Learning Representations of Satellite Images From Metadata Supervision – Supplementary Material

Jules Bourcier[1,2], Gohar Dashyan[1], Karteek Alahari[2], and Jocelyn Chanussot[2]

[1] Preligens, Paris, France
[2] Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, Grenoble, France

A Datasets details

We report properties of the used datasets in Tab. S1, including number of train and test samples, number of classes, image resolution, GSD range, location extent and the sensors comprising each dataset.

Table S1: Details of classification datasets used in experiments.

Dataset	Num. train samples	Num. test samples	Num. classes	Resolution (px)	GSD (m)	Location extent	Sensor(s)
fMoW [7]	363,572	53,043	62	224×224	0.06–23	207 countries / 400 UTM zones	WorldView-2, WorldView-3, QuickBird-2, GeoEye-1
RESISC45 [6]	18,900	6,300	45	256×256	0.2–30+	global	various (Google Earth)
Optimal31 [21]	930	930	31	256×256	0.5–8	global	various (Google Earth)
UC Merced [23]	1260	840	21	256×256	0.3	Contiguous USA	NAIP
FGSC-23 [24]	3256	824	23	variable (40–800)	0.4–2	*Unknown*	various (Google Earth), GaoFen-1
EuroSAT [13]	16,200	5,400	10	64×64	10	34 European countries	Sentinel-2
So2Sat [26]	352,366	48,307	17	32×32	10	42 cities distributed globally	Sentinel-2

Spectral bands. We perform all experiments on three-bands RGB images. For EuroSAT and So2Sat which provide additional spectral bands, we retain only the RGB bands.

Data splits. We use one train and one test split for all datasets. For fMoW, we use the official train and validation splits as our train and test splits respectively, following [1,8,18]. For RESISC45, UC Merced, EuroSAT and So2Sat, we use the train and test splits available in TorchGeo [19], which are taken from [16]. For UC Merced, we use the combined test and val splits as our test set, to inflate its size. For So2Sat, we use the "Culture-10" version of the dataset. For Optimal31, we randomly split the full dataset (1,860 samples) between train and test with a 50/50 ratio.

fMoW preprocessing. Our preprocessing of fMoW aligns with previous works [1, 7,8]. We use the fMoW-RGB dataset product composed pansharpened color images converted to 8-bit JPEGs files, and JSON metadata files. We preprocess the dataset using the standard method: each image is cropped around an area of interest (AOI) and resized to 224×224 pixels. Resized cropping affects the associated GSD and location. We transform the GSD height and width according to the size ratio of the cropped image to the resized cropped image. We replace the location polygon with the one encompassing the AOI. Other metadata fields are not affected.

Resizing and normalization. For the evaluation of fMoW-pretrained models, we follow [9] for resizing and normalizing of images. We resize to the resolution used for pretraining (224×224 pixels) or keep the original size if it is higher. Doing so tends to give optimal performance for all the compared models on all datasets, except for Scale-MAE, which we evaluate using a resolution of 128×128 pixels on all datasets as it gives better performance[3]. For pretraining and evaluation, we perform channel-wise standardisation with mean and standard deviation statistics computed on the training set of each dataset [9].

B Pretraining details

Visual encoders. We follow [15] for the configurations of the ViT backbones. We use the ViT-S variant from MoCoV3 [5] with 12 heads per attention layer (*vs*. 6 in original ViT-S [20]). We use a patch size of 16, and learnable positional embeddings. The output representation that is passed to the projection head for pretraining, and used for downstream tasks, is the `CLS` token of the last layer.

Textual metadata encoders. For our experiments with a textual representation of metadata, we apply the following processing. Following [25], we format different fields as key-value pairs of strings, and concatenate each key-value pair together to form a composite string using the syntax "$\texttt{key}_1$`: `$\texttt{value}_1$`, ...,` $\texttt{key}_n$`: `$\texttt{value}_n$". Afterwards, we tokenize the text using the CLIP's Byte Pair

[3] this is consistent with the results of [18]

Encoding (BPE) tokenizer. We then feed the sequence of tokens to a BERT-style [10] Transformer encoder with 3 layers, width 512, 8 attention heads per layer, and a FFN size factor of 4. We use learnable positional embeddings. We use the "pre-norm" variant of Transformer following [17]. The output representation that is passed to the projection head for pretraining, and used for downstream tasks, is the `CLS` token of the last layer.

