MULTI-SPECTRAL REMOTE SENSING IMAGE RETRIEVAL USING GEOSPATIAL FOUNDATION MODELS

Benedikt Blumenstiel IBM Research Europe

Viktoria Moor IBM Research Europe

Romeo Kienzler IBM Research Europe

Thomas Brunschwiler IBM Research Europe

ABSTRACT

Image retrieval enables an efficient search through vast amounts of satellite imagery and returns similar images to a query. Deep learning models can identify images across various semantic concepts without the need for annotations. This work proposes to use Geospatial Foundation Models, like Prithvi, for remote sensing image retrieval with multiple benefits: i) the models encode multi-spectral satellite data and ii) generalize without further fine-tuning. We introduce two datasets to the retrieval task and observe a strong performance: Prithvi processes six bands and achieves a mean Average Precision of 97.62% on BigEarthNet-43 and 44.51% on ForestNet-12, outperforming other RGB-based models. Further, we evaluate three compression methods with binarized embeddings balancing retrieval speed and accuracy. They match the retrieval speed of much shorter hash codes while maintaining the same accuracy as floating-point embeddings but with a 32-fold compression. The code is available at `https://github.com/IBM/remote-sensing-image-retrieval`.

***Index Terms*—** Multi-spectral, Image retrieval, Geospatial foundation model, Similarity search

1. INTRODUCTION

Remote sensing image retrieval has become increasingly essential in geospatial data analysis, with its potential applications extending across meteorology [1], economic assessment [2], and ecological analysis [3]. In recent years, machine learning enabled a shift from traditional metadata-based retrieval methods to content-based image retrieval (CBIR) [4]. CBIR focuses on the intrinsic features within a query image. This enables the retrieval of potentially any semantic concept without requiring pre-defined annotations in metadata.

Co-funded by the European Union (Horizon Europe, Embed2Scale, 101131841). The corresponding author can be reached via email at benedikt.blumenstiel@ibm.com.

Central to CBIR are retrieval speed and accuracy. While accuracy ensures the relevance of the retrieved images to the query, speed is crucial for efficient processing in large-scale databases. To balance these metrics, compression techniques, such as hash functions, are used to reduce memory requirements and increase retrieval speed. Promising works have combined pre-trained vision models with deep hash networks [5, 6, 7, 8]. Each image is represented by the model embedding, which is compressed into a hash code. However, these models only process RGB data, overlooking the potential of multi-spectral information in satellite imagery.

Geospatial Foundation Models (GeoFM), for instance Prithvi [9], open new possibilities because they are pre-trained on a vast amount of multi-spectral data. The models have been successfully applied to various tasks, including flood and wildfire segmentation [9]. GeoFMs can utilize multi-spectral data for CBIR without requiring fine-tuning.

Our contributions are threefold: i) We showcase the application of GeoFMs for remote sensing image retrieval, ii) we introduce baselines for two multi-spectral datasets to benchmark multi-spectral remote sensing image retrieval, and iii) we conduct a detailed analysis of retrieval performance between vector-based and binary hash-based approaches, focusing on the balance between speed and accuracy.

2. RELATED WORK

Image retrival in remote sensing has evolved significantly, shifting focus from traditional metadata-based methods to more advanced techniques that use computer vision to analyze the image content [4]. Among these developments, deep hash networks have played a central role [5, 8, 10, 11]. These models compress images into hash codes which are stored in a database and used for calculating distances. Researchers explored various learning techniques, such as contrastive learning [8] and metric learning [12], with the aim of minimizing the need for extensive training data.

Some approaches use the embeddings of existing computer vision models and combine them with smaller hash models [5, 6, 12]. Because these models are trained on RGB data, they cannot take advantage of all multi-spectral

data from satellites. Furthermore, the fine-tuning of hash networks on specific datasets limits their transferability to different semantic concepts, potentially reducing accuracy in varying contexts beyond their initial training datasets.

Despite the progress in various CBIR techniques, most evaluations are primarily focused on RGB datasets, i.e., UCM [13] and AID [14] being the *de-facto* standard evaluation [5, 6, 8, 10, 11, 12, 15, 16]. An exception is works on cross-modal image retrieval [4]. E.g., one study uses a subset of BigEarthNet [17], which includes multi-spectral data [7].

