A Appendix

We *anonymously* share our pretrained models and preprocessed datasets:
`https://zenodo.org/record/7954787`

A.1 Other Pretraining Experiments

In this section, we experiment with additional CROMA settings. We use the same experimental conditions as §5.1 of our paper; i.e., we linear probe representations on BigEarthNet [126] (reporting mAP on the combined validation and test sets) and patch encodings on DW-Expert-120 [138] (reporting mIoU on the validation set). We use the linear probing hyper-parameters listed in §A.3.2 of this Appendix.

Table 7: Linear probing results on radar-only ("R"), optical-only ("O"), and joint radar-optical ("RO") inputs. Across all experiments we use 2D-ALiBi with X-ALiBi, 75% shared masking, ViT-B backbones, and 100 pretraining epochs.

Cross-Modal	Cross-Modal	Decoder	Obj. Weights	HN Mixing		Classification (mAP)			Segmentation (mIoU)		
Image Obj.	Patch Obj.	Depth, Dim	$\lambda_{Con}, \lambda_{\text{MAE}}$	$(1024, 0, n)$	Cost	R	O	RO	R	O	RO
InfoNCE	MSE	1, 512	1, 1	✗	1×	77.4	83.9	84.3	40.5	56.0	56.7
InfoNCE	MSE	1, 768	1, 1	✗	1×	77.4	83.9	84.3	40.7	56.1	56.6
InfoNCE	MSE	3, 512	1, 1	✗	1.2×	77.5	83.9	84.4	40.5	56.3	57.1
InfoNCE	MSE	3, 768	1, 1	✗	1.3×	77.5	83.9	84.4	40.7	56.2	56.6
InfoNCE	MSE	6, 512	1, 1	✗	1.4×	77.5	83.9	84.4	40.3	56.0	56.7
InfoNCE	MSE	6, 768	1, 1	✗	1.6×	77.6	83.8	84.5	40.6	56.2	56.7
InfoNCE	✗	1, 512	1, 1	✗	1×	77.4	84.0	84.5	40.8	56.1	56.4
InfoNCE	✗	1, 768	1, 1	✗	1×	77.5	84.2	84.5	40.8	56.1	56.2
InfoNCE	✗	3, 512	1, 1	✗	1.2×	77.6	84.1	84.5	40.8	56.2	56.7
InfoNCE	✗	3, 768	1, 1	✗	1.3×	77.0	83.9	84.5	40.6	56.1	56.5
InfoNCE	✗	6, 512	1, 1	✗	1.4×	77.3	84.1	84.5	40.8	56.1	56.5
InfoNCE	✗	6, 768	1, 1	✗	1.6×	77.5	84.1	84.6	40.6	56.5	56.8
InfoNCE	InfoNCE	3, 512	1, 1	✗	2.2×	72.8	80.9	82.4	39.0	55.1	55.2
InfoNCE	✗	1, 512	1, 2	✗	1×	77.5	84.3	84.2	40.7	55.9	56.2
InfoNCE	✗	1, 512	1, 4	✗	1×	77.5	84.3	84.1	40.6	55.4	56.0
InfoNCE	✗	1, 512	2, 1	✗	1×	77.5	84.1	84.5	40.4	55.9	56.3
InfoNCE	✗	1, 512	4, 1	✗	1×	77.6	83.9	84.5	40.7	55.8	56.8
InfoNCE	✗	1, 512	1, 1	128	1×	73.6	81.6	83.0	38.0	53.2	55.0
InfoNCE	✗	1, 512	1, 1	256	1×	73.0	81.0	82.8	37.8	52.9	54.7
InfoNCE	✗	1, 512	1, 1	512	1×	72.5	80.2	82.4	37.6	52.6	54.4
VICReg	MSE	1, 768	1, 1	✗	1.1×	70.7	78.7	83.3	40.0	55.5	55.1

Self-supervised Objectives. Inspired by the local objective of VICRegL [143], we experiment with a mean squared error (MSE) objective between cross-modal patch encodings, i.e., $\mathcal{L}_{local}$=MSE$(\mathcal{E}_{\text{R}}, \mathcal{E}_{\text{O}})$. This attracts patch encodings if they match locations, i.e. if they represent the same 80 m × 80 m square on the ground. We find this does not improve representations. Next, we experiment with the VICReg [142] objective (calculating VICReg statistics based on a batch size of 800) between cross-modal image representations, i.e., $\mathcal{R}_{\text{R}}$ and $\mathcal{R}_{\text{O}}$; we find it underperforms InfoNCE [28]. Finally, we experiment with the InfoNCE objective between cross-modal patch encodings; positive pairs are encodings that match locations across modalities, and negative pairs are all other encodings from the matched sample and encodings from all other samples in the batch. This does not improve representations and slows pretraining by 2.2× (Table 7).

