A. Sentinel-2 Images and Data Processing Level

Sentinel-2 is a satellite mission from the European Space Agency (ESA), designed for Earth observation under the Copernicus program [18]. It comprises two satellites, Sentinel-2A and Sentinel-2B, launched in 2015 and 2017, respectively. The mission provides high-resolution optical imagery for applications such as land cover classification, environmental monitoring, agricultural analysis, and emergency response. The Sentinel-2 satellites orbit the Earth in a sun-synchronous, polar orbit, capturing images of the entire planet approximately every five days. All Sentinel-2 data is freely accessible.

Spectral Bands and Spatial Resolution. The Sentinel-2 satellites carry a MultiSpectral Instrument (MSI) that captures optical images across 13 spectral bands, spanning from visible (RGB) and near-infrared (NIR) to short-wave infrared (SWIR) regions. These bands have different spatial resolutions, allowing for detailed analysis across diverse applications:

- **10 meters:** The four bands in this range (Blue, Green, Red, and NIR) are particularly useful for visual interpretations and land cover classifications due to their high resolution.
- **20 meters:** Six bands fall in this range, including red edge and short-wave infrared bands, which are instrumental in vegetation analysis, water quality monitoring, and distinguishing various land cover types.
- **60 meters:** Three bands in this range are primarily used for atmospheric correction and cloud screening, with a coarser resolution that provides broader spatial coverage rather than detailed surface features.

Data Processing Levels. Further, Sentinel-2 images are provided at two processing levels, tailored to meet different user needs:

- **Level-1C (L1C):** L1C data consists of Top-Of-Atmosphere (TOA) reflectance values, meaning it captures reflectance as observed from the satellite. This processing level includes the effects of atmospheric conditions like haze and scattering, making it ideal for users who perform their own atmospheric corrections.
- **Level-2A (L2A):** L2A data provides Bottom-Of-Atmosphere (BOA) reflectance values, which means atmospheric corrections have been applied to adjust for atmospheric interference. This data is ready for immediate analysis, allowing users to focus on surface-level characteristics without needing to handle atmospheric correction.

B. Dataset Details

All Sentinel-2-based multispectral datasets used in this paper are preprocessed to L2A processing. Out of the originally 13 bands, 10 bands with a spatial resolution of either $10\,\mathrm{m}$ or $20\,\mathrm{m}$ are selected for experiments. In the following, the pre-training and downstream datasets are described in detail:

SSL4EO [54] is a large-scale unlabeled dataset designed to support SSL in remote sensing. It consists of images of size 264×264 pixels from approximately $250\,000$ diverse locations on Earth that are represented by four seasonal timestamps within the years 2020 and 2021. A time series is included in the dataset if each seasonal interval of 90 days contains at least one tile with less than $10\,\%$ cloud coverage. SSL4EO builds upon the sampling strategy of SeCo [35], which selected multi-seasonal image time series within a $50\,\mathrm{km}$ radius of the $10\,000$ most populated cities worldwide. To address the spatial redundancy introduced by this approach (oversampling), SSL4EO enforces non-overlapping geographic coverage across image locations. In addition to the Sentinel-2 images in L2A processing, each image is further associated with the same image in L1C processing and an image acquired by the radar satellite Sentinel-1. In this paper, we only use the Sentinel-2 image in L2A processing. It is worth noting that the sampling strategy has a strong bias towards populated regions in the northern hemisphere. Locations around the equator are less likely to be selected due to persistent cloud cover throughout the year.

EuroSAT [25] is a single-label classification dataset with $27\,000$ labeled images of size 64×64. It is annotated with 10 land-cover classes that include categories such as forests, agricultural areas, water bodies, and urban zones. The class annotations are derived from the European Urban Atlas. We utilize a stratified train/test/validation split that is composed of $60\,\%$, $20\,\%$, $20\,\%$, respectively. The original version of the dataset is published in L1C processing. To standardize all datasets we converted the images to the L2A processing, and denote the processed dataset as EuroSAT-L2A.

