this potential, we conduct a comprehensive empirical study across ten RS FMs ranging from $23\,\mathrm{M}$ to $631\,\mathrm{M}$ parameters, evaluating post-hoc slimmability on seven downstream datasets spanning scene classification, semantic segmentation, and change detection. As shown in Figure 1a, RS FMs slimmed to $1\,\%$ of their computational budget retain on average over $85\,\%$ relative accuracy across four RS classification datasets, whereas MAE and DINOv2 pretrained on natural images (denoted as CV MAE and CV DINOv2) evaluated on ImageNet subsets of matched class count retain only $30\,\%$ to $35\,\%$ at the same budget. To isolate the contribution of domain-specific pretraining from dataset characteristics, we evaluate a CV MAE directly on m-eurosat [24] (Figure 1b). Here, the CV MAE retains $76\,\%$ relative accuracy at $1\,\%$ compute budget, narrowing but not closing the gap relative to RS FMs on the same dataset, which retain $82\,\%$ to $97\,\%$. These results indicate that both dataset characteristics and domain-specific pretraining contribute to slimmability, with dataset characteristics accounting for most of the gap.

Our main contributions are as follows:

(1) We present the first systematic study of slimmability in RS FMs, evaluating ten models across seven datasets spanning scene classification, semantic segmentation, and change detection. We show that post-hoc slimming to $1\,\%$ compute budget retains $53\,\%$ to $103\,\%$ accuracy across classification datasets, with the majority of models retaining above $85\,\%$, and that this robustness persists for other tasks. Through controlled comparisons with CV baselines under the same number of classes and on shared evaluation tasks, we show that the gap reflects both task properties and domain-specific pretraining.

(2) We compare post-hoc and learned slimmability through controlled experiments with MoCo and MAE, showing that learned slimmable training improves MoCo-based models across compute budgets but interacts poorly with MAE's reconstruction objective, particularly on complex multi-label scene classification.

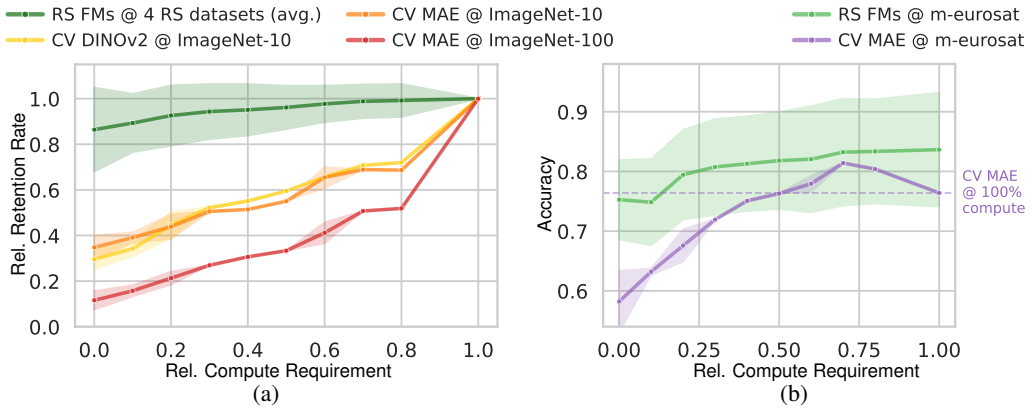


Figure 1: Post-hoc slimmability of RS FMs compared with CV baselines. Legends formatted as: "model(s) @ evaluation data". a) Relative retention rate: RS FMs across four RS classification datasets maintain on average over $85\,\%$ relative accuracy at $1\,\%$ compute budget, whereas CV DINOv2 and CV MAE retain on average only $30\,\%$ to $35\,\%$ on ImageNet-10 and a CV MAE only $12\,\%$ on ImageNet-100 at the same compute budget. b) Absolute accuracy on m-eurosat: nine RS FMs (green) retain high accuracy even at $1\,\%$ compute budget, matching or exceeding the full-scale CV MAE (dashed line at 0.76). Panopticon is excluded from this comparison because its compute scales differently with width (see Appendix A for details).

(3) We conduct a representational analysis to give an intuition on why slimming preserves performance. Through intrinsic dimensionality, explained variance, and feature correlation analysis, we show that training paradigms influence how FMs organize information across scales, indicating that task-relevant variance concentrates in few principal components and is distributed redundantly across different dimensions.

2 Related Work

Slimmable Networks and Model Compression. Slimmable neural networks enable a single model to operate at multiple widths, facilitating dynamic accuracy-efficiency trade-offs at runtime. We distinguish two paradigms: i) *post-hoc slimmability*, where uniform width reduction is applied to pretrained models without retraining; and ii) *learned slimmability*, where models are trained with explicit width sampling from scratch. Alternative compression techniques such as training-based structured pruning [26, 27], knowledge distillation [17], and quantization [19] typically require access to training data for retraining, task-specific optimization, or specialized hardware, limiting their applicability under restricted data access or varying deployment constraints. Existing work on slimmability has focused almost exclusively on the learned paradigm. Yu et al. [45] pioneer this by introducing switchable batch normalization (BN), which maintains independent BN parameters for each width, achieving competitive ImageNet accuracy through joint training. Yu and Huang [44] extend this with Universally Slimmable Networks (US-Nets), which generalize to arbitrary widths via post-statistics BN recalibration and the sandwich rule for width sampling during training. Li et al. [25] introduce Dynamic Slimmable Networks (DS-Net) with adaptive gates for test-time width selection, achieving 2–4$\times$ computation reduction on ImageNet. Extending slimmability to SSL introduces additional challenges due to the interaction between multi-width optimization and representation learning. Cao et al. [5] demonstrate that naive SSL application to slimmable networks causes training collapse, and present US3L, which implements temporally consistent guidance and dynamic width sampling to address this. Zhao et al. [46] introduce SlimCLR with slow-start training that optimizes full-width models before progressively introducing narrower sub-networks. However, these works focus exclusively on contrastive methods in CV, leaving slimmable MAE approaches and applications to RS FMs unexplored.

Post-hoc slimmability, by contrast, has received little systematic attention as a diagnostic tool for representational redundancy. We use it as such, not as a compression method: a sophisticated compression method that preserves accuracy leaves open whether the result reflects the method or the model, whereas accuracy preserved under simple uniform width reduction is evidence about the

Table 1: List of selected RS FMs. Models evaluated in this work are highlighted in gray. "—" indicates information not publicly available. $^{+}$ multi-modal; * multitemporal. For models with multiple variants, ranges indicate min–max values. Image resolution for forward pass: 224×224.

Model	**#Pretraining Images**	**Compute** Pretraining	**Parameters**	**GFLOPs/** forward pass
SatMAE [7]	713 k	768 h (V100)	330 M	—
SSL4EO [38]	251 k$^{+\star}$	100 h (A100)	23 M – 87 M	4.44 – 17.14
Satlas [1]	50.4 M$^{+\star}$	—	86 M	17.12
SkySense [12]	21.5 M^{+}	24.6 kh (A100)	2.06 B	—
Prithvi-EO-2.0 [36]	4.2 M$^{+\star}$	21 kh – 58 kh (A100)	304 M – 631 M	59.85 – 162.18
DOFA [41]	11.5 M^{+}	576 h (L40)	86 M – 330 M	17.47 – 60.47
TerraMind-1.0 [21]	9 M^{+}	9.2 kh (A100)	87 M – 305 M	17.83 – 61.73
Panopticon [37]	2.5 M^{+}	—	86 M	59.16
CopernicusFM [39]	18.7 M^{+}	512 h (A100)	86 M	17.67
RingMoE [2]	400 M^{+}	—	14.7 B	—
CSMoE [13]	1.1 M^{+}	—	271 M – 277 M	2.92 – 13.40

model itself. The closest line of work is training-free structured pruning of pretrained large language models [30, 47] for deployment, and the recent use of such pruning as a probing methodology in unified multimodal models [16]. Additionally, in RS, scaling has so far only been studied during pretraining: Dionelis et al. [9] pretrain separate models of 44 M to 300 M parameters (U-Net, ViT-UPerNet, Mamba) on three dataset scales and find that a 200M-parameter U-Net outperforms larger architectures in low-shot regression, and Burgert et al. [4] find that the downstream performance of contrastive pretraining for a ResNet-18 on multispectral Sentinel-2 imagery saturates at roughly 100 k to 200 k images. Both results are compatible with, but do not test, the redundancy we observe. We instead take single released checkpoints and ask how much of their width is needed at inference without retraining, so the two views are complementary. However, no prior work has investigated post-hoc slimmability as a property of pretrained models in RS, leaving open whether representational redundancy alone is sufficient to sustain performance under width reduction without any retraining.

Remote Sensing Foundation Models. With advances in SSL and the increasing availability of large-scale RS data, the development of FMs has gained significant attention for learning general-purpose RS representations. These models leverage SSL objectives such as contrastive learning or MIM for pretraining by using different data modalities. As shown in Table 1, widely used RS FMs span multiple orders of magnitude in parameters and computational budgets. For instance, Prithvi [20, 36] leverages Sentinel-2 and Landsat time-series data, while Satlas [1] introduces 300M+ annotations for supervision, and SSL4EO-S12 [38] provides vision transformer (ViT) encoders pretrained via contrastive objectives on a large-scale multi-seasonal Sentinel-2 dataset. Beyond scaling, architectural innovations have focused on domain-specific adaptations. First, temporal and spectral modeling advances through SatMAE [7] with temporal-spectral encodings and Scale-MAE [34] with resolution-aware embeddings. Second, modality-agnostic designs have emerged, including DOFA [41], Panopticon [37], and CopernicusFM [39], which dynamically adapt to channel configurations, and TerraMind [21], which enables cross-modal generation. Third, multi-modal integration is addressed by RingMoE [2], which employs hierarchical mixture-of-experts (MoE) with modal-specialized experts.

More recently, efficiency considerations have motivated alternative architectural approaches. For example, RS-vHeat [18] replaces attention mechanisms with heat conduction operators to reduce complexity, while CSMoE [13] achieves $2\times$ computational efficiency by utilizing the Soft MoE mechanisms. However, these models are optimized for fixed efficiency-accuracy trade-offs at design time, consequently lacking the runtime adaptability needed for dynamic resource constraints. Despite these advances, no prior research has investigated whether RS FMs exhibit different slimmability properties compared to CV models, nor explored the representational characteristics governing post-hoc slimmability in this domain.

3 Problem Setup and Evaluation Protocol

3.1 Slimmable Network Architecture

We investigate the slimmability [44] of RS FMs by implementing width reduction in transformer blocks. For this investigation, we only modify the feed-forward network (FFN) hidden dimension and the per-head dimensions of the multi-head self-attention (MHSA) layers, so that the computational budget can be varied at inference time. The embedding dimension stays unchanged, so every normalization layer operates on the same feature dimension at all widths and needs no recalibration (see Appendix A for details). In addition, we analyze the individual layer types in Appendix B.

Slimmable FFN. For a standard FFN with hidden dimension d_h, we implement a slimmable variant that operates on a reduced dimension $d'_h = \lfloor s \cdot d_h \rfloor$, where $s \in (0, 1]$ is the scaling factor. Given input $\mathbf{x} \in \mathbb{R}^{N \times d}$, the forward pass becomes:

$$\mathbf{h} = \mathrm{Act}(\mathbf{x}\mathbf{W}_1[:d'_h, :]^\top + \mathbf{b}_1[:d'_h]), \tag{1}$$

$$\mathbf{y} = \mathbf{h}\mathbf{W}_2[:, :d'_h]^\top + \mathbf{b}_2, \tag{2}$$

where $\mathbf{W}_1 \in \mathbb{R}^{d_h \times d}$, $\mathbf{W}_2 \in \mathbb{R}^{d \times d_h}$, $\mathbf{b}_1 \in \mathbb{R}^{d_h}$, and $\mathbf{b}_2 \in \mathbb{R}^d$.

Slimmable MHSA. For MHSA with H heads and head dimension d_k, we reduce each head to $d'_k = \lfloor s \cdot d_k \rfloor$ while keeping all H heads. For head h, we select per-head row indices $\mathcal{I}_h = [h \cdot d_k, h \cdot d_k + d'_k)$ and define $\mathcal{I} = \bigcup_{h=1}^{H} \mathcal{I}_h$. The query, key, value, and output projections are sliced as:

$$\mathbf{Q} = \mathbf{x}\mathbf{W}_Q[\mathcal{I}, :]^\top + \mathbf{b}_Q[\mathcal{I}], \tag{3}$$

$$\mathbf{K} = \mathbf{x}\mathbf{W}_K[\mathcal{I}, :]^\top + \mathbf{b}_K[\mathcal{I}], \tag{4}$$

$$\mathbf{V} = \mathbf{x}\mathbf{W}_V[\mathcal{I}, :]^\top + \mathbf{b}_V[\mathcal{I}], \tag{5}$$

$$\mathbf{x}_{\text{attn}} = \mathrm{softmax}(\mathbf{Q}\mathbf{K}^\top / \sqrt{d'_k})\mathbf{V}, \tag{6}$$

$$\mathbf{y} = \mathbf{x}_{\text{attn}}\mathbf{W}_O[:, \mathcal{I}]^\top + \mathbf{b}_O, \tag{7}$$

where $\mathbf{x} \in \mathbb{R}^{N \times d}$, $\mathbf{W}_{\{Q,K,V\}} \in \mathbb{R}^{(H \cdot d_k) \times d}$, $\mathbf{W}_O \in \mathbb{R}^{d \times (H \cdot d_k)}$, and the biases have matching dimensions. Both slimming operations retain the first d'_h dimensions of the FFN hidden layer and the first d'_k dimensions of each attention head. We empirically verify in the Appendix C that first, last, and random orderings yield comparable performance across models and scales, with any ordering effect small relative to the retention gap between RS FMs and CV models.