Objective Weights. We find that weighting the contrastive loss term or MAE [31] loss term does not uniformly improve representations; hence, we select equal weights.

Hard Negatives. We find that hard-negative mixing [145] (N=1024, s=0, s'=n, β=0.5, with n of 128, 256, or 512) degrades performance when used in our framework. We leave altering the contrastive learning objective to future work, for instance, other hard negative settings or nearest-neighbor contrastive learning.

Decoder Sizes. At least in these experiments, CROMA is not sensitive to the decoder size; a tiny decoder with a 1-layer, 512-d transformer performs similarly to a much larger 6-layer, 768-d transformer.

Position Encoding with Shared Masking. We find that using 2D-sinusoidal embeddings or PEG [118] with *shared* masking performs poorly. These two methods of position encoding store positional information in the internal representations, which can help solve the contrastive objective if both modalities share masks; 2D-ALiBi instead stores positional information in the attention matrix, which may prevent this from occurring. In our paper (Table 5), we show that 2D-sinusoidal or PEG can perform well in our framework if modalities are masked independently; although 2D-ALiBi still outperforms these approaches.

Table 8: Linear probing results with *shared* 75% masking, ViT-B, 100 epochs.

Method	Classification mAP			Segmentation mIoU		
	R	O	RO	R	O	RO
PEG [118]	67.9	75.9	79.0	32.6	49.8	51.0
2D-Sin.	69.4	75.6	79.8	29.0	44.1	50.7

Lower Masked Tuning. FLIP [123] performs contrastive learning using the representations of masked-out samples; after this masked pretraining, it leverages *unmasked* tuning to increase accuracy by 1.3% on zero-shot ImageNet-1K. Unmasked tuning continues FLIP pretraining by performing contrastive learning using the representations of unmasked samples to reduce the distribution gap between pretraining and inference [123]. We cannot perform fully unmasked tuning because we must mask patches for our reconstruction objective. However, we can lower our mask ratio and perform *lower* masked tuning. Following FLIP, initializing parameters with our pretrained CROMA-L model, we train for 5 additional epochs using a base learning rate of 8e-8, warmup over the first epoch, and cooldown for 4 epochs using a cosine decay schedule. We explore mask ratios $\{10\%, 25\%, 50\%\}$ and find that lower masked tuning does not improve linear probing accuracy for CROMA.

Table 9: Lower masked tuning for 5 epochs after pretraining CROMA-L.

Mask Ratio	Classification mAP			Segmentation mIoU		
	R	O	RO	R	O	RO
10%	80.8	84.7	84.7	43.8	56.8	56.6
25%	80.8	84.7	84.8	43.9	56.8	56.6
50%	80.8	84.8	85.0	43.9	56.8	56.6

A.2 Pretraining Details

A.2.1 Data

We use the SSL4EO dataset [85], which consists of Sentinel-1 & 2 imagery acquired at 250K locations around the world; each location (a 2.64 km × 2.64 km square) is imaged four times, spread out over a year. We use these 1M samples of 264 × 264 pixels for pretraining. Please see the SSL4EO paper [85] for more details.

A.2.2 Implementation

We use an NVIDIA DGX server (8×A100-80GB), the maximum batch size that can fit into 640 GB of VRAM (7,200 for our default ViT-B), bfloat16 precision, a base learning rate of 4e-6, warmup for 5% of the total epochs, and cooldown via a cosine decay schedule. We use the same normalization procedure as SatMAE [26]. For data augmentation, we randomly crop 60-180 pixel squares from the original 264 × 264 pixels and resize the crops to 120 × 120 pixels (our default image size). We also perform vertical and horizontal flipping, 90-degree rotations, and mixup=0.3. Crucially, we apply these transformations identically to both modalities; if we applied them to each modality independently, our spatial alignment would break. We use the AdamW optimizer with β_1=0.9, β_2=0.999, and a weight decay of 0.01.