So2Sat [60] is a single-label classification dataset with approximately $400\,000$ labeled image pairs from the satellites Sentinel-1 (radar) and Sentinel-2 (optical) acquired over 50 metropolitan areas worldwide. Each image is of size 32×32. The 17 classes capture both urban and non-urban land cover types and are derived from OpenStreetMap (OSM) data. The dataset provides three different splits to evaluate model performance under varying conditions. The random split (So2Sat-random) divides images randomly across training and test sets ($80\,\%$, $20\,\%$). The block split (So2Sat-block) partitions data based on geographically distinct but neighboring blocks, ensuring less correlation between training and test images ($80\,\%$, $20\,\%$).For standardized experiments, we selected only the Sentinel-2 images.

BigEarthNet-V2 (BEN-V2) [11] is a refined version of the large-scale multi-label dataset BigEarthNet-S2 [45] that includes $590\,326$ images acquired over ten countries in Europe. Each image is of size 120×120. The land use land cover (LULC) class annotations are obtained from the CLC inven-

Figure 5. Example Sentinel-2 images taken from BEN-V2.

tory [6]. Following the LULC class nomenclature proposed in [46], each image is annotated with a subset of 19 LULC classes, including different types of forests, water, or complex urban or agricultural classes. We utilize a filtered subset that excludes images with seasonal snow, clouds, and cloud shadows. The selected subset is divided by a block-wise split into a training set (50 %), a validation set (25 %), and a test set (25 %). Each set can contain different timestamps of the same geographical location.

Sen4Agri-ML is a multi-label classification dataset that was created based on the semantic segmentation dataset Sen4AgriNet [48] designed for agricultural monitoring. All images are acquired over France and Catalonia in the years 2019 and 2020. The originally 225 000 images of size 366×366 composed as time series data were subsampled into images of size 120×120. For each time series, one representative image in the summer months was randomly selected. The respective multi-labels were derived from the 120×120-pixel segmentation maps. Further, all images containing no class were discarded. The 9 high-level crop type class annotations originate from farmer declarations collected via the Land Parcel Identification System (LPIS) [39]. We utilize the random train/test split (denoted as S4A-random) and the tiles-based train/test split (denoted as S4A-tiles) that is composed of training images from France in 2019 and test images from Catalonia in 2020.

CashewPlant [31] is a semantic segmentation dataset derived from Sentinel-2 imagery collected over approximately $120\,\mathrm{km}^2$ in central Benin. It consists of images of size 256×256. Each image is annotated with pixel-wise masks that distinguish seven classes: well-managed plantations, poorly managed plantations, non-plantation, residential areas, background, uncertain, and no-data. The annotations were generated from field surveys with handheld GPS devices and refined with very high-resolution Pléiades imagery. In the GEO-Bench version, the dataset is divided into training (75 %), validation (20 %), and test (5 %) splits.

C. Implementation Details

This section in detail describes the hyperparameters used to train and evaluate the models.

C.1. Data Preprocessing

The reflectance values captured by Sentinel-2 are stored in an *uint16* format. However, the distribution of values is highly skewed towards values within the range of 0 to 4000, with a long tail distribution reaching values up to 2^{13}. To be able to apply channel augmentation techniques to Sentinel-2 data, we preprocess the *uint16* values to *uint8* values by dividing each channel by its 99^{th} percentile for BEN-V2, So2Sat, Sen4Argi-ML and EuroSAT-L2A, followed by a 0-1-clipping and a multiplication by 255. For SSL4EO we divide each channel by its 95^{th} percentile due to a larger long tail in the distribution since both pre-training datasets comprise a higher fraction of images with partial cloud cover. The exact values for the percentiles for each channel can be found in the code repository published together with this paper.

C.2. General Hyperparameter

All self-supervised methods are implemented via packages lightning [15] and lightly [47]. For both the contrastive self-supervised pre-training and the three downstream evaluation protocols we set the batch size to 512. For contrastive self-supervised pre-training that involve MoCoV2 [10], we use the LARS optimizer with a learning rate of 0.4, momentum of 0.9 and a weight decay of 0.000 001 and train the network for 50 epochs. The model with the lowest training loss is selected for downstream evaluation. The InfoNCE is applied with a memory bank size of 4092 and the temperature value of 0.04. For contrastive self-supervised pre-training with SimCLR [8], BYOL [20] and SimSiam [9] we use an SGD optimizer with learning rate of 0.06. The NT-Xent loss for SimCLR follows the default setup with a temperature value of 0.5. BYOL and SimSiam are trained with negative cosine similarity. For DINO [7] pre-training we use an Adam optimizer with learning rate of 0.001. The momentum of the exponential moving average of the model for MoCoV2, BYOL and DINO is compute by a cosine schedule via 10 steps from 0.996 to 1. The DINO loss has an output dimension of 2048 and epochs for the teacher temperature warmup is set to 5. The rest follows the default hyperparameter setting of lightly. For DINO we employ two local views at size 60×60 with scale factor for RRC of (0.25, 0.5) and two global views at size 120×120 with scale factor for RRC of (0.5, 1.0). Similar to MoCoV2, all models are trained for 50 epochs. For *GeoRank* we use the hyperparamter α set to 0.48 and d_{max} set to 2500. The set of data augmentation techniques used for pre-training includes RRC with a ratio of (0.75, 1.33) and a scale of (0.2, 1.0) applied with a probability of 1.0, a flip operation (horizontally and vertically)