Post-hoc vs. Learned Slimmability. We distinguish two paradigms: i) *post-hoc slimmability* applies uniform width reduction to pretrained encoder transformer blocks while preserving learned weight values, requiring no retraining; and ii) *learned slimmability* trains models from scratch with slimmable layers and explicit width sampling during training, encouraging weight sharing and robustness across scales. Details of the learned slimmability training protocol are given in the Appendix D.

3.2 Evaluation Protocol

Datasets. We evaluate our hypothesis on seven downstream datasets spanning three task types from the geobench benchmark collection [24] and the OSCD dataset [8]. In detail, we select: i) m-brick-kiln (binary brick-kiln/no-brick-kiln classes), m-eurosat (single-label, 10 land-use/land-cover (LULC) classes), m-so2sat (single-label, 17 local climate zone classes), and m-bigearthnet (multi-label, 43 LULC classes) for scene classification; ii) m-cashew-plant (7 classes) and m-SA-crop-type (10 crop-type classes) for semantic segmentation; and iii) OSCD (binary change/no-change classes) for change detection (CD).

Feature Extraction and Task Heads. For each model at scale s, we extract features from the frozen encoder. For classification, we use the penultimate layer and train a k-nearest neighbors (k-NN) classifier; we report macro-averaged accuracy for single-label datasets and macro-mean average precision (mAP_M) for m-bigearthnet. For dense predictions (semantic segmentation and CD), we extract multi-scale features from four equally spaced encoder layers and train a lightweight head that fuses them via concatenation and a linear projection before classifying each spatial position. Logits are bilinearly upsampled to the input resolution. For CD, both bi-temporal images are processed and the

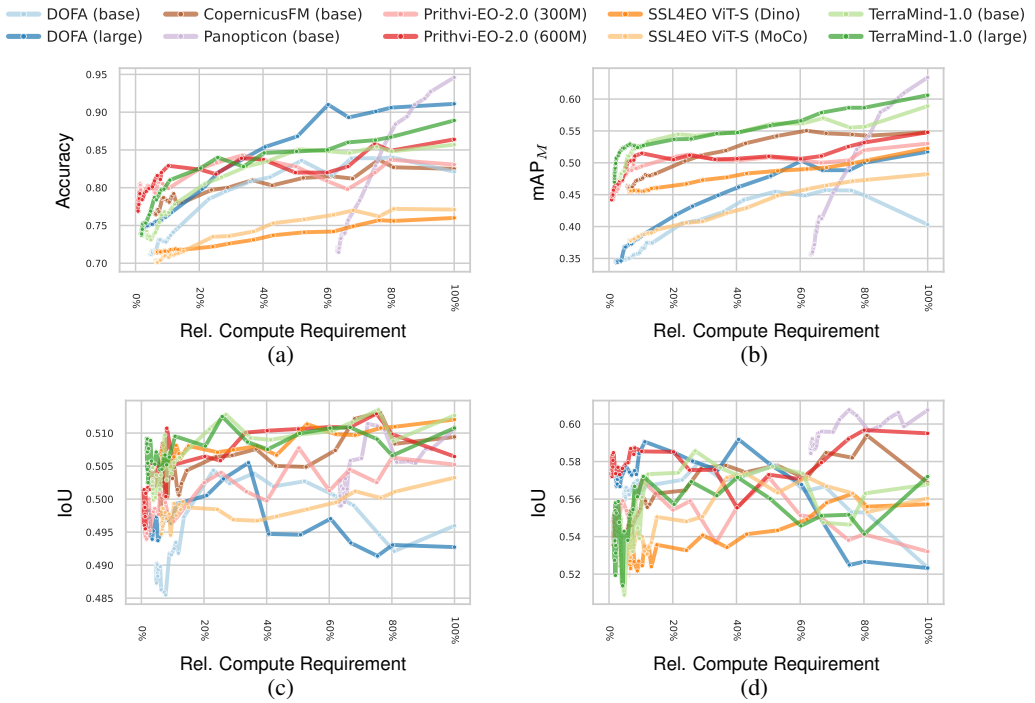


Figure 2: Downstream performance with respect to relative compute requirements of ten pretrained RS FMs on: a) m-eurosat (single-labels scene classification); b) m-bigearthnet (multi-label scene classification); c) m-cashew-plant (semantic segmentation); d) OSCD (change detection). Results for m-brick-kiln, m-so2sat, and m-SA-crop-type are provided in Appendix H.

feature difference is fed to the head. The heads are trained for 30 epochs with AdamW (learning rate 0.01; weight decay 0.01 for segmentation, 0.1 for CD) using cross-entropy loss (class-weighted for CD to address the severe class imbalance in OSCD). We report mean intersection-over-union (IoU) for both tasks. Details on the k-NN protocol, segmentation head and linear probing results are provided in the Appendix E and F.

Metrics. We evaluate each model at 31 scales s ranging from $0.1\,\%$ to $100\,\%$ (see the Appendix E.1). We report the *relative compute requirement* as the ratio of floating-point operations (FLOPs) at scale s to FLOPs at full scale, and the *relative retention rate* $\text{RetentionRate}(s) = \text{metric}(s)/\text{metric}(1.0)$, which we use when comparing datasets with different absolute performance levels.

Representational Analysis. To characterize how RS FMs behave under width reduction, we describe three representational properties of the features at each scale (see Appendix G for details): i) the mean feature correlation, i.e., the mean of $|\text{corr}(\mathbf{e}_i, \mathbf{e}_j)|$ over all pairs $i < j$ (without self-correlations), measuring average redundancy between retained dimensions; ii) the log-log slope of the explained variance ratio (EVR), i.e., the slope of $\log(\text{EVR})$ against $\log$ of the principal component index, where steeper negative values indicate that variance is concentrated in fewer components (per-scale EVR curves are provided in Appendix K); and iii) the effective dimensionality $d_{\text{eff}} = (\sum_i \lambda_i)^2 / \sum_i \lambda_i^2$, measuring how uniformly variance is spread across dimensions. We would like to note that these analyses are descriptive only and do not establish why slimming preserves performance.

4 Experiments

4.1 Post-Hoc Slimmability of Existing RS FMs

We evaluate the post-hoc slimmability of ten state-of-the-art RS FMs across four classification, two segmentation, and one change detection datasets. As shown in Figure 2, RS FMs exhibit robust post-hoc slimmability across all tasks, maintaining substantial performance even at extreme slimming. We show the results obtained by using two classification, one segmentation, and one change detection

dataset here, with the results of the remaining two classification and one segmentation dataset provided in the Appendix H.

Classification. All models with a full compute range maintain $82\,\%$ to $97\,\%$ relative retention on m-eurosat at their minimum compute budget (Figure 2a), with gradual degradation beginning below $20\,\%$ compute. Panopticon is the exception: its unslimmed embedding keeps relative compute above roughly $60\,\%$ (Section A), and it retains $76\,\%$ at its smallest scale. On the more challenging multi-label m-bigearthnet (Figure 2b), eight of nine models retain $78\,\%$ to $87\,\%$ at minimum compute budget. DOFA (large) shows a larger drop to $67\,\%$, indicating that this model is more sensitive to extreme width reduction on multi-label tasks, and Panopticon retains $57\,\%$ at its smallest scale. Both datasets exhibit non-monotonic behavior where intermediate scales frequently outperform the respective full-scale models, except for Panopticon, whose performance increases monotonically with compute. This behavior is consistent with full-scale models encoding redundant features. The critical compute threshold below which performance drops sharply appears to track dataset complexity. In detail, minimum compute suffices for stable performance on m-eurosat, while m-bigearthnet requires $2\,\%$ to $10\,\%$ compute. The exception is Panopticon which cannot be assessed here, as it does not reach this range. Results for m-brick-kiln and m-so2sat follow the same pattern and are provided in the Appendix H.

Dense Prediction. Slimmability extends to dense prediction with even higher robustness than classification. On m-cashew-plant (Figure 2c), all RS FMs retain $97\,\%$ to $101\,\%$ of full-scale IoU at their respective smallest scale, including Panopticon at its smallest scale (relative compute above roughly $60\,\%$, see Appendix A), with curves effectively indistinguishable from full scale across the entire compute range. The more complex 10-class m-SA-crop-type shows reduced but still robust retention of $78\,\%$ to $94\,\%$ at minimum compute budget, with Panopticon retaining $84\,\%$ at its smallest scale (Appendix H). On OSCD (Figure 2d), all models retain almost full performance of full-scale IoU at minimum compute budget. Both DOFA models, CopernicusFM, and Prithvi-EO-2.0 (300M) exceed full scale, while Prithvi-EO-2.0 (600M), Panopticon, both TerraMind models, and both SSL4EO models retain more than $90\,\%$. The non-monotonic pattern mirrors that observed across classification tasks, which shows that slimmability is not specific to a single task type.

Model scale and architecture effects. Larger models within each architecture (DOFA (large), Prithvi-EO-2.0 (600M), TerraMind-1.0 (large)) generally show smoother degradation curves than their smaller counterparts across classification tasks. DOFA exhibits the most pronounced non-monotonic peaks across all evaluated tasks, and full-scale performance on m-bigearthnet falls below the maximum for DOFA (base). Prithvi maintains high retention at extreme slimming on classification, while CopernicusFM and both SSL4EO models degrade smoothly with little overshoot above full scale. Panopticon increases monotonically with compute over the range it covers and reaches the highest full-scale performance on m-eurosat and m-bigearthnet, but also shows the steepest decline.

Comparison with CV models. RS FMs degrade gradually across the compute range, retaining $82\,\%$ to $97\,\%$ on m-eurosat down to their minimum compute budget. In contrast, as shown in Figure 1a, CV MAE and DINOv2 on ImageNet at matched class count (10 classes) degrade sharply over the same range, retaining less than $35\,\%$ at $1\,\%$ compute budget; CV MAE on ImageNet-100 drops to $12\,\%$. Evaluating the CV MAE directly on m-eurosat (Figure 1b) flattens its curve relative to ImageNet but leaves it steeper than the RS FMs, with retention improving to $76\,\%$ at $1\,\%$ compute budget. The flattening when the CV model is moved to an RS evaluation domain indicates that task properties contribute to slimmability; the residual gap to the RS FMs indicates that domain-specific pretraining contributes additionally. The contrast lies in the shape of the slimming behavior rather than in any single retention value.

4.2 Learned vs Post-Hoc Slimmability

To investigate whether explicit slimmable training improves over post-hoc slimmability under controlled conditions, we train ViT-Base models from scratch under both MoCo and MAE pretraining objectives, comparing learned slimmable training against regular training. All models are trained on a curated RS dataset following [13]; an ablation on pretraining length is provided in the Appendix D. Results are shown in Figure 3; results for the remaining datasets are provided in the Appendix I.

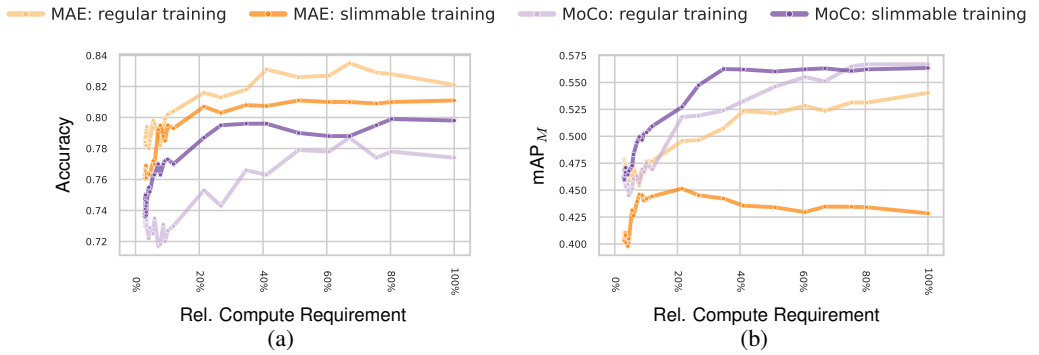


Figure 3: k-NN classification performance with respect to their relative compute requirements of a MAE-based and a MoCo-based trained model from scratch, each once with and once without slimmable training on: a) m-eurosat; and b) m-bigearthnet.

MoCo benefits consistently from learned slimmability; MAE does not. MoCo-based learned slimmable training consistently outperforms regular training at low compute budgets while maintaining equivalent or superior performance at higher scales. On m-eurosat, the advantage of slimmable MoCo over regular MoCo is modest at minimum compute budget and increases at intermediate compute budgets. On the more challenging multi-label m-bigearthnet (Figure 3b), both MoCo variants reach similar full-scale performance, but the slimmable variant achieves near-peak accuracy at substantially lower compute budgets. This pattern suggests that explicit multi-scale optimization structures MoCo representations to be more robust under width reduction without sacrificing peak performance. MAE-based learned slimmable training shows no such benefit. On m-eurosat, regular MAE outperforms learned slimmable MAE across nearly all scales. On m-bigearthnet, the slimmable MAE additionally degrades at larger scales. The advantage of MoCo over MAE on m-bigearthnet holds for both slimmable and regular training, consistent with prior findings that contrastive objectives promote greater linear separability on multi-label tasks [15, 23], and is therefore not specific to the slimmability protocol. To our knowledge, no established slimmable training procedure exists for reconstruction-based objectives, and the current protocol does not appear to transfer from contrastive to reconstructive pretraining.

4.3 Representational Analysis of Slimmability

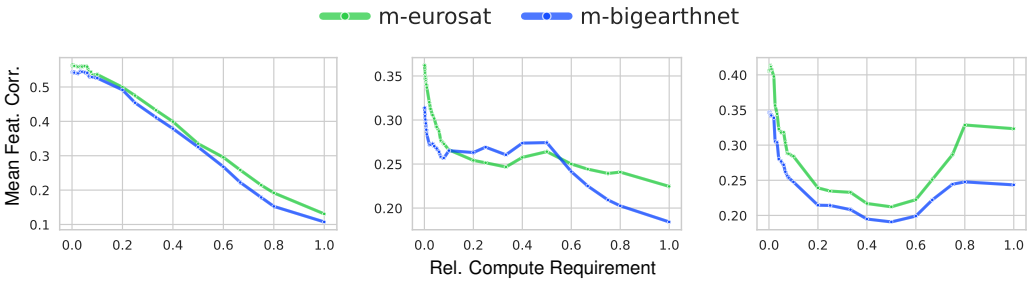


Figure 4: Mean feature correlation of the test split of m-eurosat and m-bigearthnet from the geobench benchmark collection with respect to their relative compute requirements on pretrained RS FMs: a) DOFA (large); b) TerraMind-1.0 (large); and c) Prithvi-EO-2.0 (600M).