A.3 Evaluation Details

The evaluation of foundation models for Earth Observation is less mature than in other fields. We do our best to re-use the experimental conditions of the SoTA, i.e., SatMAE [26], and improve upon them where possible. One such condition is to report results from a held-out validation set; precisely, the best validation performance measured after each finetuning epoch is reported. No test sets are used. To enable fair comparisons with prior work, we copy this approach. In trying to improve the evaluation of foundation models for Earth Observation, we detail our approach in this Appendix, share code and preprocessed datasets, re-evaluate all near-SoTA models under identical conditions, and evaluate models in more ways than prior work (i.e., linear and nonlinear probing, kNN classification, and K-means clustering).

We initialize parameters from publicly shared pretrained weights, evaluating all models ourselves under identical conditions. Although this process is laborious, we believe it significantly improves the value of our paper; several prior studies have often evaluated their models in different ways, using different data splits that cannot be directly compared. When downloading pretrained weights, we use the latest weights that are publicly available. For instance, SatMAE [26] released improved versions of their multispectral ViT-B and ViT-L models, pretrained for 200 epochs, after their manuscript was accepted for publication (edited on *arxiv* on January 15^{th}, 2023). We exclusively evaluate these improved models throughout our paper, ensuring we compare CROMA to the best models available.

A.3.1 Data

BigEarthNet. [126] We use the same splits for training (10% of the complete training set) and evaluating (the entire validation set) as SatMAE [26] and SeCo [25]. However, we use the combined validation and test sets (236,130 samples) in our ablation studies to increase the reliability of our findings with minimal added cost. Images are 120×120 pixels.

fMoW-Sentinel. [26] Inspired by how the BigEarthNet benchmark is used (i.e., training on 10% of the complete training set of 354,200 samples), we create a 10% split of the complete fMoW-Sentinel training set of 712,874 samples. We share the IDs of the 10% of fMoW-Sentinel training samples that we randomly selected. We believe this smaller training set should be used in future work to reduce the costs of hyper-parameter searches—a *single* finetuning run of SatMAE on the complete training set requires 192 hours on a V100 GPU [26]. Following SatMAE, we use the full validation set for evaluation. Images vary in size, the mean height is 45 pixels, and the mean width is 60 pixels.

In our paper, we benchmark this new split. However, we report results obtained by our CROMA models on the complete training set in Table 10. Due to the costs of finetuning on the complete training set (712,874 samples), we decide to allocate our resources elsewhere and *not* perform any hyper-parameter tuning. Instead, we select hyper-parameters we believe to be reasonable and finetune CROMA-B and CROMA-L once. For finetuning, we use a base learning rate of 1e-5 and all other hyper-parameters from §A.3.2.

Table 10: fMoW-Sentinel results (top 1 accuracy) using the *complete* training set. * denotes results reported in SatMAE (updated on *arxiv* on January 15^{th}, 2023).

Method	Backbone	Finetuning	Linear Probing
SatMAE	ViT-B	62.65*	37.40
CROMA	ViT-B	61.00	40.94
SatMAE	ViT-L	63.84*	39.19
CROMA	ViT-L	63.59	41.96

EuroSAT. [127] We use the same training and validation sets as SatMAE. Images are 64×64 pixels.

Canadian Cropland. [128] We are the first to benchmark this dataset of Canadian agricultural croplands, consisting of 10 classes (barley, canola, corn, mixedwood, oats, orchard, pasture, potato, soybean, and spring wheat). We select this dataset because it is a large dataset that evaluates different capabilities from the other benchmarks that typically consider croplands as a single class. Following EuroSAT [127], the authors selected an image size of 64×64 pixels [128]; therefore, models evaluated on EuroSAT can be evaluated on Canadian Cropland with minimal modifications. We use the training set and combine their validation and test sets to form a single held-out set for evaluation. We share these complete training and validation sets. The performance (see Table 1 in our paper) and representation visualizations (see Fig. 5 and 6 in this Appendix) indicate that the 10 classes present in this dataset are challenging to separate.