applied with a probability of 0.75 and RR90 applied with a probability of 0.75. For the differentiable softrank function that we use to approximate of the rank function we use regularization strength of 0.001 and perform l2 regularization. For MAE training, we adopt the default hyperparameters proposed in the original paper. All three RS MAE variants are trained with a batch size of 16, and learning rate scheduling is deactivated. A masking ratio of 0.75 is used, along with 10 warm-up epochs and the AdamW optimizer (with betas set to (0.9, 0.95) for SatMAE [12] and ScaleMAE [40]). For SatMAE, the output size of the random resized crop (RRC) is set to 96×96, with a scale range of (0.6, 1.0). Feature extraction is performed using all tokens except the class token. The weight decay is set to 0.0, and the learning rate is 0.0001. For CrossScaleMAE [49] and ScaleMAE, the RRC output size is set to 112×112. Additionally, ScaleMAE internally maintains a target size of 224×224 using a constant source size scheduler. Both models use a weight decay of 0.05. The learning rate is set to 0.000 05 for CrossScaleMAE and 0.000 15 for ScaleMAE. In analogy to contrastive methods the maximum training epochs are set to 50. The only data augmentation used in downstream training is random flipping with a probability of 0.8. To save computational cost, the standard preprocessing of SSL4EO consists of a 120×120 pixels centre crop (except for Section 4.5) and a training set that consists of one randomly selected timestamp per location (except for Section 4.4). The pre-training set for the experiments with temporal views (see Section 4.4) is subsampled to $\sim$62 500 locations with each location being present with four different timestamps to avoid measuring artifacts of pre-training dataset saturation.

C.3. Evaluation Protocols

The k-NN evaluation protocol applies a k-NN clustering to the learned representations, the linear evaluation protocol freezes the model backbone and trains a simple linear layer on top of the learned representations, while the fine-tuning protocol re-trains all layers of the backbone. For k-NN evaluation, we set the number of clusters to 10 and the sharpening parameter to 0.9. For linear evaluation and fine-tuning we train for 30 epochs with an AdamW optimizer scheduled by a cosine annealing learning rate scheduling with a start rate set to 0.001 and warm-up iterations based on the number of steps. The weight decay is set to 0.01. Supervised training from scratch is conducted with the same hyperparameter setting as the fine-tuning evaluation protocol. The evaluation protocol for semantic segmentation tasks employs a UPerNet decoder [56] that receives frozen features from layer 1 to 4 for ResNet backbones and is trained for 50 epochs. Hidden feature size is set to 256 and and output feature size is set to 128. We train the UPerNet with an SGD optimization with learning rate 0.02, momentum 0.9 and weight decay of 0.0001. For transformer backbones, we construct multi-scale feature maps by reshaping the encoder sequence into grids of shape $(\tilde{c}, h', w')$ at resolutions $1/4$, $1/8$, $1/16$, and $1/32$. For CrossScaleMAE, the final pyramid level is handled separately by unfolding and bilinearly interpolating features to the target resolution. Channel dimensions are reduced via group-wise averaging to match UPerNet's expected inputs (64, 128, 256, 512). This process, applied independently to each quarter of the transformer blocks, yields a four-level pyramid compatible with standard convolutional decoders.