To characterize how RS FMs exhibit robust post-hoc slimmability and why architectures degrade differently under width reduction, we analyze three complementary representational properties: i) mean feature correlation, measuring average redundancy between retained dimensions; ii) the log-log slope of the EVR, characterizing how concentrated task-relevant variance is across principal components, where steeper negative values indicate greater concentration (see Appendix K for per-scale EVR curves); and iii) effective dimensionality, measuring how many di-

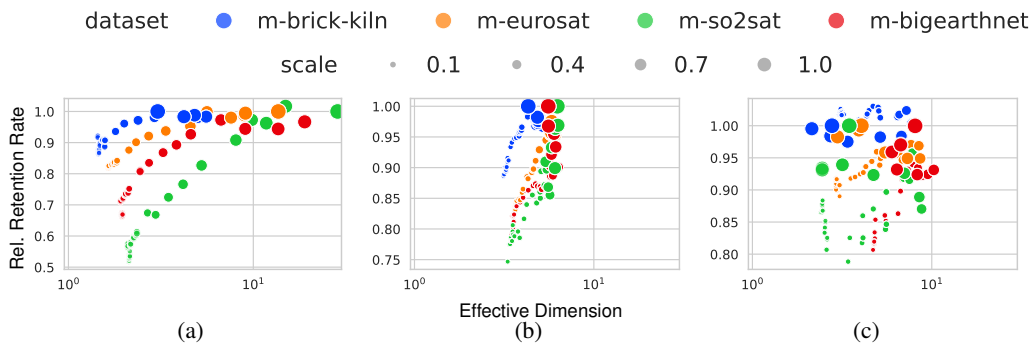


Figure 5: Effective dimension of the test split of four classification datasets from the geobench benchmark collection with respect to their relative retention rate on pretrained RS FMs: a) DOFA (large); b) TerraMind-1.0 (large);and c) Prithvi-EO-2.0 (600M).

mensions contribute independent useful information at each scale. Results for the three large models are shown in Figures 4 to 6; results for smaller models are provided in the Appendix K, G, and J and support the same narrative to a lesser degree. At minimum compute budget, all three large RS FMs converge toward high feature correlation, steep EVR slopes, and low effective dimensionality. This is consistent with slimmed representations concentrating variance in few, highly correlated dimensions while retaining most downstream performance.

DOFA (large). At minimum compute budget, DOFA (large) shows high feature correlation ($\sim$0.56 on m-eurosat; Figure 4a), a steep EVR slope (~ -3.1; Figure 6), and effective dimensionality below 2 (Figure 5a). As compute budget increases, all three measures change monotonically: correlation drops to near-zero, the slope rises to ~ -1.1, and effective dimensionality expands to 34 on m-bigearthnet at full scale. Retention and effective dimensionality are strongly correlated across scales, indicating that DOFA (large) requires many independent dimensions for full-scale performance.

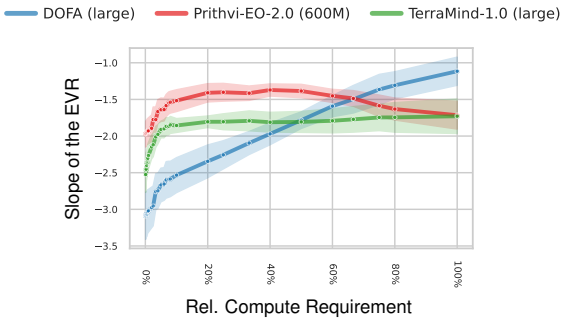


Figure 6: Log-Log-Slopes of the EVR with relative compute requirements for three different RS FMs averaged across four RS classifications datasets.

TerraMind-1.0 (large). TerraMind-1.0 (large) shows a saturating rather than monotonic structure. Feature correlation declines from moderate values ($\sim$0.36 on m-eurosat) through $10\,\%$ compute budget before stabilizing, then decreases gradually to full scale (Figure 4b). The EVR slope remains essentially flat across most of the compute range before steepening at minimum compute budget (Figure 6). Effective dimensionality rises from $\sim$3.6 at minimum compute to $\sim$5.7 by $40\,\%$ compute budget, then plateaus through full scale (Figure 5b). On m-bigearthnet it peaks mildly at intermediate scales before declining. The early plateau of effective dimensionality, together with the stable EVR slope, is consistent with TerraMind-1.0 (large) retaining a stable representational profile once a moderate compute budget is reached, as observed in Section 4.1.

Prithvi-EO-2.0 (600M). Prithvi-EO-2.0 (600M) exhibits a qualitatively distinct structure across all three analyses. Feature correlation follows a non-monotonic trajectory across the compute range (Figure 4c), and the EVR slope shows a corresponding non-monotonic pattern (Figure 6). Effective dimensionality rises to a peak of $\sim$8.6 on m-eurosat (and $\sim$10.3 on m-bigearthnet) around $40\,\%$ to $50\,\%$ compute budget, then drops sharply to $\sim$3 before recovering partially toward full scale (Figure 5c). Retention, in contrast, remains within $79\,\%$ to $103\,\%$ of full-scale across the entire compute range. This decoupling indicates that Prithvi-EO-2.0 (600M) encodes task-relevant information robustly across a wide range of active dimensions, including configurations where

effective dimensionality differs by more than $2.5\times$ at near-identical retention.
Together, these analyses show that the models differ in their representational profiles under width reduction, and that no single pattern in the analyzed properties consistently tracks retention.

5 Conclusion

We have presented the first systematic study of post-hoc slimmability in RS FMs, evaluating ten state-of-the-art models across classification, semantic segmentation, and change detection. RS FMs degrade gradually under width reduction, retaining on average over $85\,\%$ relative accuracy across four classification datasets at minimum compute budget (except Panopticon, see Appendix A). CV MAE and DINOv2 degrade sharply over the same range, retaining only $30\,\%$ to $35\,\%$ on ImageNet subsets of matched class count. On m-eurosat, where RS FMs retain $82\,\%$ to $97\,\%$, a CV MAE evaluated on it retains $76\,\%$, narrowing but not closing the gap. The shape of the slimming behavior, gradual for RS FMs and steep for CV FMs, distinguishes the two model groups clearly. Our feature correlation, EVR slope, and effective dimensionality analyses are consistent with redundant encoding in slimmed RS FMs, but their relation to retention is model-dependent, and we cannot establish a cause for slimmability. We see three plausible, non-exclusive explanations for why RS FMs differ from CV FMs. First, RS models may rely predominantly on low- and mid-level visual features that fewer independent dimensions can represent [3], whereas CV recognition draws on higher-level semantic features. Second, RS images are acquired under comparatively standardized viewing geometries and sensors, so fewer invariances may need to be learned than for natural images. Third, current RS benchmarks may not yet stress the capacity of modern FMs; richer temporal, multi-modal, or geospatial benchmarks may reveal larger benefits of scale. Class cardinality alone does not explain the gap, since CV models remain far less slimmable on ImageNet subsets matched in class count. Learned slimmable training improves over post-hoc slimmability for contrastive objectives, while reconstruction-based objectives do not benefit from current protocols.

These findings have direct practical implications. Post-hoc slimming from $5\,\%$ to $20\,\%$ of the full compute budget frequently maintains over $80\,\%$ relative accuracy without retraining, and intermediate-scale deployment often matches or exceeds full-scale performance. Computational budget can therefore be treated as a continuous deployment variable. In our experiments, the critical compute threshold tracks label-space complexity: binary m-brick-kiln tolerates the smallest budgets, 10-class m-eurosat remains effective at about $1\,\%$ compute, and 17-class m-so2sat and 43-label m-bigearthnet require roughly $2\,\%$ to $10\,\%$.

Limitations and Future Work. Our findings characterize post-hoc width redundancy in the evaluated checkpoints, not a general scaling law for RS FMs. Relative retention must be interpreted alongside absolute performance, which differs substantially across models and datasets. Our protocol slims only the FFN and MHSA layers of the encoder blocks and leaves the patch embedding unchanged, so the attainable compute reduction depends on their share of the computation. For example, in Panopticon the embedding accounts for more than $60\,\%$ of the relative compute. Our main results use k-NN, but linear probing and fine-tuning yield the same trends (Appendix F). Retention decreases under geographic distribution shift, although substantial redundancy remains, and aggregate metrics can mask class-wise degradation (Appendix L). Our study is limited to ViT-based RS FMs evaluated on optical Sentinel-2 imagery. As future work, we plan to cover other modalities such as synthetic aperture radar, non-ViT backbones (convolutional, hybrid, state-space), and generative RS models such as DiffusionSat, for which retention would have to be assessed through generation quality or learned representations rather than downstream accuracy. Finally, learned slimmable training under reconstruction-based objectives remains an open problem, as existing protocols are designed for contrastive pretraining and do not suit MAE-style objectives. Stabilizing it would extend the benefits of learned slimmability to a broader range of RS FMs.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. Post-hoc slimming reduces the computational cost of deploying remote sensing foundation models, which could lower barriers to large-scale Earth observation in sensitive contexts such as surveillance. We encourage responsible use in accordance with applicable legal and ethical frameworks.

Acknowledgements

This work is supported by the European Research Council Project Agent-BigEarth (No. 101292498) and the European Union Horizon Project PROTEUS (No. 101296397).

References

[1] Favyen Bastani, Piper Wolters, Ritwik Gupta, Joe Ferdinando, and Aniruddha Kembhavi. Satlaspretrain: A large-scale dataset for remote sensing image understanding. In *Int. Conf. Comput. Vis.*, pages 16772–16782, October 2023.

[2] Hanbo Bi, Yingchao Feng, Boyuan Tong, Mengyu Wang, Haichen Yu, Yongqiang Mao, Hao Chang, Wenhui Diao, Peijin Wang, Yue Yu, Hanyang Peng, Yehong Zhang, Kun Fu, and Xian Sun. Ringmoe: Mixture-of-modality-experts multi-modal foundation models for universal remote sensing image interpretation. *arXiv preprint arXiv:2504.03166*, 2025.

[3] Tom Burgert, Oliver Stoll, Paolo Rota, and Begüm Demir. Imagenet-trained CNNs are not biased towards texture: Revisiting feature reliance through controlled suppression. In *Adv. Neural Inform. Process. Syst.*, volume 38, 2025.

[4] Tom Burgert, Leonard Hackel, Paolo Rota, and Begüm Demir. Rank-based geographical regularization: Revisiting contrastive self-supervised learning for multispectral remote sensing imagery. In *IEEE Winter Conf. on Appl. of Comput. Vis.*, pages 4375–4385. IEEE, 2026.

[5] Yun-Hao Cao, Peiqin Sun, and Shuchang Zhou. Three guidelines you should know for universally slimmable self-supervised learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 15742–15751, June 2023.

[6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9650–9660, 2021.

[7] Yezhen Cong, Samar Khanna, Chenlin Meng, Patrick Liu, Erik Rozi, Yutong He, Marshall Burke, David Lobell, and Stefano Ermon. SatMAE: Pre-training transformers for temporal and multi-spectral satellite imagery. In *Adv. Neural Inform. Process. Syst.*, volume 35, pages 197–211, 2022.

[8] Rodrigo Caye Daudt, Bertr Le Saux, Alexandre Boulch, and Yann Gousseau. Urban change detection for multispectral earth observation using convolutional neural networks. In *IEEE Int. Geosci. Remote Sens. Symp.*, pages 2115–2118, 2018. doi: 10.1109/IGARSS.2018.8518015.

[9] Nikolaos Dionelis, Riccardo Musto, Jente Bosmans, Simone Sarti, Giancarlo Paoletti, Peter Naylor, Valerio Marsocci, Sébastien Lefèvre, Bertrand Le Saux, and Nicolas Longépé. Scaling laws for geospatial foundation models: A case study on phileo bench. *arXiv preprint arXiv:2506.14765*, 2025.

[10] Alistair Francis and Mikolaj Czerkawski. Major tom: Expandable datasets for earth observation. In *IEEE Int. Geosci. Remote Sens. Symp.*, pages 2935–2940, 2024.

[11] Carlos Gomes, Benedikt Blumenstiel, Joao Lucas De Sousa Almeida, Pedro Henrique De Oliveira, Paolo Fraccaro, Francesc Marti Escofet, Daniela Szwarcman, Naomi Simumba, Romeo Kienzler, and Bianca Zadrozny. Terratorch: The geospatial foundation models toolkit. In *IEEE Int. Geosci. Remote Sens. Symp.*, pages 6364–6368. IEEE, 2025.

[12] Xin Guo, Jiangwei Lao, Bo Dang, Yingying Zhang, Lei Yu, Lixiang Ru, Liheng Zhong, Ziyuan Huang, Kang Wu, Dingxiang Hu, et al. Skysense: A multi-modal remote sensing foundation model towards universal interpretation for earth observation imagery. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 27672–27683, 2024.

[13] Leonard Hackel, Tom Burgert, and Begüm Demir. CSMoE: An efficient remote sensing foundation model with soft mixture-of-experts. *arXiv preprint arXiv:2509.14104*, 2025.

[14] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9729–9738, 2020.

[15] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 16000–16009, 2022.

[16] Shwai He, Chaorui Deng, Ang Li, and Shen Yan. Understanding and harnessing sparsity in unified multimodal models. *arXiv preprint arXiv:2512.02351*, 2025.