DFC2020. [137] This dataset is used for evaluation in diverse ways—both the choice of data split and image size. The original dataset comprises 6,114 samples of 256×256 pixels. These samples are typically split into two; a so-called "validation set" of 986 samples and a so-called "test set" of 5,128 samples. Some studies use the "validation set" for training and the "test set" for validation; others use the "test set" for training and the "validation set" for validation. Some studies use the full 256×256 pixels as inputs to their models, while others use smaller inputs. We select the split of 5,128 samples for training, which we divide into 46,152 images of 96×96 pixels—leaving us with the split of 986 samples for validation, which we divide into 8,874 images of 96×96 pixels. We select this final resolution because it is the default image size of SatMAE, enabling a fair comparison to the SoTA. We share these complete training and validation sets.

DW-Expert. [138] The data collected by Dynamic World [138] is a new high-quality dataset annotated by experts with the help of auxiliary information. Thus, it should be used in the future when benchmarking models. Our work uses the expertly annotated data from Dynamic World, which

we split into 20,422 train samples and 51,022 validation samples. All images are 96 × 96 pixels to enable a fair comparison with SatMAE. We share these complete training and validation sets. We also create a version of this dataset that consists of 120 × 120 pixel images (i.e., DW-Expert-120) that we only use for ablations because it is the default image size of CROMA.

MARIDA. [139] We use the training set and combine the validation and test sets to form a single held-out set for evaluation. Following our approach for DFC2020 and DW-Expert, we divide the original images into images of 96 × 96 pixels. Because it is a sparsely labeled dataset (i.e., only a fraction of pixels per image are labeled), we include images with at least one labeled pixel. We select this dataset because it evaluates different capabilities from the other semantic segmentation benchmarks. It consists of the following classes: marine debris, dense *Sargassum*, sparse *Sargassum*, natural organic material, ship, clouds, marine water, sediment-laden water, foam, turbid water, shallow water, waves, cloud shadows, wakes, and mixed water. We share these complete training and validation sets.

A.3.2 Implementation

Finetuning. We select reasonable hyper-parameters that we use for all models and datasets unless otherwise stated, and sweep across learning rates. This learning rate sweep is essential to creating fair evaluation conditions across models since each model is given the same search budget (in terms of finetuning runs, not compute hours), and different models have different optimal learning rates. Models pretrained with reconstruction approaches tend to require higher base learning rates during finetuning than models pretrained with contrastive learning. For instance, MAE [31] lists a base learning rate of 1e-3, FLIP [123] lists a base learning rate of 5e-5, CoCa [49] lists base learning rates from 1e-5 to 5e-4, depending on the downstream dataset.

No single learning rate would enable a fair comparison across all models and datasets. Therefore, we sweep learning rates across an extensive range {3e-5, 5e-5, 8e-5, 1e-4, 3e-4, 5e-4, 8e-4, 1e-3} and report the best single evaluation result obtained for each dataset; this sweep is performed for CROMA models and all other models. We convert these base learning rates to actual learning rates via the widely used linear scaling rule: $lr = base\_lr \times batch\_size/256$. We use the largest batch size that can fit on an A100-40GB GPU (using bfloat16 precision), the AdamW optimizer with β_1=0.9, β_2=0.999, and a weight decay of 0.01. We warmup for 5 epochs and cooldown for 30 epochs using a cosine decay schedule (other than EuroSAT, which we cooldown for 150 epochs); this follows SatMAE [26]. For classification tasks, we use mixup=0.8, cutmix=1.0, switch probability=0.5, and label smoothing=0.1. For both classification and segmentation tasks, we perform vertical and horizontal flipping and 90-degree rotations. We enlarge images to the default image size of the model we are finetuning (i.e., the image size on which the model was pretrained), with one exception. The default image size of SatMAE is 96 × 96; however, BigEarthNet images are 120 × 120 [126], requiring that we either crop BigEarthNet samples (losing information) or adapt SatMAE to larger images. We achieve better performance by adapting SatMAE to 120 × 120 images, via the widely used position embedding interpolation algorithm, than cropping BigEarthNet samples down to 96 × 96. This allowed us to achieve an mAP of 86.18 for SatMAE, a significant improvement over the 82.62 reported in the SatMAE paper. All other datasets use images of 96 × 96, or smaller—thus, there is no reason to use this technique for other datasets.