Tabular metadata encoders. For our experiments with metadata as tabular features, we decompose the metadata into atomic numerical or categorical fields; the only field for which this is not straightforward is timestamp, which we convert into year, month, day, hour, and weekday. The numerical features are further standardized by removing the mean and scaling to unit variance. We concatenate both numerical and categorical vectors and pass as input to a FT-Transformer [11] composed of a feature tokenizer (see [11] for details) and a Transformer with 3 layers, a width of 192, 8 attention heads per layer and a FFN size factor of $4/3$. We use the "pre-norm" variant of Transformer, and remove the first normalization from the first layer following [11]. The output representation that is passed to the projection head for pretraining, and used for downstream tasks, is the `CLS` token of the last layer.

Projection heads. We follow [15] for the configuration of projection heads. The projection head for the metadata-image loss in SatMIP(s) is a linear layer specific to each modality that map each representation to a 512-dim embedding. The projection head for the image-image loss in SimCLR and SatMIPS is a MLP composed of 3 4096-dim hidden layers, interposed with BatchNorm and ReLU, and outputs 256-dim embeddings.

Temporature scaling in contrastive loss. Following [15], the temperature τ is set to 0.1 for the image-image loss in SimCLR and SatMIPS, while it is set to a learnable parameter in the metadata-image loss in SatMIP(S).

Augmentation. We use the same augmentation policy across image inputs in SimCLR/SatMIP(S). Borrowing from [2], we opt for a modified version of the standard SimCLR policy for satellite images, composed of: random resized cropping with a scale ratio sampled uniformly in $[0.2, 1.0]$ and target size 224 px, color jittering with $p = 0.8$, grayscaling with $p = 0.2$, Gaussian blurring with $p = 0.5$, horizontal flipping with $p = 0.5$; vertical flipping with $p = 0.5$, and rotation with $p = 0.75$ by an angle sampled uniformly in $\{90, 180, 270\}$. In the ablation of Tab. 6 in the main paper, the "crop" policy is random resized cropping with a scale ratio sampled between $[0.5, 1.0]$ [15].

Training. Most of our training hyperparameters are reused from [15], and we translate their recipes of CLIP and SLIP to our SatMIP and SatMIPS, respectively. We perform stochastic gradient descent with the AdamW [14] optimizer

(with $(\beta_1, \beta_2) = (0.9, 0.98)$ and $\epsilon = 1e-8$). We use a global batch size of 1024, and a cosine learning rate decay, with 1 epoch of linear warmup [4]. We apply the linear scaling rule [12] to set the initial learning rate: $\text{lr} = \text{lr}_{\text{base}} \cdot \text{bs}/256$, with bs the batch size and lr_{base} a base learning rate. We train the models with mixed precision. Base learning rate and weight decay have different values depending on the model, given in the following table:

Model	SimCLR	SatMIP	SatMIPS
base learning rate	2e-4	1.875e-4	3.75e-4
weight decay	0.1	0.5	0.5

In SatMIPS loss, we set the value of λ to 1. We show the impact of λ in Tab. S4.

Code and compute environment. Our implementation of SimCLR, SatMIP and SatMIPS is based on the official code of SLIP[4]. We use PyTorch 2.1. Trainings are performed on compute nodes with 4 Nvidia V100-32GB or 8 Nvidia A100-40GB. kNN evaluations are performed on one V100-32GB.

C kNN classification details

After pretraining, the representation we evaluate is the `CLS` token output of ViT backbones. We use a weighted kNN classifier following standard practice [3, 18, 22]. We freeze the pretrained model and extract the representations of the training and testing set examples. We classify each test sample by performing a weighted vote among the top k training samples sorted by decreasing cosine similarity. We do not use any data augmentation. We sweep the number of neighbors k in the set $\{1, 5, 20, 100\}$ for each model and dataset combination, and report optimal results. For all contrastive-based models, we select $k = 100$ on fMoW and $k = 5$ on the other datasets. For MAE-based models, we select $k = 20$ on fMoW and $k = 5$ on the other datasets. We enable mixed precision for feature extraction and calculating the pairwise similarities between samples.

D Linear probing classification details

For linear probing, we fit a logistic regression classifier on the training set embeddings, using L-BFGS optimizer with 200 maximum iterations, and no regularization. For bimodal classification, we first extract image and metadata features and concatenate both [CLS] token embeddings, standardize the feature to zero mean and unit variance, and fit a logistic regression classifier.