[17] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[18] Huiyang Hu, Peijin Wang, Hanbo Bi, Boyuan Tong, Zhaozhi Wang, Wenhui Diao, Hao Chang, Yingchao Feng, Ziqi Zhang, Yaowei Wang, et al. RS-vHeat: heat conduction guided efficient remote sensing foundation model. In *Int. Conf. Comput. Vis.*, pages 9876–9887, 2025.

[19] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2704–2713, 2018.

[20] Johannes Jakubik, Sujit Roy, C. E. Phillips, Paolo Fraccaro, Denys Godwin, Bianca Zadrozny, Daniela Szwarcman, Carlos Gomes, Gabby Nyirjesy, Blair Edwards, Daiki Kimura, Naomi Simumba, Linsong Chu, S. Karthik Mukkavilli, Devyani Lambhate, Kamal Das, Ranjini Bangalore, Dario Oliveira, Michal Muszynski, Kumar Ankur, Muthukumaran Ramasubramanian, Iksha Gurung, Sam Khallaghi, Hanxi (Steve) Li, Michael Cecil, Maryam Ahmadi, Fatemeh Kordi, Hamed Alemohammad, Manil Maskey, Raghu Ganti, Kommy Weldemariam, and Rahul Ramachandran. Foundation models for generalist geospatial artificial intelligence. *arXiv preprint arxiv:2310.18660*, 2023.

[21] Johannes Jakubik, Felix Yang, Benedikt Blumenstiel, Erik Scheurer, Rocco Sedona, Stefano Maurogiovanni, Jente Bosmans, Nikolaos Dionelis, Valerio Marsocci, Niklas Kopp, et al. TerraMind: Large-scale generative multimodality for earth observation. In *Int. Conf. Comput. Vis.*, 2025.

[22] Licheng Jiao, Zhongjian Huang, Xiaoqiang Lu, Xu Liu, Yuting Yang, Jiaxuan Zhao, Jinyue Zhang, Biao Hou, Shuyuan Yang, Fang Liu, et al. Brain-inspired remote sensing foundation models and open problems: A comprehensive survey. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, 16:10084–10120, 2023.

[23] Taegoo Kang, Sung-Ho Bae, and Chaoning Zhang. Contrastive learning or masked autoencoder? understanding and improving self-supervised knowledge distillation. *IEEE Access*, 2026.

[24] Alexandre Lacoste, Nils Lehmann, Pau Rodriguez, Evan Sherwin, Hannah Kerner, Björn Lütjens, Jeremy Irvin, David Dao, Hamed Alemohammad, Alexandre Drouin, et al. Geo-bench: Toward foundation models for earth monitoring. In *Adv. Neural Inform. Process. Syst.*, volume 36, pages 51080–51093, 2023.

[25] Changlin Li, Guangrun Wang, Bing Wang, Xiaodan Liang, Zhihui Li, and Xiaojun Chang. Dynamic slimmable network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8607–8617, 2021.

[26] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. In *Int. Conf. Learn. Represent.*, 2017.

[27] Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang. Learning efficient convolutional networks through network slimming. In *Int. Conf. Comput. Vis.*, pages 2736–2744, 2017.

[28] Siqi Lu, Junlin Guo, James R. Zimmer-Dauphinee, Jordan M. Nieusma, Xiao Wang, Parker vanValkenburgh, Steven A. Wernke, and Yuankai Huo. Vision foundation models in remote sensing: A survey. *IEEE Int. Geosci. Remote Sens. Mag.*, pages 2–27, 2025.

[29] Derya Maktav, FS Erbek, and C Jürgens. Remote sensing of urban areas. *T & F Int. J. Remote Sens.*, 26(4):655–659, 2005.

[30] Xin Men, Mingyu Xu, Qingyu Zhang, Qianhao Yuan, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. Shortgpt: Layers in large language models are more redundant than you expect. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20192–20204, 2025.

[31] Emmanuel Omia, Hyungjin Bae, Eunsung Park, Moon Sung Kim, Insuck Baek, Isa Kabenge, and Byoung-Kwan Cho. Remote sensing in field crop monitoring: A comprehensive review of sensor systems, data analyses and recent advances. *MDPI Remote Sens.*, 15(2):354, 2023.

[32] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. 2024. ISSN 2835-8856. URL `https://openreview.net/forum?id=a68SUt6zFt`. Featured Certification.

[33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Int. Conf. Mach. Learn.*, pages 8748–8763. PMLR, 2021.

[34] Colorado J Reed, Ritwik Gupta, Shufan Li, Sarah Brockman, Christopher Funk, Brian Clipp, Kurt Keutzer, Salvatore Candido, Matt Uyttendaele, and Trevor Darrell. Scale-MAE: A scale-aware masked autoencoder for multiscale geospatial representation learning. In *Int. Conf. Comput. Vis.*, pages 4088–4099, 2023.

[35] Esther Rolf, Konstantin Klemmer, Caleb Robinson, and Hannah Kerner. Position: Mission critical – satellite data is a distinct modality in machine learning. In *Int. Conf. Mach. Learn.*, 2024. URL `https://openreview.net/forum?id=PQ0ERKKYJu`.

[36] Daniela Szwarcman, Sujit Roy, Paolo Fraccaro, Þorsteinn Elí Gíslason, Benedikt Blumenstiel, Rinki Ghosal, Pedro Henrique de Oliveira, João Lucas de Sousa Almeida, Rocco Sedona, Yanghui Kang, Srija Chakraborty, Sizhe Wang, Ankur Kumar, Myscon Truong, Denys Godwin, Hyunho Lee, Chia-Yu Hsu, Ata Akbari Asanjan, Besart Mujeci, Trevor Keenan, Paulo Arévolo, Wenwen Li, Hamed Alemohammad, Pontus Olofsson, Christopher Hain, Robert Kennedy, Bianca Zadrozny, Gabriele Cavallaro, Campbell Watson, Manil Maskey, Rahul Ramachandran, and Juan Bernabe Moreno. Prithvi-EO-2.0: A versatile multi-temporal foundation model for earth observation applications. *arXiv preprint arXiv:2412.02732*, 2024.

[37] Leonard Waldmann, Ando Shah, Yi Wang, Nils Lehmann, Adam Stewart, Zhitong Xiong, Xiao Xiang Zhu, Stefan Bauer, and John Chuang. Panopticon: Advancing any-sensor foundation models for earth observation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2229–2239, 2025.

[38] Yi Wang, Nassim Ait Ali Braham, Zhitong Xiong, Chenying Liu, Conrad M Albrecht, and Xiao Xiang Zhu. SSL4EO-S12: A large-scale multimodal, multitemporal dataset for self-supervised learning in earth observation. *IEEE Int. Geosci. Remote Sens. Mag.*, 11(3):98–106, 2023.

[39] Yi Wang, Zhitong Xiong, Chenying Liu, Adam J Stewart, Thomas Dujardin, Nikolaos Ioannis Bountos, Angelos Zavras, Franziska Gerken, Ioannis Papoutsis, Laura Leal-Taixé, et al. Towards a unified copernicus foundation model for earth vision. In *Int. Conf. Comput. Vis.*, pages 9888–9899. IEEE, 2025.

[40] Aoran Xiao, Weihao Xuan, Junjue Wang, Jiaxing Huang, Dacheng Tao, Shijian Lu, and Naoto Yokoya. Foundation models for remote sensing and earth observation: A survey. *IEEE Int. Geosci. Remote Sens. Mag.*, pages 2–29, 2025.

[41] Zhitong Xiong, Yi Wang, Fahong Zhang, Adam J Stewart, Joëlle Hanna, Damian Borth, Ioannis Papoutsis, Bertrand Le Saux, Gustau Camps-Valls, and Xiao Xiang Zhu. Neural plasticity-inspired foundation model for observing the Earth crossing modalities. *arXiv preprint arXiv:2403.15356*, 2024.

[42] Fumio Yamazaki and Masashi Matsuoka. Remote sensing technologies in post-disaster damage assessment. *Journal of Earthquake and Tsunami*, 1(03):193–210, 2007.

[43] Jun Yang, Peng Gong, Rong Fu, Minghua Zhang, Jingming Chen, Shunlin Liang, Bing Xu, Jiancheng Shi, and Robert Dickinson. The role of satellite remote sensing in climate change studies. *Nature climate change*, 3(10):875–883, 2013.

[44] Jiahui Yu and Thomas S Huang. Universally slimmable networks and improved training techniques. In *Int. Conf. Comput. Vis.*, pages 1803–1811, 2019.

[45] Jiahui Yu, Linjie Yang, Ning Xu, Jianchao Yang, and Thomas Huang. Slimmable neural networks. In *Int. Conf. Learn. Represent.*, 2019.

[46] Shuai Zhao, Linchao Zhu, Xiaohan Wang, and Yi Yang. Slimmable networks for contrastive self-supervised learning. *Int. J. Comput. Vis.*, 133(3):1222–1237, 2025.

[47] Longguang Zhong, Fanqi Wan, Ruijun Chen, Xiaojun Quan, and Liangzhi Li. Blockpruner: Fine-grained pruning for large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5065–5080, 2025.

Technical appendices and supplementary material

Appendix Overview

A Details on Layer Slimming
B Slimmability of Different Layer Types
C Implicit Dimension Ordering
D Details on Training from Scratch
E Details on the Evaluation Protocol
F Linear Evaluation and Finetuning
G Effective Dimensionality Analysis
H Additional Experiments on Post-Hoc Slimmability
I Additional Experiments on Learned vs Post-Hoc Slimmability
J Feature Correlation at different Model Scales
K Additional Experiments on Explained Variance
L Failure Modes and Edge Cases
M Computational Resources
N Licenses and Assets

A Details on Layer Slimming

Slimmed dimensions. Post-hoc slimming reduces only the FFN hidden dimension and the per-head dimensions of the MHSA layers in each encoder transformer block. The transformer embedding dimension is never changed, nor is the patch embedding layer. Consequently, every normalization layer operates on the same feature dimension at all widths and requires neither recalibration nor switchable parameters, unlike batch normalization in learned slimmable networks of other works (e.g., [45, 44, 25, 5, 46].

Panopticon. For all other models, the patch embedding accounts for only about $5\,\%$ of the parameters, so leaving it unslimmed has little influence on the reported retention. Panopticon is an exception: its patch embedding accounts for more than $66\,\%$ of the parameters. Since our protocol slims only the MHSA and FFN, the patch embedding layer is never reduced. The figures plot total relative compute rather than the slimming scale, i.e., the width fraction applied to the two layers. For most models, both axes nearly coincide as their patch embeddings are small compared to the rest of the model. For Panopticon, even the smallest scale corresponds to a relative compute of more than $60\,\%$ (e.g., Figure 2), so its curves cannot reach the low-compute regime covered by the other models. We keep the same protocol to remain comparable, as slimming the patch embedding for a single model would break comparability. Panopticon is therefore excluded from the main comparisons that plot retention against relative compute.

B Slimmability of Different Layer Types

To understand the computational contributions of different architectural components, we compare three slimming strategies: i) slimming only FFN layers while keeping MHSA at full width; ii) slimming only MHSA layers while keeping FFN at full width; and iii) slimming both layer types simultaneously. The results of these experiments are shown in Figure 7.

As one can see from the figure, our analysis reveals that MHSA and FFN layers contribute asymmetrically to both computational cost and representational capacity. Models slimmed exclusively through MHSA width reduction (blue lines) cannot achieve relative compute requirements below $60\,\%$, as the fixed-width FFN layers dominate the computational budget. Moreover, MHSA-only slimming exhibits steeper performance degradation relative to compute reduction compared to FFN-only approaches, suggesting that attention mechanisms encode more task-critical information per compute. In contrast, FFN-only slimming (orange lines) enables compression down to $35\,\%$ relative compute while maintaining more gradual performance decay. This indicates that FFN layers contribute disproportionately to FLOPs but contain more redundant parameters that can be removed with minimal performance impact.

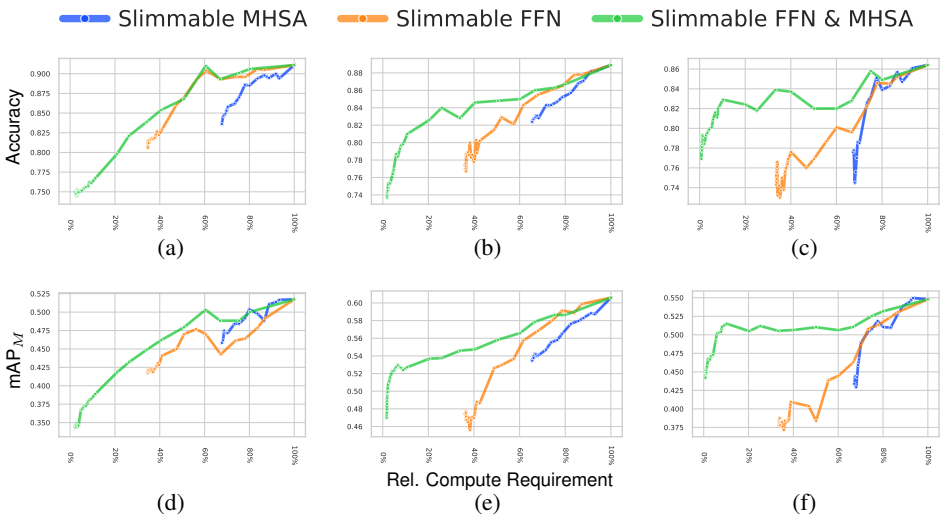


Figure 7: k-NN accuracy and mAP$_M$ with respect to their relative compute requirements for slimming only the FFN layer, only the MHSA layer, or both layers simultaneously of: a) DOFA (large); b) TerraMind-1.0 (large); and c) Prithvi-EO-2.0 (600M) on m-eurosat; and d) DOFA (large); e) TerraMind-1.0 (large); and f) Prithvi-EO-2.0 (600M) on m-bigearthnet.