Linear and Nonlinear Probing. We encode each image without data augmentation, then train linear and nonlinear probes on the frozen representations. Since each model only encodes each image once, we can sweep through a large range of learning rates ({1, 2, 3, 4, 5, 6, 7, 8, 9}e{-4, -3, -2}) very quickly. Unlike finetuning, we do not evaluate probes after every epoch; instead, we evaluate trained probes after all epochs are complete. We use a batch size of 1024, bfloat16 precision, the AdamW optimizer with β_1=0.9, β_2=0.999, and a weight decay of 0.01. We warmup for 5 epochs and cooldown for 100 epochs using a cosine decay schedule.

Non-parametric kNN and K-means. For kNN, we use the implementation from [27]. This consists of encoding all training and validation samples, then using the representations of validation samples as queries and training samples as keys to fetch training labels. These fetched training labels are used to classify validation samples. We use k=20, other values for k (i.e., 10, 50) ranked models in the same order as k=20. For K-means, we use the implementation from [129]. This consists of encoding all training and validation samples, then clustering training samples with K-means (K-means++ initialization run 10 times). Then, we assign validation samples to clusters, and we assign clusters to classes via the Hungarian matching algorithm.

Table 11: CROMA vs SatMAE training and inference throughput on an A100-40GB GPU.

Model	Backbone	Image Size	Train Imgs/s	Inference Imgs/s
SatMAE	ViT-B	96×96	249.3	692.5
CROMA	ViT-B	96×96	1,079.3	2,957.7
CROMA	ViT-B	120×120	555.0	1,532.1
SatMAE	ViT-L	96×96	84.2	263.2
CROMA	ViT-L	96×96	389.1	1,168.2
CROMA	ViT-L	120×120	209.6	640.3

SatMAE Specifics. SatMAE [26] divides spectral bands into three groups and outputs patch encodings for every group; thus, SatMAE outputs three patch encodings per patch location. To be as fair as possible to SatMAE, we explore four ways of merging these co-located patch encodings to perform segmentation: unnormalized spatial concatenation, normalized spatial concatenation, unnormalized spatial pooling, and normalized spatial pooling. We find unnormalized spatial concatenation (i.e., concatenating the patch encodings of co-located patches before the LayerNorm) performed best. Thus, we use the unnormalized spatially concatenated patch encodings for all segmentation datasets and methods (i.e., finetuning and probing). Conversely, CROMA does not divide spectral bands into groups—resulting in $3\times$ shorter sequence lengths. The computation required to process a sequence of tokens with a transformer increases with increasing sequence lengths. This makes CROMA much more computationally efficient than SatMAE for a given ViT backbone and image size (Table 11).

A.4 Societal Impact

Since we pretrain our models on the SSL4EO dataset [85], our models may be biased towards the distribution from which SSL4EO data were sampled. Although SSL4EO samples are geographically diverse (please see Fig. 2 from the SSL4EO paper [85]), locations are sampled from areas surrounding human settlements. As a result, large geographic areas that are sparsely populated—for instance, the Amazon rainforest, the Sahara desert, and the Australian outback—are underrepresented. This could negatively impact the quality of representations in these locations and any decisions made on their basis.

Another distribution shift—this time, between finetuning and inference—is our primary concern. For example, finetuning a model on the imagery of one geography, then making predictions on the imagery of another geography, creates a distribution shift. As a result, biases from the finetuning geography may be realized in the predictions made by the finetuned model. This is particularly problematic when these predictions are used in decision-making, for instance, allocating poverty assistance. However, it is well-demonstrated that pretrained models are more robust to distribution shifts than models trained from scratch. Additionally, as we develop better foundation models for Earth Observation, we reduce the need for annotated data; this may allow practitioners to be more selective of the data they wish to leverage during finetuning.

We do not expect our pretrained models to be particularly valuable for military applications, as militaries likely have access to higher resolutions (spatially, spectrally, and temporally) than Sentinel-1 & 2 provide. However, our framework may be leveraged to pretrain models on higher-resolution imagery, which could be useful for military applications, although this is a risk of all novel learning algorithms.