E Description of fMoW metadata

In Tab. S2, we detail the full set of 15 metadata fields from fMoW that we considered throughout our experiments. Recall that by default, we used the

[4] `https://github.com/facebookresearch/SLIP`

subset of GSD, timestamp, and location fields (row (1), (4), and (5) in Tab. S2, respectively). This full set of 15 fields was used in the ablation in **??** of the main paper.

Additionally, we visualize the distribution of the main metadata fields:

- *GSD:* In Fig. S1, we observe that the vast majority of GSD width and height values are concentrated between 0.3 m and 2 m. The distribution has a long tail of higher GSD values ranging up to 23 m.
- *Location:* In Fig. S2, we see that locations span a global distribution across all five continents. However, we note an overall bias towards the global North, while some regions, such as Subsaharan Africa and South Asia, are underrepresented.
- *Timestamp:* In Fig. S3, we see that dates are unequally spread in the full 12-years time range, with the majority being 2014 and 2017. Months and weekdays, however, are more uniformly distributed.

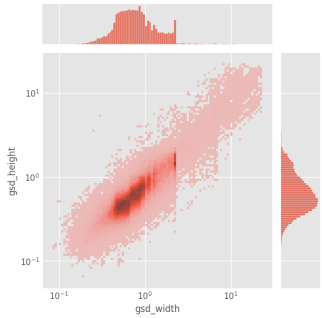


Fig. S1: Distribution of ground sampling distances (width and height) in the fMoW training set. Note the log scale.

F Examples of images and metadata

Tab. S3 presents sample images and metadata pairs from the fMoW dataset, that we used for metadata-image pretraining within SatMIP and SatMIPS. Metadata is here shown as text form.

G Additional ablations

We present additional ablations of our SatMIP and SatMIPS models.

Table S2: Details of the full set of 15 metadata fields selected from the fMoW dataset for our experiments. Colors designate two types of metadata: (a) sensor: fields that are determined by the sensor's characteristics and/or it's relative position to the target); (b) environment: fields that are determined by the environment (*i.e.*, the geotemporal context). Refer to the fMoW paper [7] for a detailed documentation.

Field	Description	Example value
Ground sample distance	GSD of panchromatic band in the raw image strip, in meters. Transformed according to resized cropping (*cf*. Sec. A).	`[0.3749, 0.2916]`
Multispectral ground sample distance	GSD of multispectral bands in the raw image strip, in meters. We include the average of width and height. Transformed according to resized cropping.	`1.3365`
Pixel size	Size of a pixel in longitude and latitude units in the panchromatic band, in degrees. Transformed according to resized cropping.	`[3.27e-06, 2.54e-06]`
Timestamp	ISO UTC timestamp of acquisition down to the second.	`2016-07-02 T12:40:44Z`
Location	Longitude (-180–180) and latitude (-90–90) of the image centroid, in degrees. Transformed according to cropping (*cf*. Sec. A).	`[-43.246798, -22.982982]`
UTM zone	Provides a number for the UTM zone (1–60), along with a letter representing the latitude band ("C"–"X").	`23K`
Country code	ISO alpha-3 country code.	`BRA`
Cloud cover	Percentage of the raw image strip that is completely obscured by clouds (0–100).	`0`
Scan direction	Direction in which the sensor is pointed during take, relatively to the orbital path. Equals "Forward" if taken ahead of the orbital path and "Reverse" if taken behind.	`Reverse`
Wavelengths	Approximate central wavelength of the red, green and blue bands. [b]	`[661, 545, 477]`
Target azimuth	Azimuth angle of the sensor to the center of the image strip, from north, clockwise, in degrees (0–360).	`0.58`
Sun azimuth	Azimuth angle of the sun to the center of the image strip, from north, clockwise, in degrees (0–360).	`67.86`
Sun elevation	Elevation angle of the sun from the horizon, in degrees (0–90).	`61.34`
Off-nadir angle	The off-nadir angle of the sensor to the center of the image strip, in degrees (0–90).	`43.92`
Sensor platform	Name of the sensor, among: WorldView-2, WorldView-3, QuickBird-2, and GeoEye-1.	`GEOEYE01`

[a] Note that all sensors capture at the same wavelengths, so this field is constant throughout the dataset, making it inoperative.