GeoFMs, such as Prithvi and Presto [18], are characterized by their ability to process multi-spectral data. Their pre-training on diverse satellite images overcomes the limitations of general models and benefits earth observation tasks [9].

In conclusion, while existing approaches have advanced the field, they are focused on RGB datasets and depend on annotated training data. These limitations emphasize the need for new approaches utilizing GeoFMs and multi-spectral data.

3. APPROACH

The CBIR task requires identifying and retrieving images from a large database based on their visual similarity to a given query image. This process requires an efficient mechanism to compare and rank the database images in terms of their similarity to the query.

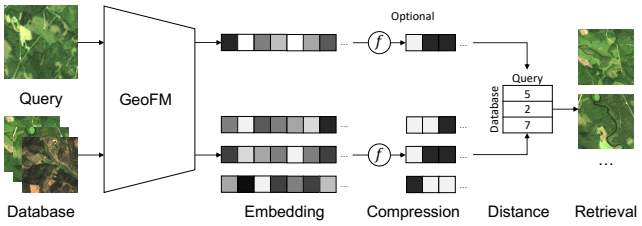


Fig. 1: GeoFM embeddings enable simple but accurate CBIR. Optionally, the embeddings are compressed into smaller binary vectors. For each query image, similar images from the database are returned and sorted based on a distance function.

In our method, as illustrated in Figure 1, the first step involves processing the query images through a GeoFM. The model generates embedding vectors that comprehensively represent latent features of the images. The query embedding is compared with the pre-computed embeddings stored in a vector database. The similarity between query vectors and database vectors is computed using common distance functions like Hamming, Jaccard, Euclidean or Cosine distance. This approach takes advantage of the robust pre-training of the underlying GeoFM, which improves the accuracy and generalizability of the retrieval results. Optionally, the vector embeddings are compressed into shorter binary vectors using existing methods like hash functions. This reduces memory usage and inference time but can lead to information loss.

The simplest compression is the binarization of the embedding vector through a sign function. This reduces the memory usage by a factor of 32. For further compression, we propose a trivial hash function as a simple baseline: The embeddings are split into an equally sized number of dimensions and averaged to reduce the vector to the hash length. We then apply the sign function to generate binary hash codes. This approach assumes an equal distribution of information across the dimensions and around zero within each dimension.

4. EXPERIMENTAL SETUP

Following literature [5, 6, 8, 10, 11, 12, 15, 16], we evaluate our experiments with mean Average Precision (mAP) based on the top 20 retrieved images. For multi-label datasets, any overlap within the labels is counted as a positive match [7, 19]. Each validation split serves as test queries, and the test split is used as the database. We use different splits to avoid a geographical overlap between the queries and the database[1]. Further, we use the L1 norm as a distance function in our experiments, which is equal to the hamming distance for binary values. We also tested the L2 norm and observed minimal differences within $\pm$ 1 pp. mAP.

We use Milvus [20] for our speed experiments. Milvus is a production-ready vector database and includes a search functionality based on binary and floating-point vectors. The L2 norm is used for float vectors, as L1 is only available for binary vectors. Milvus indexes[2] include a cluster-based approach: First, the query is compared to the cluster centers, followed by a comparison with the images in the top clusters. This drastically reduces the retrieval time.

4.1. Models

The underlying GeoFM is Prithvi-100M with a model input of 224x224 pixels. The model is a Vision Transformer (ViT) [21] with 100M parameters. Prithvi processes six bands, unlike the three RGB channels of the vanilla ViT. The pre-training includes a subset of the Harmonized Landsat-Sentinel (HLS) dataset consisting of images from the Landsat 8 and Sentinel-2 satellites. We also report the results of

[1]Other works iterate over the test split as query images, using the remaining test images as the database [6, 12]. This is appropriate for aerial datasets but not for satellite datasets. The splits are often geographically grouped to avoid similar regional patterns benefiting the model performance. We also find some implementations using train images as part of the queries or database [7, 8, 15, 16]. Using the same images during training and testing can lead to data leakage, which benefits over-fitted models.