A.4.1 Compute

We approximate the computational resources we use for pretraining and finetuning (frozen representation evaluations are negligible in comparison). For pretraining, estimates are in A100-80GB GPU hours; for finetuning, estimates are in A100-40GB GPU hours. Please see Table 12.

Table 12: Estimated GPU hours used for developing and validating CROMA.

Method	Backbone	Task	GPU Hours
radar↔optical [87]	ResNet50	Classification Finetuning	10
radar↔optical [87]	Swin-T	Classification Finetuning	25
MAE [31, 85]	ViT-S	Classification Finetuning	20
DINO [125, 85]	ViT-S	Classification Finetuning	20
SatMAE [26]	ViT-B	Classification Finetuning	75
CROMA	ViT-B	Classification Finetuning	35
SatMAE [26]	ViT-L	Classification Finetuning	215
CROMA	ViT-L	Classification Finetuning	90
SatMAE [26]	ViT-B	Segmentation Finetuning	25
CROMA	ViT-B	Segmentation Finetuning	10
SatMAE [26]	ViT-L	Segmentation Finetuning	65
CROMA	ViT-L	Segmentation Finetuning	30
CROMA	ViT-B	Pretraining 300 epochs	80
CROMA	ViT-L	Pretraining 600 epochs	380
CROMA	ViT-B	Pretraining Ablations	1,100

A.5 Visualizations

We visualize representations and patch encodings using UMAP and t-SNE. For both segmentation datasets (DFC2020 [137] and DW-Expert [138]), we visualize patch encodings of 50,000 randomly sampled patches and use the most dominant class in a patch as its label.

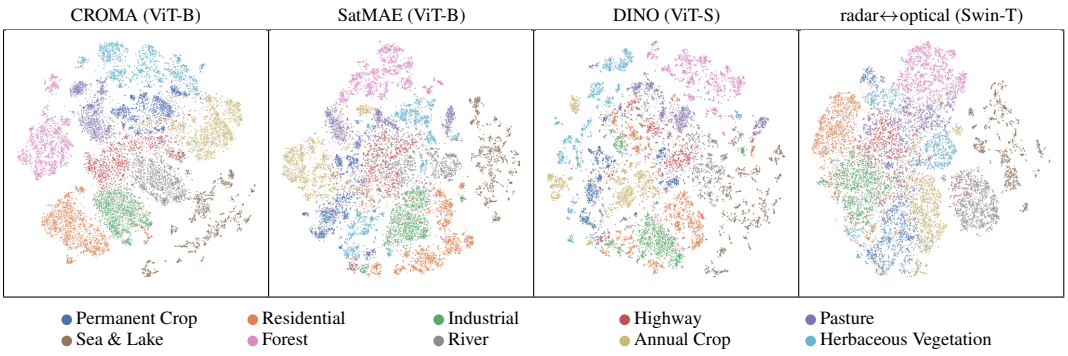

Figure 4: t-SNE plots of EuroSAT [127] representations.

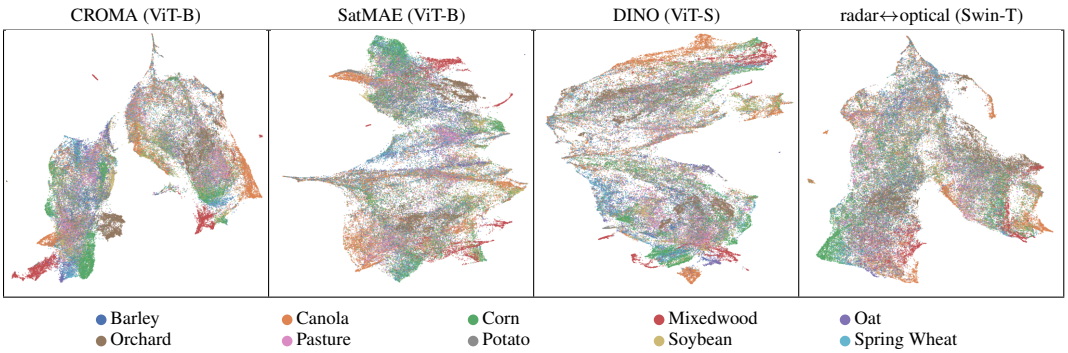

Figure 5: UMAP plots of Canadian Cropland [128] representations.