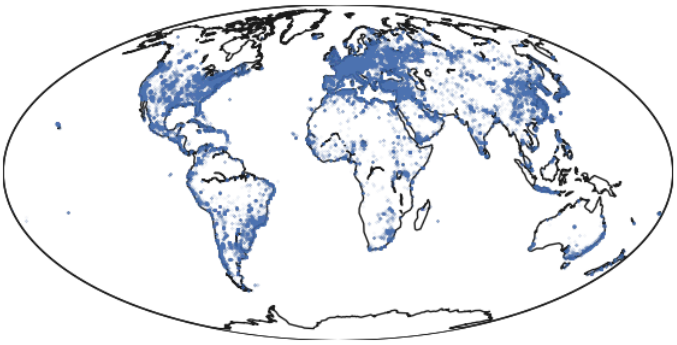


Fig. S2: Distribution of geographic locations in the fMoW training set.

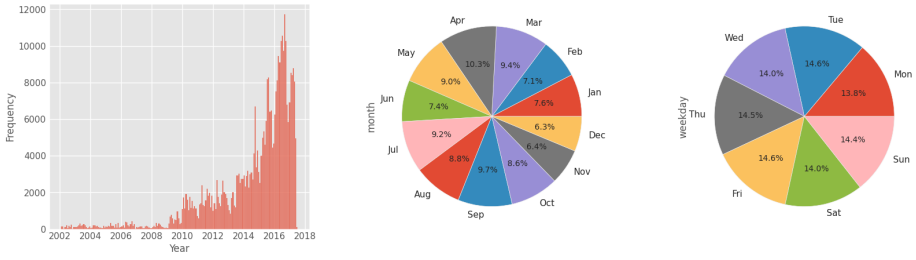
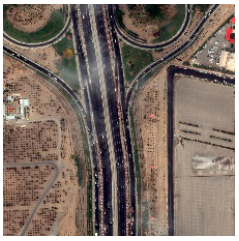


Fig. S3: Distribution of timestamps' years, months and weekdays in the fMoW training set.

Table S3: Sample images from the fMoW dataset with their metadata as a formatted text, using the full set of 15 fields described in Tab. S2. We also report the number of resulting text tokens (excluding start-of and end-of-text tokens), and the class the sample belongs to.

	Image	Metadata (text)	Class
(1)		ground_sample_distance: [11.4546, 9.1344], multispectral_ground_sample_distance: 41.1896, pixel_size: [1.03e-04, 8.23e-05], timestamp: 2015-09-21T15:30:08Z, location: [-73.310776, -3.785814], utm_zone: 18M, country_code: PER, cloud_cover: 14, scan_direction: Reverse, wavelengths: [661, 545, 477], target_azimuth: 39.12, sun_azimuth: 77.28, sun_elevation: 70.44, off_nadir_angle: 27.70, sensor_platform: GEOEYE01	Airport
(2)		ground_sample_distance: [2.0638, 1.8120], multispectral_ground_sample_distance: 7.7561, pixel_size: [1.86e-05, 1.64e-05], timestamp: 2016-07-02T07:33:42Z, location: [51.253020, 35.711928], utm_zone: 39S, country_code: IRN, cloud_cover: 3, scan_direction: Forward, wavelengths: [661, 545, 477], target_azimuth: 312.30, sun_azimuth: 126.03, sun_elevation: 71.28, off_nadir_angle: 22.31, sensor_platform: WORLDVIEW03_VNIR	Interchange
(3)		ground_sample_distance: [1.8129, 1.9807], multispectral_ground_sample_distance: 7.5664, pixel_size: [1.65e-05, 1.80e-05], timestamp: 2014-05-27T03:05:01Z, location: [120.572210, 14.984249], utm_zone: 51P, country_code: PHL, cloud_cover: 6, scan_direction: Reverse, wavelengths: [661, 545, 477], target_azimuth: 85.94, sun_azimuth: 58.19, sun_elevation: 76.37, off_nadir_angle: 25.91, sensor_platform: WORLDVIEW02	Crop Field
(4)		ground_sample_distance: [2.1283, 1.3499], multispectral_ground_sample_distance: 6.9531, pixel_size: [1.89e-05, 1.20e-05], timestamp: 2005-12-21T17:59:22Z, location: [-105.222951, 39.749676], utm_zone: 13S, country_code: USA, cloud_cover: 2, scan_direction: Forward, wavelengths: [661, 545, 477], target_azimuth: 289.08, sun_azimuth: 164.96, sun_elevation: 25.32, off_nadir_angle: 25.12, sensor_platform: QUICKBIRD02	Educational Institution
(5)		ground_sample_distance: [0.3191, 0.4155], multispectral_ground_sample_distance: 1.4697, pixel_size: [3.24e-06, 4.22e-06], timestamp: 2017-04-26T10:08:48Z, location: [6.462309, 13.403795], utm_zone: 32P, country_code: NGA, cloud_cover: 0, scan_direction: Reverse, wavelengths: [661, 545, 477], target_azimuth: 341.90, sun_azimuth: 84.84, sun_elevation: 69.78, off_nadir_angle: 17.99, sensor_platform: GEOEYE01	Single-Unit Residential