Therefore, we select two different splits for the queries and databases. Note that we do not train any model since we use a pre-trained GeoFM. However, we avoid using the train splits to facilitate a fair comparison with potential future works that use trained models. For training, we recommend further splitting the train images and not using the validation splits.

[2]See `https://milvus.io/docs/index.md` for details. We used the indexes INV_FLOAT and BIN_IVF_FLAT with the default values of 128 clusters and 8 top clusters.

Model	Method	BigEarthNet-43	BigEarthNet-19	ForestNet-12	ForestNet-4	Mean
Prithvi-100M	Embedding	**97.62**	**97.98**	**44.51**	**60.76**	**75.22**
	Binary emb.	<u>97.44</u>	<u>97.83</u>	<u>43.28</u>	<u>59.85</u>	<u>74.6</u>
	Trivial hash	82.75	84.26	34.91	51.71	63.41
	LSH	68.36 ± 4.96	71.18 ± 4.66	28.98 ± 1.85	45.63 ± 1.39	53.47
Prithvi-100M-RGB	Embedding	92.15	93.17	38.65	53.85	69.46
	Binary emb.	91.38	92.43	38.11	53.31	68.81
	Trivial hash	73.36	75.47	32.35	50.47	57.91
	LSH	53.09 ± 8.43	55.21 ± 8.61	29.02 ± 1.45	45.42 ± 1.26	45.65
ViT-B/16-RGB	Embedding	89.31	90.21	38.92	56.49	68.73
	Binary emb.	88.71	89.7	39.19	57.01	68.65
	Trivial hash	75.74	77.34	33.01	49.95	59.01
	LSH	76.52 ± 0.95	78.02 ± 0.92	32.77 ± 1.32	49.13 ± 2.09	59.08

Table 1: mAP@20 results for all models and datasets. We highlight the best-performing method in bold and underline the second-best one. LSH is reported with a 95% confidence interval based on five seeds as this method uses random planes.

Prithvi only processing the RGB channels (Prithvi-100M-RGB). The model input of the infrared channels is set to zero. For comparison, we include the vanilla ViT-B/16 model (ViT-B/16-RGB) [21], which is pre-trained on ImageNet.

We run the experiments with the model embeddings of size 768, binary embeddings, and two different hash encodings: The trivial hash function and Locality-Sensitive Hashing (LSH) [22]. LSH splits the latent space into two areas for each binary value using a random hyperplane. Both hash methods use a hash length of 32 to balance compression and accuracy. We do not vary the length in the reported results as other works already studied the effect [5, 6, 12]. Generally, decreasing the hash length reduces the precision.

4.2. Datasets

Reviewing remote sensing datasets revealed a limited availability of multi-spectral multi-class datasets with an image size of equal or more than 224 pixels [4, 23, 24]. BigEarthNet [17] and ForestNet [25] fulfill these requirements.

BigEarthNet consists of Sentinel-1 and -2 images with 120x120 pixels and includes two sets of multi-label annotations with 19 resp. 43 classes. The classes cover different land-use types such as *mixed forest*, *water bodies*, or *airports*. We are using the six Sentinel-2 channels supported by Prithvi. The images are bi-linear scaled up to match the model input, which corresponds to a five-meter resolution.

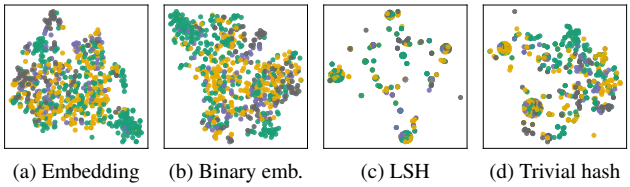
ForestNet includes Landsat 8 imagery from forest loss events. The images are annotated with twelve classes and four super-classes. The classes indicate types of deforestation, such as *timber plantation* or *small-scale agriculture*. ForestNet uses composite images created by averaging up to five cloud-free images. The images have a size of 332x332 and a 22-meter resolution after re-scaling.

Note that the data of both datasets differs from the Prithvi pre-training: The HLS data has a 30-meter resolution and includes additional data processing for harmonization. Furthermore, Prithvi-100M is pre-trained on US data and does not include any data from BigEarthNet and ForestNet which cover Europe and Indonesia, resp. [9, 17, 25].