However, simultaneous slimming of both layer types (green lines) proves superior across all evaluated models, enabling compression to $1\,\%$ relative compute while exhibiting the most gradual performance degradation curves. This synergistic effect suggests that maintaining architectural balance, where MHSA and FFN widths scale proportionally, preserves the learned feature processing pipeline more effectively than asymmetric compression. The ability to slim both components simultaneously also provides access to the full computational range, from 1–100 $\%$ of original FLOPs, enabling fine-grained deployment optimization. From a practical deployment perspective, practitioners should prioritize simultaneous slimming of both layer types to maximize compression ratios while maintaining performance, rather than attempting to preserve specific architectural components at full capacity.

C Implicit Dimension Ordering

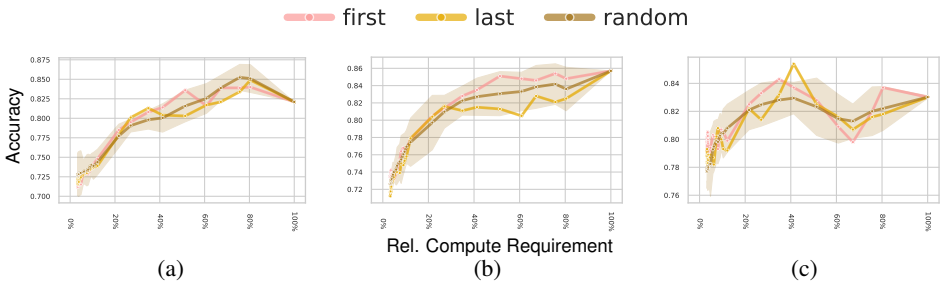


Figure 8: KNN classification performance with respect to relative compute requirements on the test split of m-eurosat when selecting the first, last, or a random subset of neurons for slimming. Random order experiments are repeated 20 times: a) DOFA (base); b) TerraMind-1.0 (base); and c) Prithvi-EO-2.0 (300M).

Figure 8 shows k-NN classification performance on m-eurosat for pretrained RS FMs when slimming retains the first d' dimensions, the last d' dimensions, or a randomly selected subset of d' dimensions, where random selection was repeated 20 times. Error bars denote 2-sigma intervals, consistent with the visualization convention used throughout this paper. For TerraMind-1.0 (base), first retention lies marginally above and last retention marginally below the 2-sigma interval of random at compute

scales below $1\,\%$, with all three conditions converging above this threshold. For DOFA (base), first and last retention fall within the 2-sigma interval of random retention across all compute scales, indicating that dimension order has no detectable effect beyond the variability of random subset selection. For Prithvi-EO-2.0 (300M), first and last retention are nearly identical across all compute scales and both lie slightly above the random 2-sigma interval only at the lowest compute scales, with no detectable ordering between them. Across all three models, the deviations from random at very low compute are inconsistent in structure: for TerraMind, first and last bracket the random interval from opposite sides, whereas for Prithvi, first and last deviate jointly in the same direction. This cross-model inconsistency rules out a shared implicit dimension ordering in the pretrained weights as an explanation, and confirms that the deviations reflect model-specific noise rather than a structural property of the slicing convention. The findings reported in Section 4.1 are therefore not an artifact of the first-dimension slicing convention.

D Details on Training from Scratch

To investigate learned slimmability (Contribution C2), we train both MoCo and MAE architectures with and without slimmable training on a curated RS dataset. We use the dataset from Hackel et al. [13], which applies entropy-maximizing stratified sampling to Major Tom Core (MTC) [10] based on Köppen-Geiger climate zones and ESA WorldCover classes, yielding approximately 100 spatially diverse samples per joined stratum and a total of $\sim 1\,\mathrm{M}$ training patches at 120×120 pixels. All models use a ViT-Base backbone with 10 input channels ($10\,\mathrm{m}$ and $20\,\mathrm{m}$ Sentinel-2 bands) and image size 224×224 pixels.

Training Protocol. Regular (non-slimmable) training is conducted for 300 epochs with batch size of 384 per GPU and AdamW optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.95$, weight decay 0.05). The learning rate follows a linear warmup for 1% of total steps, followed by cosine annealing from base learning rate $\mathrm{lr}_{\text{base}} \times \frac{\text{batch size}}{256}$ to $\frac{\mathrm{lr}_{\text{base}}}{100}$, where $\mathrm{lr}_{\text{base}} = 10^{-4}$. For MoCo, we use momentum encoder update with cosine schedule from 0.996 to 1.0 over 10 epochs, projection head dimension 128, and normalized temperature-scaled cross entropy (NT-Xent) loss with memory bank size 4096. For MAE, we use mask ratio 0.75, decoder dimension 512, decoder depth 1, and 16 decoder heads with mean squared error (MSE) reconstruction loss. All pretraining experiments were run on 4 NVIDIA A100-80GB GPUs or 4 NVIDIA H200-141GB GPUs.

Slimmable Pretraining. Learned slimmable training extends the regular protocol to 800 epochs to account for the increased optimization difficulty. During each training step, we perform manual optimization with multiple forward-backward passes: i) the full network at scale $s = 1.0$ with regular task loss; ii) the minimal network at scale s_{min} with task loss plus knowledge distillation from the full network; and iii) three randomly sampled intermediate scales with the same combined loss. The minimal scale s_{min} linearly decreases from 0.5 to 0.01 over the first $50\,\%$ of training to encourage robust feature learning across scales following [46]. Knowledge distillation uses temperature-scaled KL divergence with $T = 1.0$ between features passed through a two-layer MLP distillation head ($d \to 2d \to d$ with GELU activation). This multi-scale training encourages weight sharing and robustness across the entire range of network widths.

Evaluation. We evaluate all resulting models at multiple intermediate checkpoints (every 50 epochs for the first 300 epochs, then every 100 epochs) on the m-eurosat dataset from the geobench benchmark collections using our k-NN evaluation protocol. The results of this evaluation can be seen in Figure 9. As one can see from the figures, for regular training the accuracy of the full network increases with additional training for both MAE and MoCo architectures (Figure 9a and 9c) although only small or marginal improvements can be seen after 100 epochs. For smaller scales (with respective lower compute requirements), the accuracy is not monotonously increasing with longer training.

For slimmable training, in the case of MAE (Figure 9b) the accuracy increases across all scales for the first 200 epochs, then sharply declines by almost $20\,\%$ and then slowly increases over the following 600 epochs but never fully recovers. The slimmably trained MoCo model decreases accuracy for higher compute requirements (larger than $20\,\%$) but increases for smaller ones. Overall the accuracy equalizes for this model across all compute requirements with longer training.

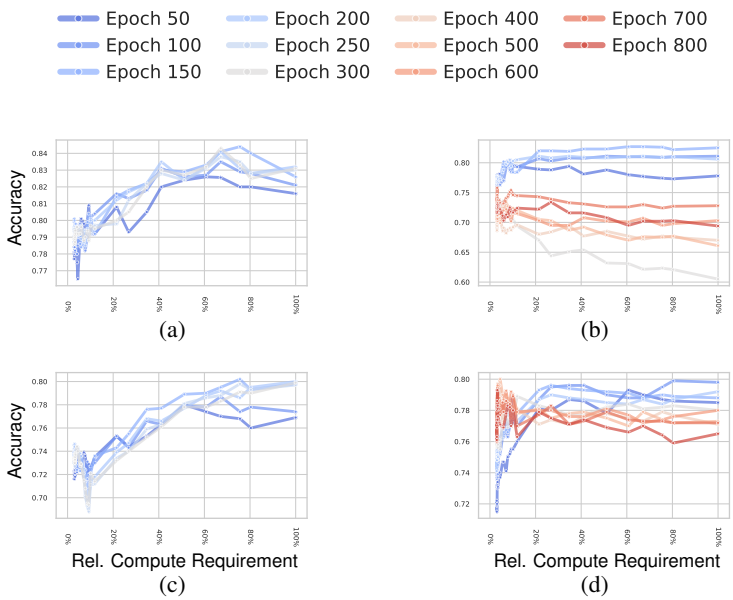


Figure 9: k-NN classification performance with respect to relative compute requirements of a MAE-based and a MoCo-based trained model from scratch, each once with and once without slimmable training on m-eurosat for different training durations: a) regular MAE; b) slimmable MAE; c) regular MoCo; d) slimmable MoCo.

E Details on the Evaluation Protocol

E.1 Scale Evaluation

During k-NN and linear evaluation, we evaluate each model on 31 scales: {0.001, 0.002, 0.0025, 0.00333, 0.004, 0.005, 0.006, 0.00667, 0.0075, 0.008, 0.01, 0.02, 0.025, 0.0333, 0.04, 0.05, 0.06, 0.0667, 0.075, 0.08, 0.1, 0.2, 0.25, 0.333, 0.4, 0.5, 0.6, 0.667, 0.75, 0.8, 1.0}. The reduced dimensions $d'_k = \lfloor s \cdot d_k \rfloor$ and $d'_h = \lfloor s \cdot d_h \rfloor$ are at minimum 1, meaning multiple scales can result in the same reduced dimension for small values of s. For finetuning only the following five scales are evaluated due to computational constraints: {0.001, 0.01, 0.1, 0.5, 1.0}.

E.2 Feature Extraction and Downstream Tasks

E.2.1 Classification

Feature Extraction. For all models and scales, we extract features from the penultimate layer (before the final classification head or pooling operation) on both training and test sets of each downstream dataset. Features are saved as NumPy arrays along with their corresponding labels to enable efficient repeated evaluation without recomputing features.

k-NN Evaluation. We use $k = 5$ neighbors and temperature parameter $t = 0.9$ for the weighted KNN classifier. Classification is performed using cosine similarity between test features and the training feature bank. For the binary dataset (m-brick-kiln) and multi-class datasets (m-eurosat, m-so2sat), we compute macro-averaged accuracy. For the multi-label m-bigearthnet dataset, we report the mAP$_M$.

Linear Probing. Linear classifiers are trained for 50 epochs using AdamW optimizer with learning rate 0.05, batch size 64, and no weight decay. For multi-class tasks, we use cross-entropy loss; for multi-label m-bigearthnet, we use binary cross-entropy with logits. We report the same metrics as for k-NN evaluation.

Finetuning. We finetune FMs for 40 epochs using AdamW optimizer with learning rate 3e-3. All other parameters are the same as for linear probing.

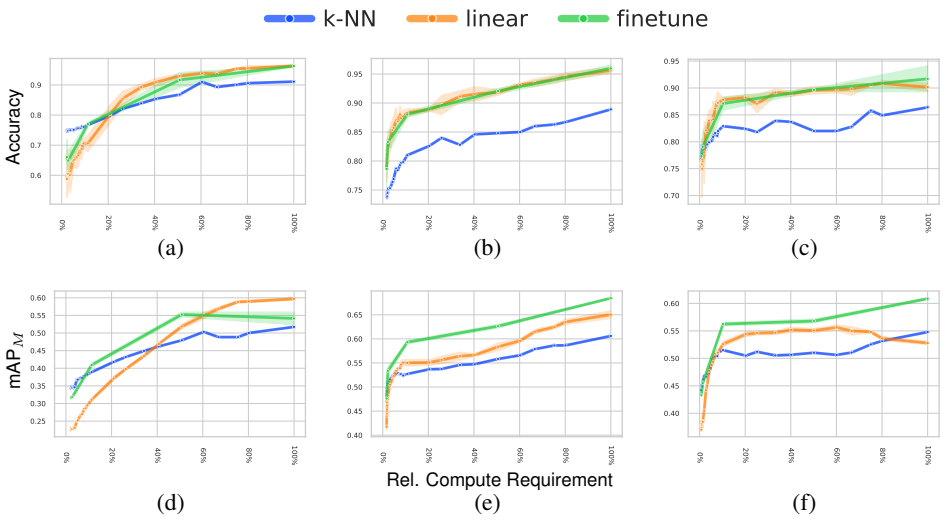


Figure 10: k-NN, linear, and finetuned classification accuracy with respect to their relative compute requirements of: a) DOFA (large); b) TerraMind-1.0 (large); and c) Prithvi-EO-2.0 (600M) on m-eurosat; and d) DOFA (large); e) TerraMind-1.0 (large); and f) Prithvi-EO-2.0 (600M) on m-bigearthnet.

E.2.2 Dense Prediction

Semantic Segmentation. Following common practice for dense prediction with frozen ViT backbones, we extract multi-scale features from four equally spaced encoder transformer blocks rather than the penultimate layer alone (indices $\mathrm{round}(i \cdot (n-1)/3)$ for $i \in \{0,1,2,3\}$, where n is the total number of blocks). Class tokens are removed and the resulting tokens are reshaped into spatial feature maps of shape $(B, 4, D, H_p, W_p)$, where $H_p = W_p = 224/p$ and p is the patch size of the respective backbone. Features are L2-normalized along the feature dimension. We train a SegFormer-style multi-layer perceptron (MLP) head that projects each of the four feature maps to a shared embedding dimension of 256, fuses them by concatenation followed by a linear projection, classifies per patch, and bilinearly upsamples to the input resolution of 224×224. Training uses AdamW with learning rate 0.01, weight decay 0.01, cosine annealing, batch size 32, and 30 epochs, with cross-entropy loss ignoring index 255. We report the IoU on the test set.

Change Detection. We extract features from both timestamps independently using the same four-block selection and reshaping procedure as for semantic segmentation, yielding tensors of shape $(N, 2, 4, D, H_p, W_p)$. Per-pixel feature differences $f_{t_2} - f_{t_1}$ are computed along the timestamp dimension, producing inputs of shape $(N, 4, D, H_p, W_p)$ that match the semantic segmentation input format. We apply a head with the same architecture as for semantic segmentation but with embedding dimension 128 and binary output. Training uses the same protocol as for semantic segmentation, with weight decay 0.1 and class weights $[1, n_{\text{neg}}/n_{\text{pos}}]$ in the cross-entropy loss to address the strong class imbalance of OSCD. We report the IoU averaged over the change and no-change classes on the test set.