G.1 Multi-task balancing in SatMIPS loss

In Tab. S4, we ablate the value of the hyperparameter λ, which balances the metadata-image and image-image objectives. We pretrain on fMoW for 25 epochs with a textual metadata encoder. We observe that performance is not significantly impacted by the choice of λ, provided that the value is greather than 0 (or it is equivalent to SatMIP). SatMIPS can benefit equally from both objectives regarless of their weighting.

Table S4: Impact of the multi-task loss balancing factor λ in SatMIPS. Note that $\lambda = 0$ is equivalent to SatMIP as the SimCLR objective is null.

λ	fMoW F1	R45 Acc.	F23 Acc.	So2 Acc.
0	$45.6_{\pm 0.4}$	$81.9_{\pm 0.2}$	$54.1_{\pm 0.4}$	$54.7_{\pm 0.5}$
0.5	$53.7_{\pm 0.3}$	$86.8_{\pm 0.03}$	$57.0_{\pm 0.4}$	$56.0_{\pm 0.1}$
1.0	$53.9_{\pm 0.1}$	$87.1_{\pm 0.02}$	$57.6_{\pm 0.5}$	$56.2_{\pm 0.6}$
2	$53.9_{\pm 0.4}$	$86.7_{\pm 0.1}$	$58.4_{\pm 1.1}$	$56.3_{\pm 0.2}$

G.2 Textual *vs.* tabular metadata encoders in SatMIP

We present an extensive comparison of the two approaches we adopt for encoding metadata within SatMIP: using a text encoder (BERT-style Transformer on textualized inputs), and a tabular encoder (FT-Transformer on featurized inputs). Our choice for using a textual encoder was motivated by [25], who demonstrated the flexibility and effectiveness of textual encoding on EXIF tags. Nevertheless, we may hypothesize that a textual representation should be ill-suited for numerical fields such as location or GSD: as it treats them as sequences of digit tokens, it must limited understanding of those fields. Using vectorized features as input to a tabular encoder must be more suited for numerical fields by definition. First, in Tab. S5, we compare the kNN classification performance of SatMIP(S) trained with both type of encoders on the various datasets as well as their efficiency. We observe that for SatMIP, surprisingly, a textual encoder tends to perform better, with higher accuracies on 5 out of the 7 datasets. For SatMIPS, their performance is on par overall, except in favor of the textual encoder on one dataset (O31). These results indicate that a textual encoder tends to be more effective, although the tabular encoder is competitive. However, this observation may just be due to the choice of hyper-parameters, as we mostly reused the hyperparameters from SLIP [15] with minimal tuning, and SLIP uses a textual encoder on language captions. Therefore, we cannot draw any definitive conclusions. However, we note that the tabular encoder is more memory efficient, as it requires way less tokens to train (about 10× for GSD, timestamp and location as inputs).

Then, in Sec. G.2, we compare the linear probing performance of SatMIP(S) trained with both encoders using multiple modalities. We observe than when metadata features are used as input to classification, the tabular encoder peforms much better than the textual encoder, which is in contrast to using image features alone. This clearly shows that numerical fields understanding is important for deploying metadata encoders. The textual encoder might be good at solving the image-metadata matching task from token sequences, but it is limited in it's ability to generalize to new data on downstream tasks. This shows that a tabular encoder should be favored when considering multimodal classification with SatMIP(S).