C.4. Data Augmentation

The default data augmentation pipeline adopted from computer vision (CV) includes RRC with the same hyperparameter setting as in the general hyperparameter, ColorJitter (only applying Contrast and Brightness adjustments) with a limit of 0.4 applied with a probability of 0.8, GrayScale applied with a probability of 0.2, GaussianBlur with a sigma of (0.1, 2.0) applied with a probability of 0.5 and horizontal flipping applied with a probability of 0.5. Hue and Saturation are not defined for more than 3 channels. For the ablation study, we add individual augmentation techniques to the three geometric augmentation techniques from the general hyperparameter with a probability of 0.2. The base magnitudes can be seen in the right column of Table 2. If applicable these are applied with a scalar of 1, 2 or 3. We resized the datasets So2Sat-rand and So2Sat-block to a spatial resolution of 120×120 pixels for all experiments except for Section 4.5. All augmentation techniques are taken from the `albumentation` library.

C.5. Compute Resources

All experiments were conducted on an internal server equipped with 2× AMD EPYC 9554 64-core processors (256 threads), 6× NVIDIA H100 PCIe GPUs (each with 81 GB memory, CUDA 12.2), and 1.5 TiB of system RAM. The system runs Ubuntu 22.04 with Linux kernel 5.15 and NVIDIA driver version 535.183.01. Each training run was executed on a dedicated GPU. Standard pre-training took between 4 and 7 hours, depending on the dataset size and the extent of data augmentation. K-Nearest Neighbors evaluations for downstream tasks required up to 15 minutes per dataset, while fine-tuning evaluations took up to 3 hours. The pre-training of experiments involving geographical regularization required between 10 and 14 hours. Moreover, reproducing results from Ayush et al. involved precomputing k-means clusters, which incurred an additional small overhead.

D. Extended Experiments

In this section, we present complementary results on different pre-training datasets.

Table 7. Ablation study for enabling or disabling one of the three basic geometric augmentation techniques. Performance is the averaged score (Avg. Result) in the k-NN protocol over all six downstream tasks when pre-training on SSL4EO.

RRC	RR90	Flip	Avg. Result
✓	-	-	63.64
-	✓	-	60.19
-	-	✓	55.07
✓	✓	-	**68.59**
✓	-	✓	65.73
-	✓	✓	62.25
✓	✓	✓	**68.62**

D.1. Data Augmentation Ablation for Geometric Augmentation

We extend the ablation study presented in Section 4.1 and also evaluate the average performance on all downstream tasks on permutations of the three geometric augmentation techniques RRC, Flip and RR90 (see Table 7). The results indicate that the biggest driver for downstream performance is RRC. Nonetheless, the combinations of RRC with RR90 and Flip and yield the highest averaged downstream performance.

D.2. Data Augmentation for BEN-V2

In line with the results of comparing the default computer vision data augmentation pipeline with a geometric augmentation pipeline when pre-training on the SSL4EO dataset, we find that the geometric pipeline outperforms the standard pipeline by values of up to 15 % in the k-NN protocol when pre-training on BEN-V2 (see Figure 6a). It is noteworthy that this effect diminishes when more hyperparameters are involved in the evaluation protocol: while the average improvement for linear evaluation is between 3 % and 5 %, the differences become marginal when observing the results for the fine-tuning protocol (see Figure 6c). Especially the evaluation under the k-NN protocol emphasizes the relevance of adjusting the data augmentation pipeline to multispectral RS images.

D.3. Qualitative Analysis of Representation Space

To assess the qualitative effect of the proposed regularization, we compare latent representations obtained from the baseline model (MoCoV2) and MoCoV2 with *GeoRank*. From the BEN-V2 training set, we randomly sample 2560 images and extract features from the penultimate layer of each model. The resulting representations are reduced in dimensionality using principal component analysis (PCA) to 50 components, followed by t-SNE with perplexity set to 30 and learning rate set to 200. The first row of Figure X visualizes the embeddings colored by normalized latitude, while the second row uses normalized longitude. MoCoV2 with *GeoRank* exhibits smoother spatial organization in the embedding space, with representations reflecting relative geographical ordering rather than forming rigid clusters. This observation is consistent with the intended rank-based formulation, which preserves ordering relations without enforcing strict alignment.