E.2.3 Experimental Repeatability

Linear probing experiment is performed five times with different seeds and averaged results are reported to account for variability in both feature extraction (at different scales) and downstream evaluation. k-NN experiments are deterministic and don't require different seeds. Finetuning experments are repeated three times with different seeds. All downstream evaluations were run on single NVIDIA A40-48GB GPUs.

F Linear Evaluation and Finetuning

We evaluate DOFA (large), TerraMind-1.0 (large), and Prithvi-EO-2.0 (600M) on m-eurosat and m-bigearthnet on k-NN, linear probing, and finetuning using the evaluation protocol described in Section E.2. The results of this evaluation are shown in Figure 10. As one can see from the figure, for all evaluations except for m-bigearthnet on Prithvi-EO-2.0 (600M) the linear classification accuracy is higher for larger scales (with larger compute requirements). Still, for m-bigearthnet on Prithvi-EO-2.0 (600M) the linear classification accuracy is higher for $0.0667 \leq s \leq 0.8$. Additionally, for most cases there is a specific size $\tilde{s}$ where the k-NN accuracy is higher if $s < \tilde{s}$ and lower if $s > \tilde{s}$. Only for m-eurosat on Terramind-1.0 (large) the linear classification accuracy is higher than the k-NN accuracy for all sizes and for m-bigearthnet on Prithvi-EO-2.0 (600M) there are three sizes ($5\,\%$, $6\,\%$, and $100\,\%$) that do not follow this monotonicity.

For finetuning, in both datasets and all models the accuracy matches or exceeds the accuracy of the linear probing. Only for m-bigearthnet and DOFA (large) the linear probing model significantly outperforms the finetuned model. Across all evaluated dataset-model combinations, the finetuned models show the same accuracy drop for very small scales as was seen in k-NN and linear probing experiments.

To summarize, in general pretrained models benefit from linear probing only for larger sizes whereas k-NN evaluation is stronger for models which were more rigorously slimmed. However, the size $\tilde{s}$ where smaller sizes benefit from k-NN evaluation is not consistent for a specific model or dataset. Finetuned models also have a decrease in accuracy at small scales but are not as effected as models using linear probing.

G Effective Dimensionality Analysis

Figure 11 relates effective dimensionality to relative retention across the evaluated RS FMs. The analysis is descriptive: it shows how variance is distributed across feature dimensions, not which dimensions carry task-relevant information.

DOFA. DOFA models (Figures 11a and 11b) concentrate variance in few dimensions: effective dimensionality is between 1.4 and 2.0 at $1\,\%$ compute, while $53\,\%$ to $92\,\%$ of full-scale performance is retained. Reaching full performance goes along with a much larger effective dimensionality, up to 34 for the large model on the multi-label m-bigearthnet (roughly 16–$24\times$ the minimum-compute value).

SSL4EO. The SSL4EO ViT-S models (Figures 11g and 11h) have the lowest variance of effective dimensionality of all evaluated models, within 2.4 to 4.4 across datasets and compute scales, with monotonic trajectories. On the simpler m-eurosat and m-brick-kiln, both models retain $91\,\%$ to $94\,\%$ at $1\,\%$ compute with effective dimensionalities of 2.8 to 3.5, which increase to 3.2 to 4.4 at full scale. On the more complex m-bigearthnet and m-so2sat, retention at low compute drops to $78\,\%$ to $87\,\%$, while full-scale effective dimensionality stays at 3.1 to 4.2. DINO shows slightly higher effective dimensionality and retention on the harder datasets than MoCo.

Prithvi-EO-2.0 (600M). At minimum compute, Prithvi-EO-2.0 (600M) (Figure 11f) retains $81\,\%$ to $103\,\%$ with effective dimensionality between 2.4 and 4.7, comparable to SSL4EO. The trajectory is strongly non-monotonic: effective dimensionality peaks at 7.7 to 10.3 around $40\,\%$ to $50\,\%$ compute and decreases to 2.8 to 8.1 at full scale, mirroring the non-monotonic EVR slope (Figure 6). Across this range, retention varies by less than 15 percentage points while effective dimensionality differs by more than $2.5\times$.

TerraMind-1.0. TerraMind-1.0 base and large (Figures 11c and 11d) start at slightly higher minimum-compute effective dimensionality than DOFA (3.0 to 3.7 for base, 3.1 to 3.6 for large), with retention at $1\,\%$ compute of $75\,\%$ to $92\,\%$ (base) and $75\,\%$ to $89\,\%$ (large), lower on harder datasets. Full-scale effective dimensionality reaches 4.2 to 6.6 (base) and 4.3 to 6.3 (large), an increase of roughly 1.5–$2.2\times$ from minimum to full scale. At high compute, effective dimensionality typically peaks at intermediate scales before plateauing or decreasing.

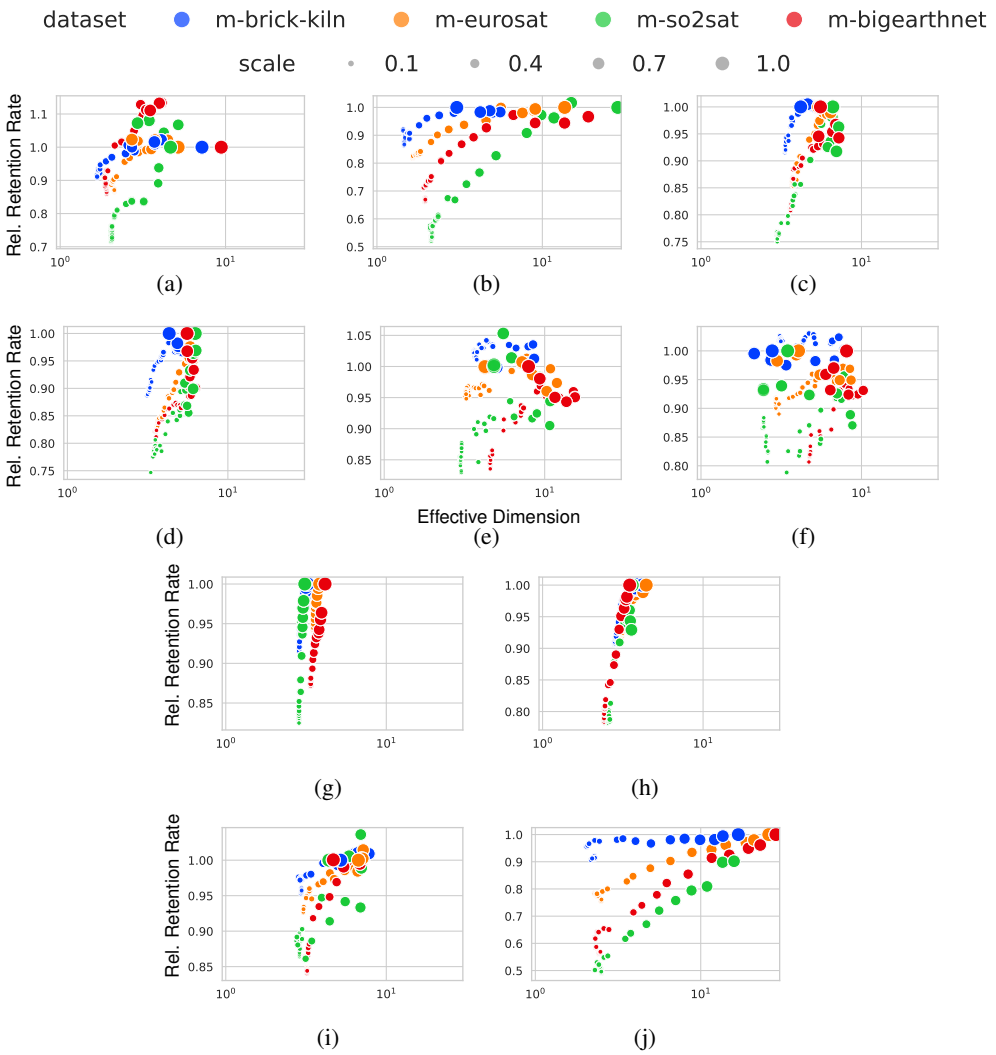


Figure 11: Effective Dimension of the test split of four classification datasets from the geobench benchmark collection with respect to their relative retention rate on pretrained RS FMs: a) DOFA (base); b) DOFA (large); c) TerraMind-1.0 (base); d) TerraMind-1.0 (large); e) Prithvi-EO-2.0 (300M); f) Prithvi-EO-2.0 (600M); g) SSL4EO ViT-S (Dino); h) SSL4EO ViT-S (MoCo); i) CopernicusFM; and j) Panopticon.

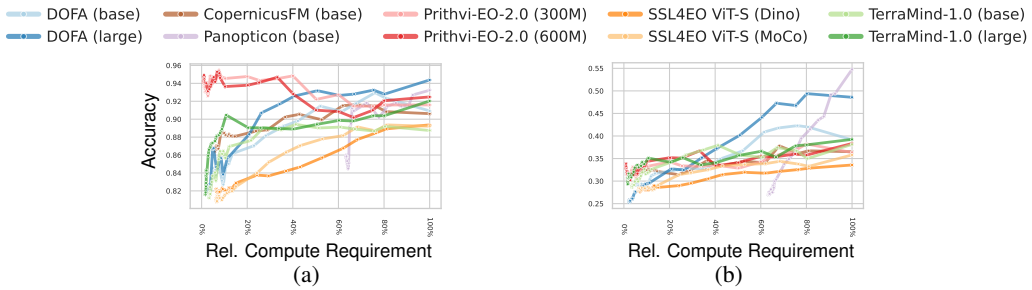


Figure 12: k-NN classification performance with respect to their relative compute requirements of ten pretrained RS FMs on: a) m-brick-kiln; and b) m-so2sat.

Copernicus-FM. Copernicus-FM (Figure 11i) closely resembles TerraMind-1.0 base: at the smallest scale, effective dimensionality is 2.7 to 3.3 with a retention of 84 % to 98 %, lower on the harder datasets, and it increases by a factor of about 1.5–2.3× to 4.5 to 7 at full scale.

Panopticon. Panopticon (Figure 11j) behaves similarly to DOFA (large): at the smallest scale, effective dimensionality is low (2.2 to 2.7) with a retention of 49 % to 91 %, lowest on m-so2sat and highest on m-brick-kiln. Effective dimensionality grows by more than an order of magnitude toward full scale, accompanied by retention gains of up to about 45 percentage points. Note that Panopticon's compute never falls below roughly 60 % because its patch embedding is not slimmed (Section A), so its smallest scale does not correspond to the minimum compute of the other models.

Summary. The relation between effective dimensionality and retention differs across models: DOFA and Panopticon show a large dimensional expansion alongside sizeable retention gains, whereas Prithvi-EO-2.0 and TerraMind-1.0 retain performance across scales with markedly different effective dimensionality. The non-monotonic pattern shared by Prithvi-EO-2.0 and TerraMind-1.0 (and absent in SSL4EO) is not explained by these analyses; incomplete convergence during pretraining is one untested possibility.

H Additional Experiments on Post-Hoc Slimmability

H.1 Additional Experiments on Classification

m-brick-kiln. As shown in Figure 12a, all RS FMs exhibit high retention at minimum compute, consistent with the low task complexity of binary scene classification. Both Prithvi-EO-2.0 models retain more than full-scale performance across the majority of the compute range, with Prithvi-EO-2.0 (300M) reaching 102 % and Prithvi-EO-2.0 (600M) reaching 103 % relative retention even at minimum compute budget. Performance remains above full scale through approximately 40 % compute before converging. TerraMind-1.0 (base) similarly exceeds full-scale performance at intermediate scales (27 % to 80 % compute), while TerraMind-1.0 (large) reaches full scale without surpassing it, apart from a small dip between 10 % to 40 % compute. DOFA (base) retains 92 % at minimum compute and exceeds full scale by up to 2 % at intermediate scales before converging; DOFA (large) retains 87 % at minimum compute and increases monotonically. CopernicusFM retains 97 % at minimum compute and exceeds full scale by up to 1 % between 60 % to 80 % compute. Both SSL4EO models reach their minimum compute at approximately 6 % of full budget due to their smaller architecture, retaining about 91 % and increasing monotonically to full scale. Panopticon, whose relative compute does not fall below roughly 60 % (see Appendix A), retains 91 % at its smallest scale and increases to full scale. The non-monotonic behavior, where intermediate-scale models outperform full-scale models, is consistent with the pattern observed on m-eurosat.

m-so2sat. As shown in Figure 12b, in this 17-class single-label classification task, retention at minimum compute is substantially lower than on m-brick-kiln, reflecting the higher task complexity. DOFA (base) retains 75 % at minimum compute and exhibits pronounced non-monotonic behavior, with performance at 76 % and 81 % compute exceeding full scale by 8 % and 7 % respectively before falling back at full scale. DOFA (large) shows the lowest minimum-compute retention of any model

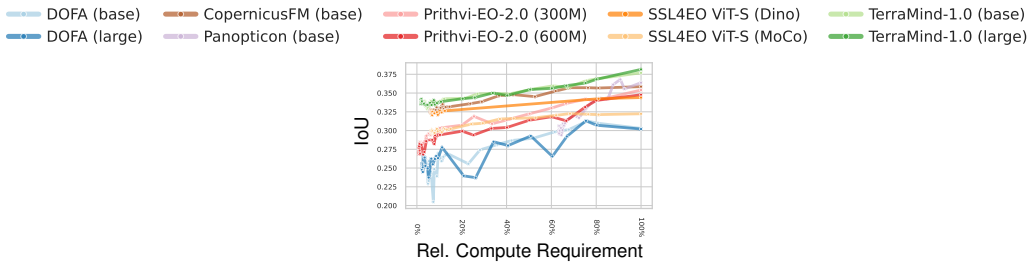
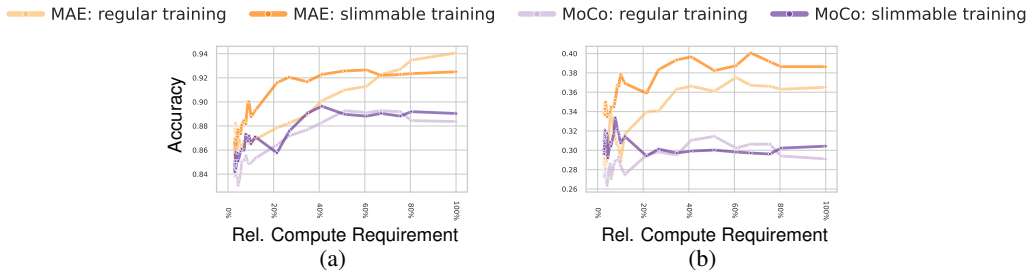