Table S5: kNN classification performance with a tabular encoder *vs*. a textual encoder for metadatas. 200 epochs pretraining on fMoW. Training time and memory usage are relative to the baseline, SimCLR.

Model	Encoder	fMoW F1	R45 Acc.	O31 Acc.	UCM Acc.	FGSC-23 F1	Euro Acc.	So2 Acc.	Train. time	Mem. /GPU
SatMIP	Textual	$55.2_{\pm 0.2}$	$87.5_{\pm 0.1}$	$\mathbf{84.8}_{\pm \mathbf{0.6}}$	$\mathbf{95.2}_{\pm \mathbf{0.8}}$	$\mathbf{56.4}_{\pm \mathbf{0.2}}$	$\mathbf{95.7}_{\pm \mathbf{0.5}}$	$\mathbf{55.9}_{\pm \mathbf{0.2}}$	0.56	0.62
	Tabular	$\mathbf{55.8}_{\pm \mathbf{0.3}}$	$87.2_{\pm 0.3}$	$82.4_{\pm 0.9}$	$94.3_{\pm 0.3}$	$55.3_{\pm 1.3}$	$94.2_{\pm 0.4}$	$55.2_{\pm 0.5}$	0.56	**0.52**
SatMIPS	Textual	$62.4_{\pm 0.1}$	$89.7_{\pm 0.1}$	$\mathbf{88.1}_{\pm \mathbf{1.1}}$	$95.2_{\pm 0.6}$	$58.8_{\pm 0.5}$	$94.8_{\pm 0.1}$	$57.3_{\pm 0.1}$	1.05	1.11
	Tabular	$62.5_{\pm 0.4}$	$89.6_{\pm 0.2}$	$86.5_{\pm 0.9}$	$95.6_{\pm 0.4}$	$59.2_{\pm 1.1}$	$94.9_{\pm 0.2}$	$57.2_{\pm 0.4}$	1.05	**1.01**

H Additional results

We report additional results corresponding to the experiments presented in the main paper.

We have analyzed the time and memory efficiency of method relatively to our baseline (SimCLR). In Tab. S7, we report the absolute numbers, corresponding to the 200 epochs pretraining runs in **??** of the main paper.

I CO_2 emissions related to experiments

Experiments performed throughout this project consumed a total of 5,123 hours of V100-SXM2-32GB compute and 9358 hours of A100-SXM4-80GB compute. We performed our experiments on the Jean Zay HPC cluster from IDRIS, located in Orsay, France. As reported by our HPC cluster monitoring tool, the experiments amount to a total of 0,504 T CO_2eq.

References

1. Ayush, K., Uzkent, B., Meng, C., Tanmay, K., Burke, M., Lobell, D., Ermon, S.: Geography-aware self-supervised learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10181–10190 (2021)

Table S6: Linear probing classification on fMoW using multiple modalities: image, metadata, or both, after pretraining on fMoW for 25 or 200 epochs.

		fMoW F1	
Model	**Modality**	**25 epochs**	**200 epochs**
SatMIP	Image	49.5±0.3	57.6±0.5
	Meta	18.7±0.6	19.5±0.3
	Image+Meta	**54.0**±0.5	**59.3**±0.6
SatMIPS	Image	57.7±0.1	66.3±0.4
	Meta	19.4±1.2	19.6±0.2
	Image+Meta	**60.5**±0.9	**67.0**±1.5

(a) Textual metadata encoder

		fMoW F1	
Model	**Modality**	**25 epochs**	**200 epochs**
SatMIP	Image	50.7±0.4	59.3±0.3
	Meta	27.8±0.1	27.8±0.2
	Image+Meta	**57.7**±0.5	**63.1**±0.1
SatMIPS	Image	59.5±0.1	65.8±0.1
	Meta	27.9±0.1	27.8±0.1
	Image+Meta	**64.6**±0.2	**68.6**±0.2

(b) Tabular metadata encoder

Table S7: Absolute training time and peak memory usage of the different pretraining methods, for 200 epochs, with ViT-S backbone and a batch size of 1024 distributed over 4 Nvidia V100-32GB. The reported training times are averages of 3 runs.