D.4. Compatibility with RS-Specific Contrastive SSL Methods

Beyond standard contrastive algorithms, we also tested *GeoRank* with RS-specific SSL methods that incorporate temporal or multimodal information. Specifically, we combined *GeoRank* with Seasonal Contrast (SeCo) [35] and CROMA [17], which represent temporal and multimodal contrastive learning respectively. Results are reported in Table 8. Improvements are generally modest, with gains on most benchmarks. Consistent with the main experiments, *GeoRank* shows a drop in performance only in the presence of geographical domain shift, as observed on S4A-tiles. Overall, these experiments confirm that *GeoRank* can be integrated into temporal and multimodal SSL setups without interfering with their design objectives.

D.5. Dataset Cardinality for SSL4EO under different Model Sizes

Against the hypothesis of Wang et al. [54], we observe no significant differences in saturation for different model sizes when we pre-train different sizes of ResNets on SSL4EO (see Figure 8).

D.6. Downstream Image Size for different Pre-Training Image Sizes

We find that for a fixed pre-training image size of 60×60 pixels, 120×120 pixels or 264×264 pixels, resizing the downstream images to larger image sizes tends to result in an increase in performance for all downstream tasks (see Table 9, Table 10 and Table 11). Similar to Corley et al. [13], we observe a saturation effect at 264×264 pixels for the resizing of the downstream task for a pre-training image size of 264×264 pixels. We note that for a larger gap between pre-training image size and downstream resizing, e.g., 60×60 to 264×264, a downstream resizing of 120×120 can be already effective.

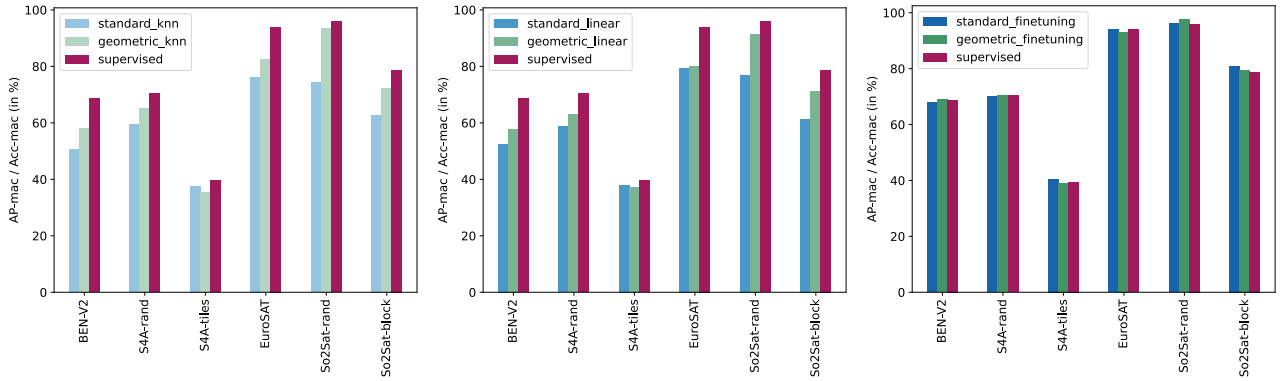

Figure 6. Performance comparison between the standard augmentation pipeline (blue), the geometric augmentation pipeline (green) and supervised training (red) on all six downstream tasks when pre-training on BEN-V2: (a) evaluated with the k-NN evaluation protocol, (b) evaluated with the linear evaluation protocol, (c) evaluated with the fine-tuning protocol.