5. RESULTS

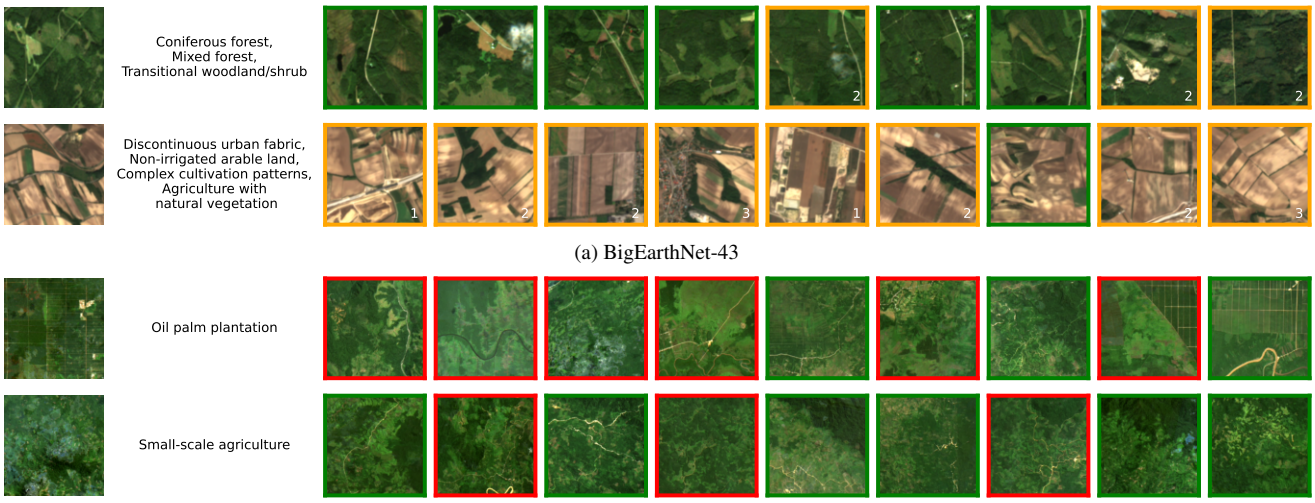
Our results are presented in Table 1. Prithvi-100M outperforms Prithvi-100M-RGB in every method by over 5.5 pp. on average. The differences for hash methods are significant for the BigEarthNet dataset, while the performance is comparable on ForestNet. The comparison of Prithvi-100M with the vanilla ViT-B/16-RGB leads to similar results, except a better-performing LSH method. LSH seems better suited for the vanilla ViT as it performs 5.61 pp. better than Prithvi-100M and on pair with the trivial hash method. Overall, the comparison highlights the information gain when using multi-spectral data with GeoFMs.

The transition from float embeddings to hash-based approaches typically results in a notable drop in model performance. E.g., Prithvi-100M with a 32-bit trivial hash has an 11.81 pp. lower mAP. This drop highlights the challenge of maintaining accuracy while simplifying the data representations. However, we observe that the performance loss is mainly influenced by the hash length and not by binarization.



(a) Embedding (b) Binary emb. (c) LSH (d) Trivial hash

Fig. 2: t-SNE plots of the ForestNet-4 test set with colored classes comparing the embedding space with hash codes.



(a) BigEarthNet-43

(b) ForestNet-12

Fig. 3: Examples from two datasets with query images (left), their labels, and retrieved images (right) using Prithvi and the trivial hash method. Images with green frames indicate positive matches, while those with red frames have different labels. Orange shows partial correct matches, where the number represents the number of label matches within the multi-labels.

The binary embeddings perform nearly as well as the floating-point embeddings, with an average difference of only 0.62 pp. mAP for Prithvi.

We visually compare the ForestNet-4 embeddings and hash functions with t-SNE plots in Figure 2. The latent space is well distributed for the default and binarized embeddings but clustered with LSH. The clusters do not discriminate the semantic classes, which explains the lower performance. In comparison, trivial hash codes also include some larger clusters but retain the overall distribution better.