Model	**Training time (minutes)**	**Peak memory usage (GiB)**
SimCLR	392	17.3
SatMIP	222	10.7
SatMIPS	414	19.2

2. Bourcier, J., Dashyan, G., Chanussot, J., Alahari, K.: Evaluating the label efficiency of contrastive self-supervised learning for multi-resolution satellite imagery. In: Image and Signal Processing for Remote Sensing XXVIII. vol. 12267, pp. 152–161. SPIE (2022)
3. Caron, M., Touvron, H., Misra, I., J'egou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. 2021 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 9630–9640 (2021)
4. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
5. Chen, X., Xie, S., He, K.: An empirical study of training self-supervised vision transformers. 2021 IEEE/CVF International Conference on Computer Vision (ICCV) pp. 9620–9629 (2021)
6. Cheng, G., Han, J., Lu, X.: Remote sensing image scene classification: Benchmark and state of the art. Proceedings of the IEEE **105**(10), 1865–1883 (2017)
7. Christie, G., Fendley, N., Wilson, J., Mukherjee, R.: Functional map of the world. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6172–6180 (2018)
8. Cong, Y., Khanna, S., Meng, C., Liu, P., Rozi, E., He, Y., Burke, M., Lobell, D., Ermon, S.: SatMAE: Pre-training transformers for temporal and multi-spectral satellite imagery. Advances in Neural Information Processing Systems **35**, 197–211 (2022)
9. Corley, I., Robinson, C., Dodhia, R., Ferres, J.M.L., Najafirad, P.: Revisiting pretrained remote sensing model benchmarks: resizing and normalization matters. arXiv preprint arXiv:2305.13456 (2023)
10. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
11. Gorishniy, Y., Rubachev, I., Khrulkov, V., Babenko, A.: Revisiting deep learning models for tabular data. Advances in Neural Information Processing Systems **34**, 18932–18943 (2021)
12. Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., He, K.: Accurate, large minibatch sgd: Training imagenet in 1 hour. arXiv preprint arXiv:1706.02677 (2017)
13. Helber, P., Bischke, B., Dengel, A., Borth, D.: EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **12**(7), 2217–2226 (2019)
14. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (2019)
15. Mu, N., Kirillov, A., Wagner, D., Xie, S.: Slip: Self-supervision meets language-image pre-training. In: European Conference on Computer Vision. pp. 529–544. Springer (2022)
16. Neumann, M., Pinto, A.S., Zhai, X., Houlsby, N.: In-domain representation learning for remote sensing. In: International Conference on Learning Representations Workshops. pp. 1–20 (2020)
17. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning. pp. 8748–8763. PMLR (2021)

18. Reed, C.J., Gupta, R., Li, S., Brockman, S., Funk, C., Clipp, B., Keutzer, K., Candido, S., Uyttendaele, M., Darrell, T.: Scale-mae: A scale-aware masked autoencoder for multiscale geospatial representation learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4088–4099 (2023)
19. Stewart, A.J., Robinson, C., Corley, I.A., Ortiz, A., Ferres, J.M.L., Banerjee, A.: Torchgeo: deep learning with geospatial data. In: Proceedings of the 30th international conference on advances in geographic information systems. pp. 1–12 (2022), `https://github.com/microsoft/torchgeo`
20. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International conference on machine learning. pp. 10347–10357. PMLR (2021)
21. Wang, Q., Liu, S., Chanussot, J., Li, X.: Scene classification with recurrent attention of vhr remote sensing images. IEEE Transactions on Geoscience and Remote Sensing **57**(2), 1155–1167 (2018)
22. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3733–3742 (2018)
23. Yang, Y., Newsam, S.: Bag-of-visual-words and spatial extensions for land-use classification. In: Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems. pp. 270–279 (2010)
24. Zhang, X., Lv, Y., Yao, L., Xiong, W., Fu, C.: A new benchmark and an attribute-guided multilevel feature representation network for fine-grained ship classification in optical remote sensing images. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **13**, 1271–1285 (2020)
25. Zheng, C., Shrivastava, A., Owens, A.: Exif as language: Learning cross-modal associations between images and camera metadata. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6945–6956 (2023)
26. Zhu, X.X., Hu, J., Qiu, C., Shi, Y., Kang, J., Mou, L., Bagheri, H., Haberle, M., Hua, Y., Huang, R., et al.: So2sat lcz42: A benchmark data set for the classification of global local climate zones [software and data sets]. IEEE Geoscience and Remote Sensing Magazine **8**(3), 76–89 (2020)