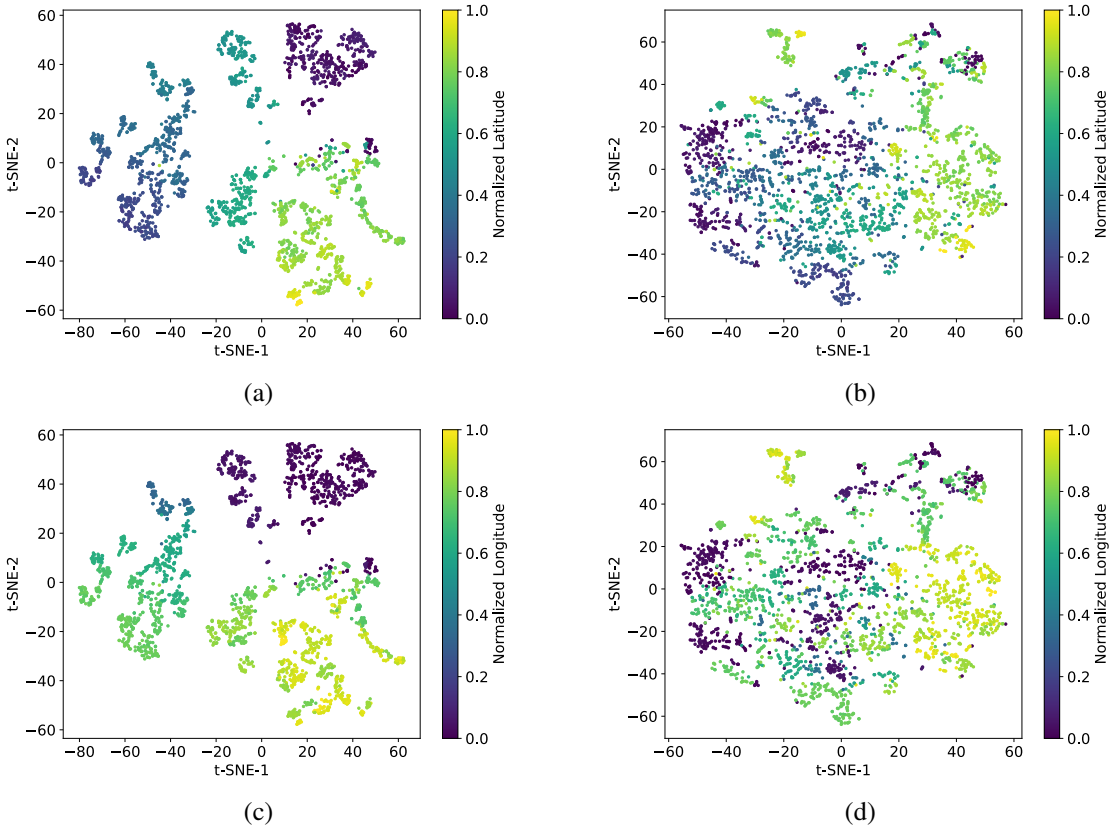

Figure 7. t-SNE of penultimate layer representations for 2560 BENV2 samples after PCA (50 components). Points are colored by normalized latitude in (a) MoCoV2 with *GeoRank* and (b) MoCoV2, and by normalized longitude in (c) MoCoV2 with *GeoRank* and (d) MoCoV2.

Table 8. Extending existing RS-specific SSL methods with *GeoRank* , when pre-training on SSL4EO, evaluated by the k-NN protocol.

Method	BEN-V2	EuroSAT	S4A-rand	S4A-tiles	So2Sat-rand	So2Sat-block	BEN-V2 S1+S2
CROMA [17]	61.13	89.77	65.53	**36.23**	94.69	**75.09**	61.62
CROMA [17] + GeoRank	**61.34**	**89.96**	**65.67**	35.27	**94.79**	**75.10**	**61.72**
SeCo [35]	57.64	86.22	64.42	**36.89**	91.52	**75.69**	
SeCo [35] + GeoRank	**57.99**	**86.60**	**64.72**	36.34	**92.08**	**75.67**	-