Figure 3 shows examples of retrieved images based on the trivial hashes. Retrieved images from BigEarthNet-43 have a large share of images with partial label matches, which is also represented by a 82.75 mAP. Those images are often similar in their overall color and appearance. However, some classes are less frequently matched, e.g., specific infrastructure classes like *construction sites* or *airports*. This is also observable in ForestNet-12, which has more fine-grained classes that are harder to differentiate. Still, the trivial approach returns some matching examples for most queries.

Our analysis demonstrated the effect of compression on the accuracy. Next, we discuss the effect on retrieval speed. The overall retrieval speed depends on three factors: i) model inference, ii) similarity search, and iii) data loading. The first step mainly depends on the model used for the encoding. We observed an inference time of 100 ms for Prithvi on a NVIDIA V100 GPU. Using smaller GeoFMs can reduce the processing time. The last step depends on the specific hardware and is not affected by the retrieval approach. Therefore, we focus our speed analysis on the actual retrieval computation. We used a VM with 12 cores and 24 Gb of memory on an AMD EPYC 7452 processor for our experiments and provide the results in Table 2.

		Images in database		
Data type	Length	10K	50K	100K
Binary	32	16 ms	16 ms	17 ms
Binary	768	16 ms	16 ms	17 ms
Float	768	21 ms	32 ms	33 ms

Table 2: Experimental retrieval speeds with different vector types for a varying number of images in the database.

The retrieval speed for binary vectors is almost not influenced by the database size or the vector length. Floating-point embeddings have a retrieval time of up to 33 ms. This is two times longer than binary embeddings with 17 ms. The results demonstrate the efficient implementation of retrieval algorithms in vector databases like Milvus. Therefore, the hash length is mainly constrained by memory usage rather than the retrieval speed.

6. CONCLUSION

This work demonstrates the applicability of GeoFMs for image retrieval in remote sensing. Due to the learned representations through the pre-training, the model encodes multiple semantics and does not require further fine-tuning. We introduce two multi-spectral datasets to the retrieval task and provide strong baselines, enabling a more holistic evaluation for remote sensing in the future. Accordingly, we could evaluate three compression methods with binary embeddings, having shown the best trade-off between accuracy and retrieval speed. The approach can be easily implemented in various applications with any existing GeoFM and combined with more advanced compression methods to improve performance.

7. REFERENCES

[1] Fabio Dell'Acqua and Paolo Gamba, "Query-by-shape in meteorological image archives using the point diffusion technique," *IEEE transactions on geoscience and remote sensing*, vol. 39, no. 9, pp. 1834–1843, 2001.

[2] Jan De Leeuw, Anton Vrieling, Apurba Shee, Clement Atzberger, Kiros M Hadgu, Chandrashekhar M Biradar, Humphrey Keah, and Calum Turvey, "The potential and uptake of remote sensing in insurance: A review," *Remote Sensing*, vol. 6, no. 11, pp. 10888–10912, 2014.

[3] O James Reichman, Matthew B Jones, and Mark P Schildhauer, "Challenges and opportunities of open data in ecology," *Science*, vol. 331, no. 6018, pp. 703–705, 2011.

[4] Yansheng Li, Jiayi Ma, and Yongjun Zhang, "Image retrieval from remote sensing big data: A survey," *Information Fusion*, vol. 67, pp. 94–115, 2021.

[5] Weiwei Song, Zhi Gao, Renwei Dian, Pedram Ghamisi, Yongjun Zhang, and Jón Atli Benediktsson, "Asymmetric hash code learning for remote sensing image retrieval," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2022.

[6] Subhankar Roy, Enver Sangineto, Begüm Demir, and Nicu Sebe, "Metric-learning-based deep hashing network for content-based retrieval of remote sensing images," *IEEE Geoscience and Remote Sensing Letters*, vol. 18, no. 2, pp. 226–230, 2020.

[7] Gencer Sumbul, Markus Müller, and Begüm Demir, "A novel self-supervised cross-modal image retrieval method in remote sensing," in *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp. 2426–2430.

[8] Xu Tang, Yuqun Yang, Jingjing Ma, Yiu-Ming Cheung, Chao Liu, Fang Liu, Xiangrong Zhang, and Licheng Jiao, "Meta-hashing for remote sensing image retrieval," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–19, 2022.