Figure 13: IoU with respect to relative compute requirements of ten pretrained RS FMs on m-SA-crop-type (10-class semantic segmentation).

across all evaluated tasks (except for Panopticon, excluded due to embedding layer differences as noted in Appendix A), retaining only 53 % at minimum compute, with gradual recovery to full scale. This is consistent with the pronounced non-monotonic peak on m-bigearthnet observed in Section 4.1 and suggests that DOFA (large) is particularly sensitive to width reduction on complex multi-label tasks. Prithvi-EO-2.0 (300M) retains 85 % at minimum compute and peaks at 5 % above full scale near 80 % compute before converging. Prithvi-EO-2.0 (600M) retains 87 % at minimum compute but exhibits an irregular trajectory across the compute range, with retention varying between 79 % and 96 % at intermediate scales before reaching full scale. Both SSL4EO models retain 78 % to 83 % at their minimum compute budget and increase monotonically, with MoCo showing a slight non-monotonic dip near 81 % compute. Both TerraMind models retain about 75 % at minimum compute, with TerraMind-1.0 (base) peaking near full-scale performance at 41 % compute before a non-monotonic dip at 50 % to 61 % compute and recovery to full scale. Across all models, the critical compute threshold below which performance drops sharply is higher on m-so2sat than on any classification task reported in Section 4.1, indicating that task complexity is the primary determinant of the effective lower bound on compute budget.

H.2 Additional Experiment on Segmentation

m-SA-crop-type. On this 10-class semantic segmentation task, slimmability is robust across all models, with retention at minimum compute ranging from 78 % to 94 %. Both SSL4EO ViT-S models exhibit the highest retention at minimum compute of all evaluated models on this task, with the DINO model retaining 94 % and the MoCo model retaining 93 % at approximately 6 % compute. Both curves are remarkably flat across the full compute range, with DINO retention varying by less than 2 % between minimum compute and 81 % compute before converging to full scale, and MoCo remaining similarly stable before marginally exceeding full scale by 0.1 % at approximately 68 % compute. TerraMind-1.0 models also exhibit stable trajectories. TerraMind-1.0 (base) retains 89 % at minimum compute with a gradual, monotonically increasing curve through full scale. TerraMind-1.0 (large) retains 88 % at minimum compute and shows the flattest curve of the large-scale models, increasing gradually and monotonically to full scale without non-monotonic deviations. CopernicusFM retains 91 % at minimum compute and increases gradually and almost monotonically to full scale. Panopticon retains 84 % at its smallest scale and slightly exceeds full scale by about 1 % near 90 % compute. DOFA (base) retains 82 % at minimum compute but exhibits the most irregular trajectory of any model on this task, including a pronounced dip to 68 % retention near 7 % compute followed by recovery to 89 % at 10 % compute. Both DOFA models exceed full scale by 2 % to 4 % at approximately 77 % compute, consistent with the non-monotonic pattern observed across other tasks. DOFA (large) retains 84 % at minimum compute with a similarly irregular mid-range trajectory, including a secondary dip to 79 % near 26 % compute. Both Prithvi-EO-2.0 models retain 78 % to 79 % at minimum compute and show generally increasing but irregular trajectories across the full compute range, never exceeding full-scale performance. Prithvi-EO-2.0 (300M) shows a notable non-monotonic dip at 33 % compute after peaking at 25 % compute (90 % retention), while Prithvi-EO-2.0 (600M) varies irregularly between 77 % and 85 % retention through the moderate compute regime before converging. The task-complexity-dependent pattern holds: minimum-compute retention on m-SA-crop-type is lower than on the simpler m-cashew-plant task reported in Section 4.1, consistent with the 10-class label structure requiring greater representational capacity at small widths.



(a) (b)

Figure 14: k-NN classification performance with respect to their relative compute requirements of a MAE-based and a MoCo-based trained model from scratch, each once with and once without slimmable training on: a) m-brick-kiln; and b) m-so2sat.

I Additional Experiments on Learned vs Post-Hoc Slimmability

We provide additional results for the remaining two datasets (m-brick-kiln and m-so2sat) to complement the main analysis of learned versus post-hoc slimmability. These additional evaluations further validate the training paradigm-dependent patterns observed on m-eurosat and m-bigearthnet, while revealing nuanced task-complexity effects.

Binary and fine-grained classification results. For the binary m-brick-kiln dataset (Figure 14a), slimmable MoCo achieves marginal improvements at low computational budgets (0.842 vs 0.838 accuracy at $3\,\%$ compute) and peaks at 0.896 at $41\,\%$ compute compared to 0.893 for regular training at $51\,\%$ compute. Both trained models converge near 0.89 at moderate scales, confirming that learned slimmability maintains performance parity at full capacity while improving low-scale robustness. For the learned slimmable MAE, the pattern shows disadvantages at very low scales (0.869 vs 0.878 at $3\,\%$) that inverse at intermediate budgets (0.916 vs 0.879 at $21\,\%$), though regular MAE ultimately dominates at maximum scale (0.940 vs 0.925).

The fine-grained m-so2sat dataset (Figure 14b) reveals more pronounced differences. Slimmable MoCo substantially outperforms regular training, achieving 0.296 accuracy at $3\,\%$ compute versus 0.280 and reaching a peak of 0.334 at $8\,\%$ compute, thereby surpassing regular MoCo's maximum of 0.314 at $51\,\%$ compute by $6.4\,\%$ relative accuracy while requiring less than $20\,\%$ of the compute. Remarkably, the learned slimmable MAE completely reverses the degradation pattern observed on m-bigearthnet, dominating across all evaluated scales with 0.340 accuracy at $3\,\%$ compute versus 0.292 for regular MAE, maintaining this advantage through full scale (0.386 vs 0.365).

Unclear relation betwee task complexity and MAE slimmability. These additional results establish no clear relationship between task complexity and MAE slimmability performance. Learned slimmable MAE exhibits severe degradation on the complex multi-label m-bigearthnet task, marginal disadvantages on the binary m-brick-kiln task, underperformance on the intermediate m-eurosat task, but strong advantages on the fine-grained single-label m-so2sat task. The benefit of learned slimmability for MAE does not follow a clear pattern across tasks. It is strongly positive on m-so2sat, mixed on m-brick-kiln, negative on m-eurosat, and severely negative on m-bigearthnet. We do not have an explanation for this variation, and it does not track class count monotonically. In contrast, MoCo's instance discrimination objective demonstrates consistent slimmability improvements across all task complexities, confirming that contrastive learning and slimmable training can be successfully combined to develop FMs for flexible deployment across diverse downstream applications.

J Feature Correlation at different Model Scales

We provide an extended feature correlation analysis across ten RS FMs and four datasets in Figure 15. The analysis reveals that distinct correlation shapes remain consistent across both model sizes and datasets, indicating that these patterns are fundamental characteristics of training paradigms rather than task-specific adaptations.

The DOFA models (Figures 15a and 15b) exhibit a characteristic correlation shape across all datasets: highly organized feature correlations at small scales transitioning to near-independence at full

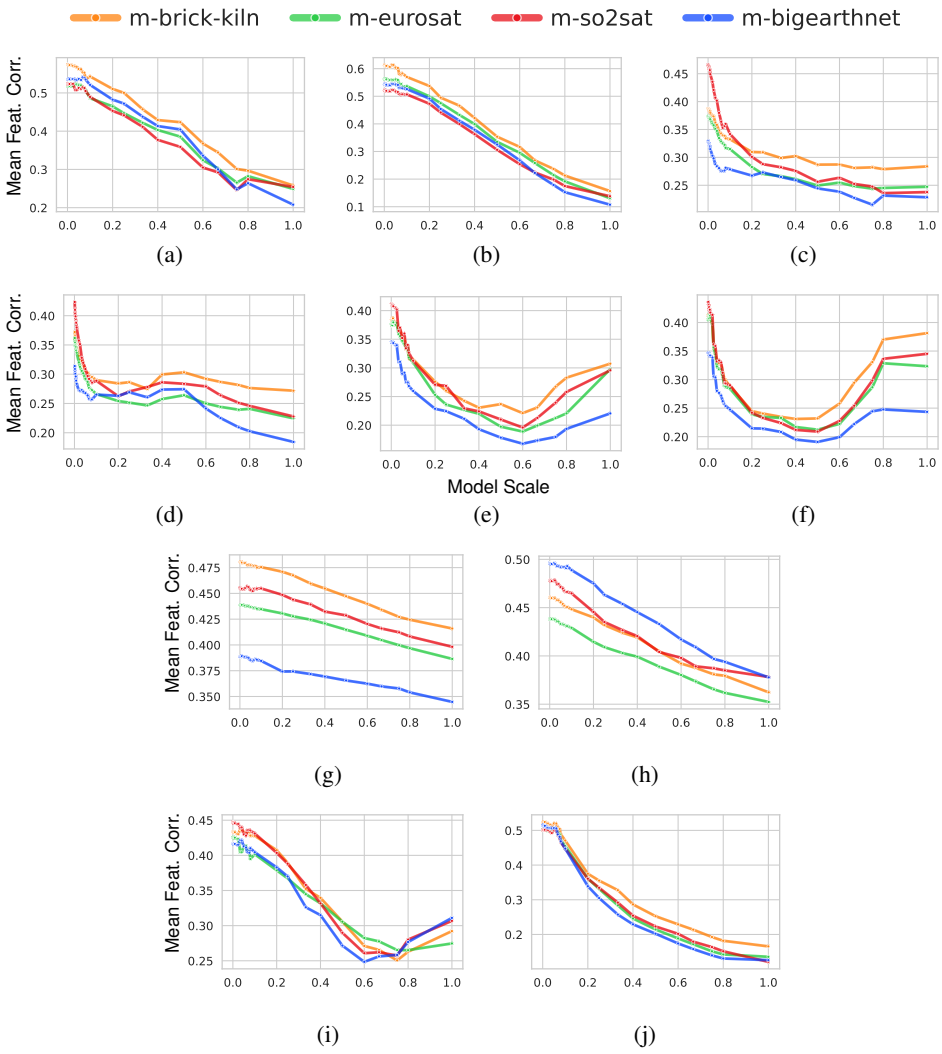


Figure 15: Mean Feature Correlation of the test split of four classification datasets from the geobench benchmark collection with respect to the model scale on pretrained RS FMs: a) DOFA (base); b) DOFA (large); c) TerraMind-1.0 (base); d) TerraMind-1.0 (large); e) Prithvi-EO-2.0 (300M); f) Prithvi-EO-2.0 (600M); g) SSL4EO ViT-S (Dino); h) SSL4EO ViT-S (MoCo). i) CopernicusFM; and j) Panopticon.
Figure partially overlaps with Figure 4, but we visualize all data for ease of comparison.

scale. Notably, the DOFA (base) model shows less correlated features at small scales but more at large scales compared to the DOFA (large), resulting in a less steep transition slope. Despite this difference in slope steepness, both models share the same fundamental correlation shape that remains consistent across binary (m-brick-kiln), intermediate (m-eurosat), fine-grained (m-so2sat), and multi-label (m-bigearthnet) tasks. This consistent pattern, combined with largely monotonic performance improvement across scales, confirms that additional uncorrelated dimensions at full scale contribute useful orthogonal information despite the correlation collapse.

The TerraMind-1.0 models (Figures 15c and 15d) demonstrate identical correlation shapes across both base and large sizes, characterized by smooth and mostly monotonic decrease from moderate to low correlations. The TerraMind-1.0 (base) model maintains higher correlation values across all scales compared to TerraMind-1.0 (large), and exhibits a slightly smoother slope, but the fundamental shape remains unchanged. This shape consistency across model sizes and datasets indicates that TerraMind-1.0's training paradigm induces a stable feature organization strategy where core discriminative features are efficiently encoded in early dimensions, with larger models simply distributing this information more independently.

Prithvi-EO-2.0 models (Figures 15e and 15f) exhibit the distinctive non-monotonic U-shaped correlation pattern across both Prithvi-EO-2.0 (300M) and Prithvi-EO-2.0 (600M) and all datasets. While the Prithvi-EO-2.0 (300M) model shows a less pronounced rebound at large scales, not reaching as high correlations as the Prithvi-EO-2.0 (600M) variant, the U-shaped character remains clearly visible. This consistent U-shape, characterized by moderate initial correlations, drops at low-to-intermediate scales, and subsequent rebounds at full scale, suggests that Prithvi's temporal training objective may induce systematic scale-dependent feature reorganization regardless of model capacity or downstream task complexity.

The SSL4EO ViT-S models (Figures 15g and 15h) show a roughly linear decrease in mean feature correlation across scales, but the total reduction is much smaller than for the other models. This is consistent with their limited slimmability, as there is little redundancy to remove when correlations stay high at small scales. Two factors may contribute: the smaller overall model size and the smaller embedding dimension. However, these models cannot be compared directly to the families above, as they are not different scales of the same training method, architecture, and dataset. The shape is also consistent across all four datasets and both models.

CopernicusFM (Figure 15i) reaches a minimum in mean correlation around scale 0.6. This coincides with its slimming profile, where downstream performance on some datasets is slightly higher at similar scales.