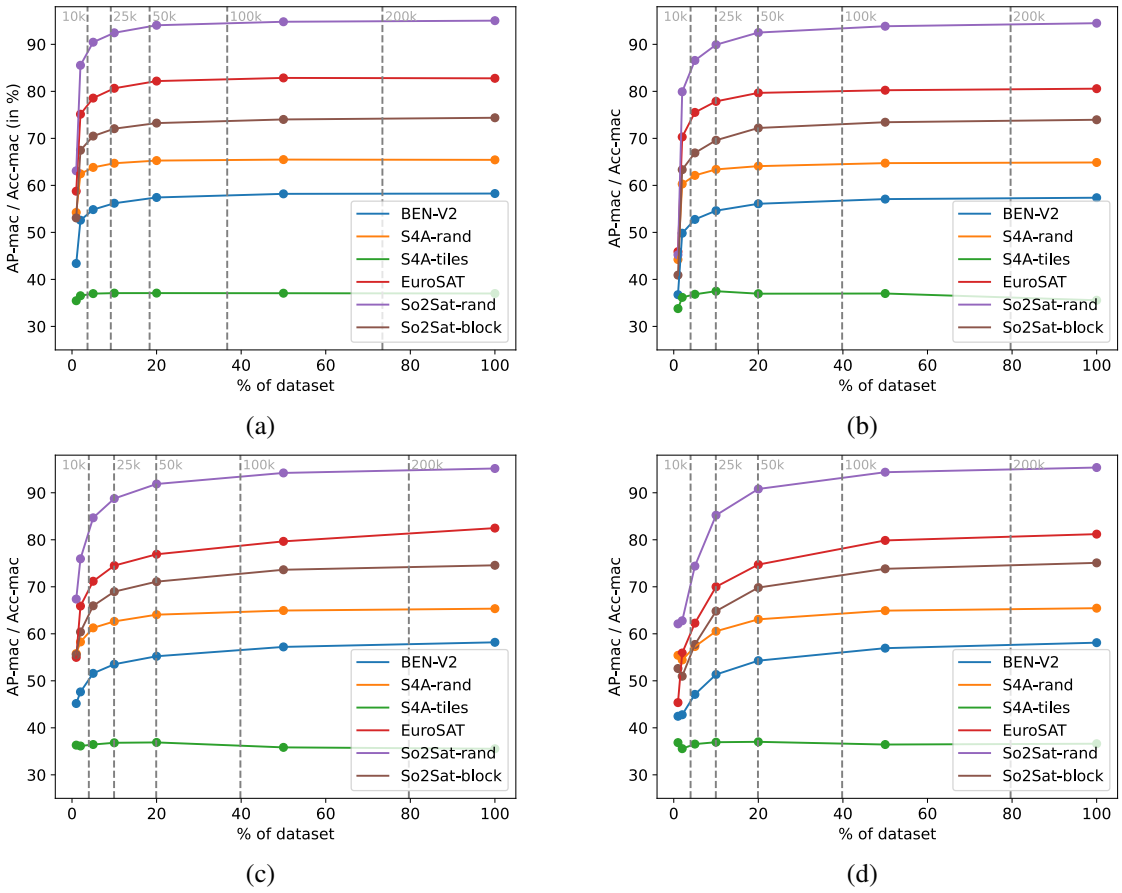

Figure 8. Performance of different subset sizes of the pre-training dataset SSL4EO evaluated on all six downstream tasks by k-NN with different backbones. (a) ResNet18. (b) ResNet34. (c) ResNet50. (d) ResNet101.

Table 9. Performance (in %) of different resizing strategies for downstream datasets evaluated by the k-NN protocol when pre-training on SSL4EO with fixed image size. The first image size (left of the arrow) is the center cropped size of the pre-training dataset, and the second image size (right of the arrow) is the resized downstream image size.

Image Size	BEN-V2	EuroSAT	S4A-rand	S4A-tiles	So2Sat-rand	So2Sat-block
60x60 →original	56.72	82.85	**65.21**	35.62	85.63	69.09
60x60 →120x120	56.72	**84.13**	**65.21**	35.62	94.14	**73.71**
60x60 →264x264	**56.95**	83.97	64.83	**35.94**	**94.51**	**73.84**

Table 10. Performance (in %) of different resizing strategies for downstream datasets evaluated by the k-NN protocol when pre-training on SSL4EO with fixed image size. The first image size (left of the arrow) is the center cropped size of the pre-training dataset, and the second image size (right of the arrow) is the resized downstream image size.

Image Size	BEN-V2	EuroSAT	S4A-rand	S4A-tiles	So2Sat-rand	So2Sat-block
120x120 →original	58.34	82.22	65.39	**36.21**	78.76	64.80
120x120 →120x120	58.34	85.57	65.39	**36.21**	95.06	74.57
120x120 →264x264	**59.18**	**86.32**	**66.21**	**36.27**	**96.02**	**75.10**

Table 11. Performance (in %) of different resizing strategies for downstream datasets evaluated by the k-NN protocol when pre-training on SSL4EO with fixed image size. The first image size (left of the arrow) is the center cropped size of the pre-training dataset, and the second image size (right of the arrow) is the resized downstream image size.

Image Size	BEN-V2	EuroSAT	S4A-rand	S4A-tiles	So2Sat-rand	So2Sat-block
264x264 →original	57.44	80.41	64.65	**36.45**	70.68	59.15
264x264 →120x120	57.44	84.44	64.65	**36.45**	92.87	73.45
264x264 →264x264	**59.08**	**86.22**	**66.06**	**36.61**	**95.64**	**75.04**