[9] Johannes Jakubik, Sujit Roy, CE Phillips, Paolo Fraccaro, Denys Godwin, Bianca Zadrozny, Daniela Szwarcman, Carlos Gomes, Gabby Nyirjesy, Blair Edwards, et al., "Foundation models for generalist geospatial artificial intelligence," *arXiv preprint arXiv:2310.18660*, 2023.

[10] Dongjie Zhao, Yaxiong Chen, and Shengwu Xiong, "Multi-scale context deep hashing for remote sensing image retrieval," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2023.

[11] Chao Liu, Jingjing Ma, Xu Tang, Fang Liu, Xiangrong Zhang, and Licheng Jiao, "Deep hash learning for remote sensing image retrieval," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 4, pp. 3420–3443, 2020.

[12] Subhankar Roy, Enver Sangineto, Begüm Demir, and Nicu Sebe, "Deep metric and hash-code learning for content-based retrieval of remote sensing images," in *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2018, pp. 4539–4542.

[13] Yi Yang and Shawn Newsam, "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*, 2010, pp. 270–279.

[14] Gui-Song Xia, Jingwen Hu, Fan Hu, Baoguang Shi, Xiang Bai, Yanfei Zhong, Liangpei Zhang, and Xiaoqiang Lu, "Aid: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 3965–3981, 2017.

[15] Mengluan Huang, Le Dong, Weisheng Dong, and Guangming Shi, "Supervised contrastive learning based on fusion of global and local features for remote sensing image retrieval," *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

[16] Yansheng Li, Yongjun Zhang, Xin Huang, Hu Zhu, and Jiayi Ma, "Large-scale remote sensing image retrieval by deep hashing neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 2, pp. 950–965, 2017.

[17] Gencer Sumbul, Marcela Charfuelan, Begüm Demir, and Volker Markl, "Bigearthnet: A large-scale benchmark archive for remote sensing image understanding," in *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2019, pp. 5901–5904.

[18] Gabriel Tseng, Ivan Zvonkov, Mirali Purohit, David Rolnick, and Hannah Kerner, "Lightweight, pre-trained transformers for remote sensing timeseries," *NeurIPS 2023 Workshop on Tackling Climate Change with Machine Learning: Blending New and Existing Knowledge Systems*, 2023.

[19] Qin Zou, Ling Cao, Zheng Zhang, Long Chen, and Song Wang, "Transductive zero-shot hashing for multilabel image retrieval," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 4, pp. 1673–1687, 2020.

[20] Jianguo Wang, Xiaomeng Yi, Rentong Guo, Hai Jin, Peng Xu, Shengjun Li, Xiangyu Wang, Xiangzhou Guo, Chengming Li, Xiaohai Xu, et al., "Milvus: A purpose-built vector data management system," in *Proceedings of the 2021 International Conference on Management of Data*, 2021, pp. 2614–2627.

[21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2020.

[22] Yi-Kun Tang, Xian-Ling Mao, Yi-Jing Hao, Cheng Xu, and Heyan Huang, "Locality-sensitive hashing for finding nearest neighbors in probability distributions," in *Social Media Processing: 6th National Conference, SMP 2017, Beijing, China, September 14-17, 2017, Proceedings*. Springer, 2017, pp. 3–15.

[23] Zhitong Xiong, Fahong Zhang, Yi Wang, Yilei Shi, and Xiao Xiang Zhu, "Earthnets: Empowering ai in earth observation," *arXiv preprint arXiv:2210.04936*, 2022.

[24] Alexandre Lacoste, Nils Lehmann, Pau Rodriguez, Evan David Sherwin, Hannah Kerner, Björn Lütjens, Jeremy Andrew Irvin, David Dao, Hamed Alemohammad, Alexandre Drouin, et al., "Geo-bench: Toward foundation models for earth monitoring," *Advances in Neural Information Processing Systems*, 2023.

[25] Jeremy Irvin, Hao Sheng, Neel Ramachandran, Sonja Johnson-Yu, Sharon Zhou, Kyle Story, Rose Rustowicz, Cooper Elsworth, Kemen Austin, and Andrew Y Ng, "Forestnet: Classifying drivers of deforestation in indonesia using deep learning on satellite imagery," *NeurIPS 2020 workshop on Tackling Climate Change with Machine Learning*, 2020.