Panopticon (Figure 15j) shows a smooth, strictly declining curve, which fits its slimming profile. We would like to note, however, that due to its large patch embedding layer, the curve is not strictly comparable to those of the other models (see Appendix A).

The consistency of these correlation shapes across model sizes and datasets suggests that, within the three model families (DOFA, TerraMind, and Prithvi), each develops a characteristic feature organization that is stable across downstream tasks. In details, DOFA's multi-modal training coincides with high-to-low correlation transitions, TerraMind-1.0's land cover focus with a smooth, mostly monotonic decrease, and Prithvi-EO-2.0's temporal modeling with a non-monotonic U-shaped pattern. The remaining models do not allow a family comparison, but they also keep their correlation shape across all four datasets.

K Additional Experiments on Explained Variance

We provide an extended EVR analysis across ten RS FMs and four datasets in Figure 16. The extended analysis confirms the log-log slope patterns observed in Figure 6 while revealing model family-specific consistency across datasets and scales. Both Prithvi-EO-2.0 and both TerraMind-1.0 models exhibit remarkable internal consistency: within each model family, both sizes demonstrate nearly identical EVR decay patterns across all datasets and scales. This consistency indicates that these models' variance distribution strategies are influenced by their training objectives. The Prithvi-EO-2.0 models maintain their characteristic non-monotonic pattern (becoming more isotropic at intermediate scales before reconsolidating at full scale) across all datasets, while the TerraMind-1.0 models preserve stable intermediate anisotropy regardless of whether evaluated on binary, multi-class,

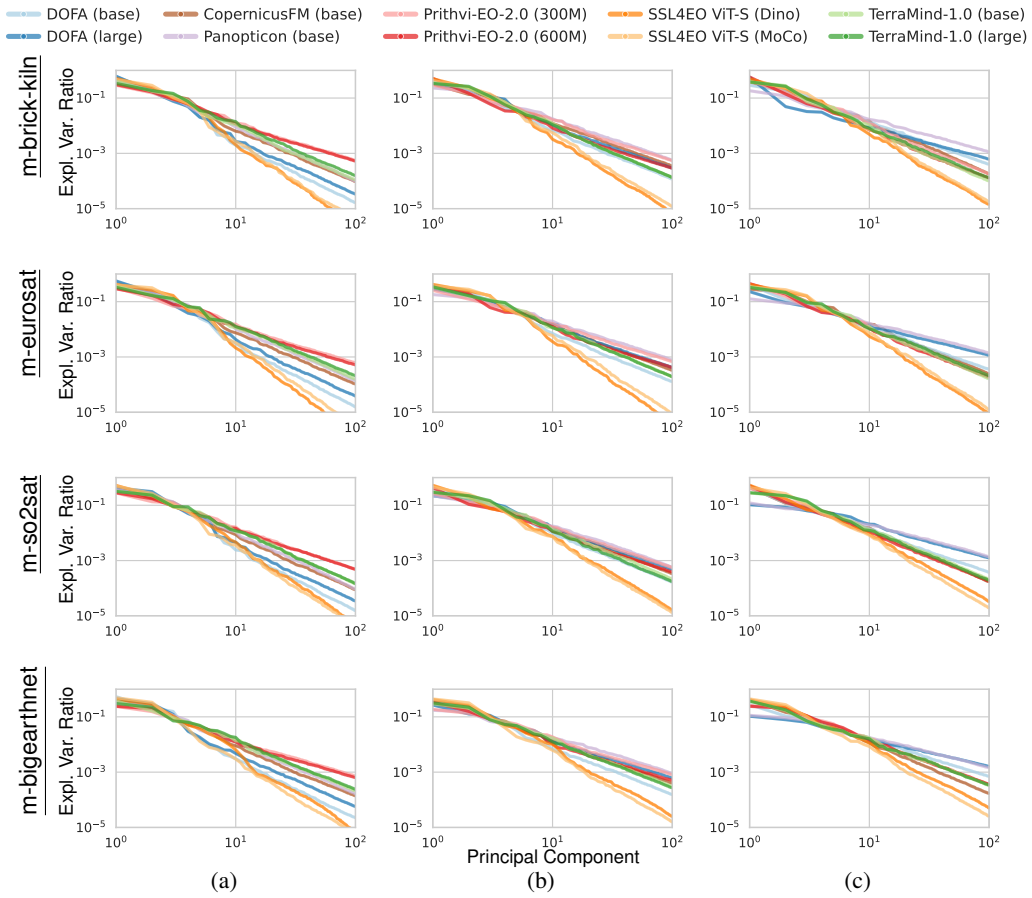


Figure 16: EVR with respect to the singular value rank index for ten RS FMs on four classification datasets from the geobench benchmark collection at three relative compute requirements: a) at 0.33; b) at 0.67; and c) 1.0 on (from top to bottom): m-brick-kiln; m-eurosat; m-so2sat; and m-bigearthnet.

or multi-label tasks. In contrast, the DOFA models exhibit systematic scale-dependent behavior that varies between base and large variants but remains consistent across datasets. Both DOFA models demonstrate steeper slopes (more concentrated variance) at smaller scales and shallower slopes (more distributed variance) at larger scales across all four datasets, with the base model maintaining systematically steeper slopes than the large variant at equivalent relative compute budgets. This pattern confirms the observation that DOFA's multi-modal training drives monotonic variance redistribution as capacity increases, with larger models distributing variance more evenly. The consistency of this pattern across binary (m-brick-kiln), intermediate (m-eurosat), fine-grained (m-so2sat), and multi-label (m-bigearthnet) tasks demonstrates that DOFA's scaling strategy is an architectural property rather than a task-specific adaptation. The four additional models (SSL4EO ViT-S DINO, SSL4EO ViT-S MoCo, CopernicusFM, and Panopticon) do not belong to a model family, so within-family consistency cannot be assessed. They nonetheless show stable EVR decay patterns across all four datasets, indicating that their variance organization is largely independent of the downstream task.

These findings reinforce the intuition that training objectives may influence how models organize information across scales, with this organization remaining stable across diverse downstream task requirements.

L Failure Modes and Edge Cases

Class-wise average precision of m-bigearthnet To check whether aggregate mAP_M hides uneven degradation, we report the class-wise average precision (AP) of the best and worst class of the full-width model at each scale during k-NN evaluation in Figure 17. The easiest classes keep most

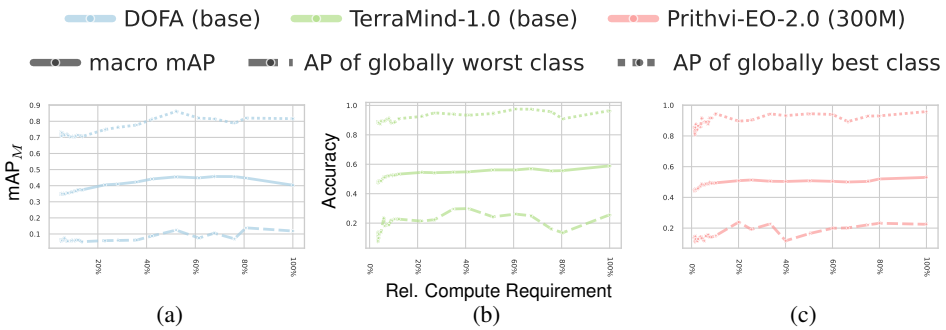


Figure 17: KNN classification performance with respect to relative compute requirements on the test split of m-bigearthnet evaluating the global best, worst, and macro-mean averaged precision in multi-label classification of: a) DOFA (base); b) TerraMind-1.0 (base); and c) Prithvi-EO-2.0 (300M).

of their AP at 1 % compute (89 % to 92 % of full-width AP across DOFA, Prithvi-EO-2.0, and TerraMind). The hardest classes lose a larger share: Prithvi and TerraMind retain roughly 50 % to 60 % of their full-width AP, while DOFA's hardest class is already very low (0.11) at full width. Slimming therefore degrades classes unevenly, and aggregate metrics partly conceal this, but hard classes lose accuracy gradually rather than collapsing.

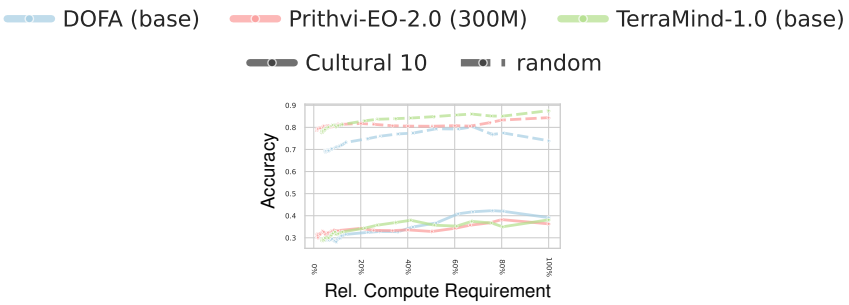


Figure 18: KNN classification performance with respect to relative compute requirements on m-so2sat under random and predefined disjoint (Cultural 10) geographic shifts of: DOFA (base); TerraMind-1.0 (base); and Prithvi-EO-2.0 (300M).

Geographical shift in m-so2sat To quantify the effect of geographic shift, we compare the geographically disjoint Cultural-10 split of the geobench benchmark collection against a random split. The random split pools train and test data, reshuffles, resamples to identical sizes, and averages five draws, using the same k-NN protocol as used in all other experiments. As shown in Figure 18, retention at minimum compute is higher on the random split (about 93 % for DOFA (base), accuracy drops from 74 % to 69 %) than on Cultural-10 (about 77 %, accuracy drops from 39 % to 30 %), with the same pattern for Prithvi-EO-2.0 (300M) and TerraMind-1.0 (base). Geographic shift thus reduces the measured redundancy, but substantial redundancy remains under geographically disjoint evaluation.

Class imbalance To test whether slimming interacts with class imbalance, we impose an exponential imbalance on the training set of m-eurosat: class c keeps a fraction $r^{c/9}$ of its samples, so class 0 keeps all samples and class 9 keeps the imbalance ratio r, with r between 0.01 and 1.0 and the classes in between decaying geometrically. The result is averaged over five subsampling seeds. The result of this subsampling and the following k-NN evaluations are shown in Figure 19. Absolute accuracy drops substantially at every width as imbalance grows. The retention curves stay roughly parallel across imbalance ratios for DOFA (base) and Prithvi-EO-2.0 (300M). We therefore conclude that slimming does not interact strongly with imbalance for these models. TerraMind (base) shows a mild interaction at the most extreme imbalance ratio (0.01), where retention at 1 % compute falls to about 77 % from 86 % when balanced.



Figure 19: KNN classification performance with respect to relative compute requirements on the test split of m-eurosat evaluating a synthetic class imbalance in the training set of: a) DOFA (base); b) TerraMind-1.0 (base); and c) Prithvi-EO-2.0 (300M).

M Computational Resources

All SSL pretraining experiments were conducted on $4\times$NVIDIA A100 GPU or $4\times$NVIDIA H200 GPU. Regular pretraining runs required approximately one day; slimmable pretraining runs required approximately four days due to the multi-width training protocol. Post-hoc slimmability evaluation, feature extraction, and representational analyses (k-NN classification, EVR, feature correlation, effective dimensionality) can be performed on CPU. All such experiments were run using 16 CPU cores and 128 GB RAM. Feature extraction across all compute scales can be completed in under 30 minutes per model; GPU acceleration is beneficial for this step and was used where available (NVIDIA H200, A100, and A40 GPUs). Downstream evaluation runtimes vary by task type.

k-NN classification and change detection complete within a few hours per model. Linear probing requires moderately longer due to the additional training loop. Semantic segmentation is the most compute-intensive downstream task, requiring up to 8 hours per model due to dataset size and the higher complexity of dense prediction relative to k-NN evaluation. The reported runtimes reflect the final experimental runs; total compute including preliminary and repeated experiments is not tracked.

N Licenses and Assets

The pretraining data used for learned slimmability experiments is derived from a subsampled subset of Major-TOM Core-S2L2A [10], available at `https://huggingface.co/datasets/Major-TOM/Core-S2L2A` under the Creative Commons Attribution-ShareAlike 4.0 International license (CC-BY-SA 4.0). Downstream evaluation datasets are sourced from the geobench benchmark collection [24], which is released under the Apache 2.0 license (`https://github.com/ServiceNow/geo-bench`). All pretrained RS FMs evaluated in this work are accessed via TerraTorch [11], which is released under the Apache 2.0 license (`https://github.com/IBM/terratorch`).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract and introduction are in line with the contributions and scope of the paper. Additionally, a paragraph on scope limitations is included in the conclusion.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: A paragraph on limitations is included in the conclusion section.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All relevant information to reproduce the results are provided. Additionally, the code for full reproduction is available at `https://github.com/hackelle/slimmable-remote-sensing-foundation-models` under MIT license.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code required to reproduce training and evaluation results is available at `https://github.com/hackelle/slimmable-remote-sensing-foundation-models` under MIT license.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: The experimental setup follow standard training splits and parameters, which are detailed in the respective sections.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Error bars show 2-sigma across random seeds and are provided for all linear probe and finetuning classification evaluations; pretraining and segmentation experiments are excluded due to computational cost and k-NN results are deterministic. Error bars may be smaller than the line width in some figures.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: A section detailing the computational resources used is included in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conforms with the NeurIPS Code of Ethics. We note that m-bigearthnet results rely on the geobench benchmark, which uses BigEarthNet v1.0 (43 labels), a superseded dataset version; this is a limitation of the benchmark dependency, not an ethical violation. Broader societal risks associated with AI applied to remote sensing data (e.g., surveillance) are discussed in the paper.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: The paper contains a section on practical implications as well as an impact statement.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: There are no models or datasets released in this work.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: A section detailing the licenses of all datasets and models used is included in the appendix. All used datasets and models are cited with the appropriate specifications.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: All code required to reproduce training and evaluation results is publicly available; no model weights or datasets will be released.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.