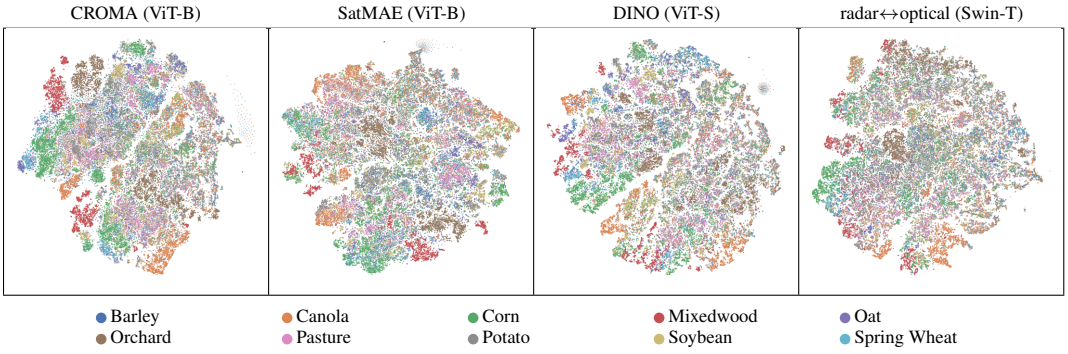

Figure 6: t-SNE plots of Canadian Cropland [128] representations.

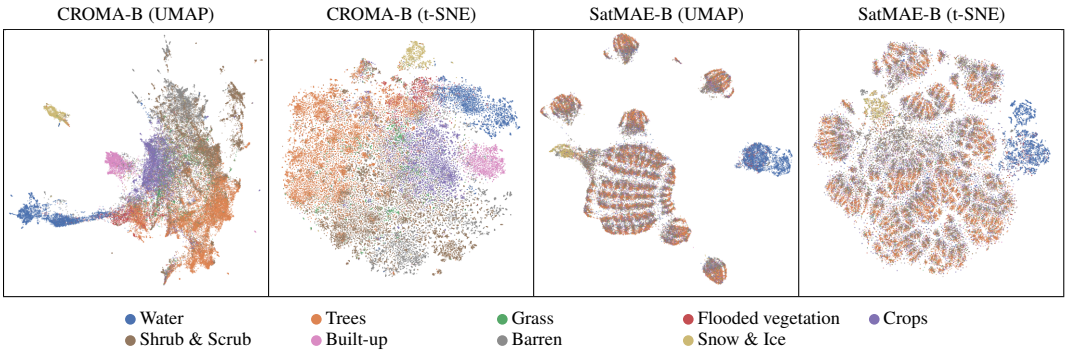

Figure 7: UMAP and t-SNE plots of DW-Expert [138] patch encodings.

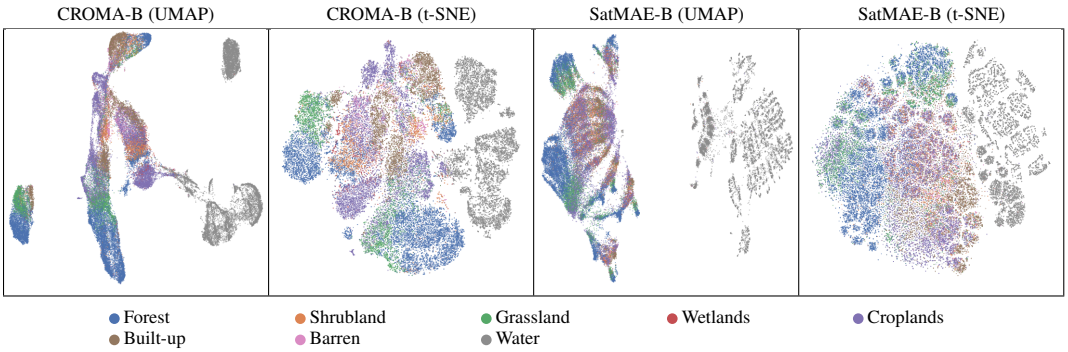

Figure 8: UMAP and t-SNE plots on DFC2020 [137] patch encodings.