Self-Supervised Learning for Multi- and Hyperspectral Foundation Models

[1]Behnood Rasti and [2]Nassim Ait Ali Braham

[1]Technische Universität Berlin (TUB) & Berlin Institute for the Foundations of Learning and Data (BIFOLD)

[2]German Aerospace Center (DLR) & Technical University of Munich (TUM)

BIFOLD

IEEE GRSS IADF Image Analysis and Data Fusion

TECHNISCHE UNIVERSITÄT BERLIN

Technische Universität München

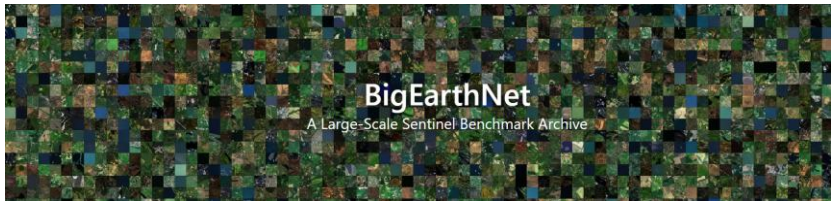
DLR Deutsches Zentrum für Luft- und Raumfahrt

Outline

- Introduction to SSL and Foundation Models
- SSL Models
  - Joint Embedding SSL
  - Masked Image Modeling
- Geospatial Foundation Models
  - RGB/Multispectral FMs
  - Hyperspectral FMs
  - Multi-sensor/modal FMs
  - Evaluating Geo FMs
- Conclusion
- Hands-on

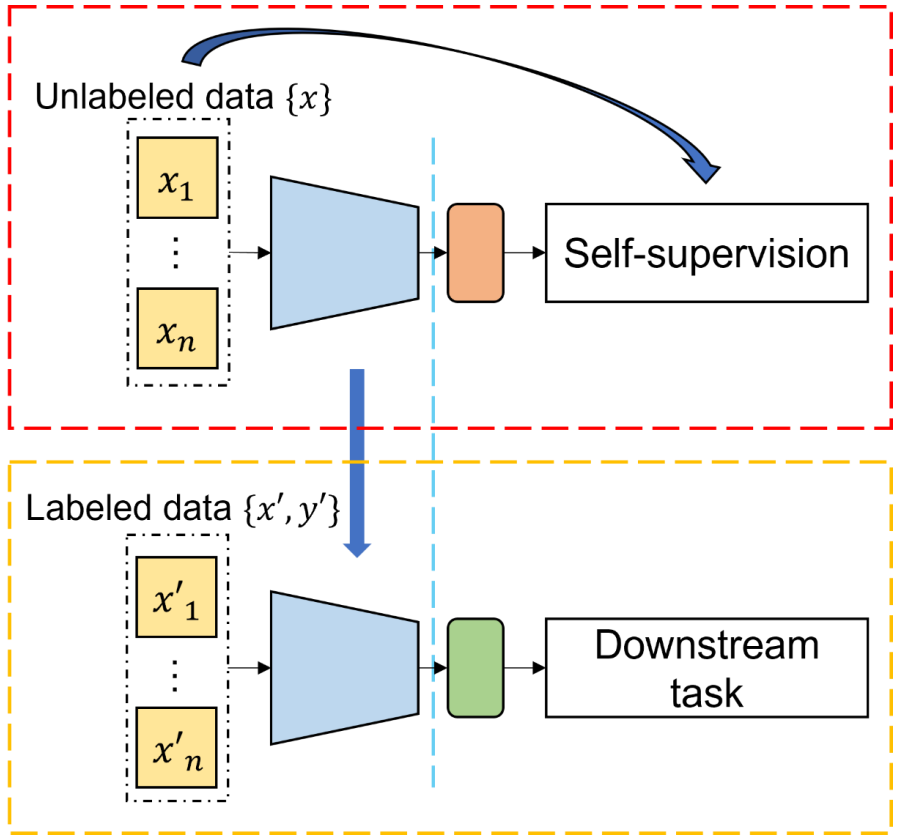
Why Self-Supervised Learning (SSL)?

- Deep Learning often requires considerable amount of annotated data
- Scarcity of labeled data
  - Costly to obtain
  - Tedious annotation process
- Unlabeled data is abundant
  - Satellite archives with Petabytes of data
- Transferability: A pre-trained model can be transferred to downstream tasks



What is Self-supervised Learning (SSL)?

- Pretrain a model from intrinsic features of data
  - Spectral
  - Spatial
  - Temporal
- How can we pretrain a model without labeled Data?

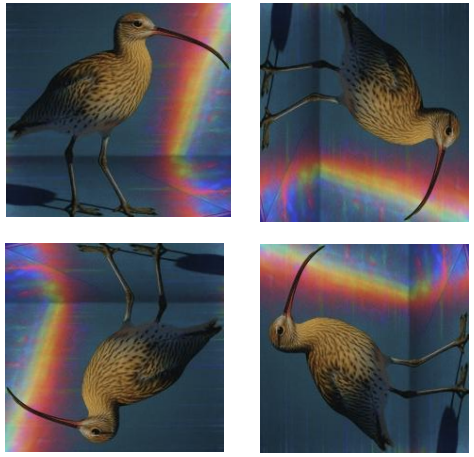


A pre-trained model can be transferred to downstream tasks

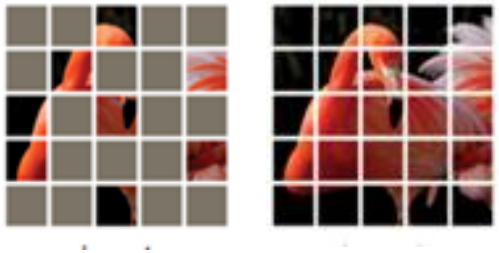
Pretext Tasks for Self-Supervised Learning

- Pretext tasks: Apply a transformation to the data→ The network predicts how the data was transformed
- Enable models to learn meaningful representations from unlabeled data

Rotation



Inpainting

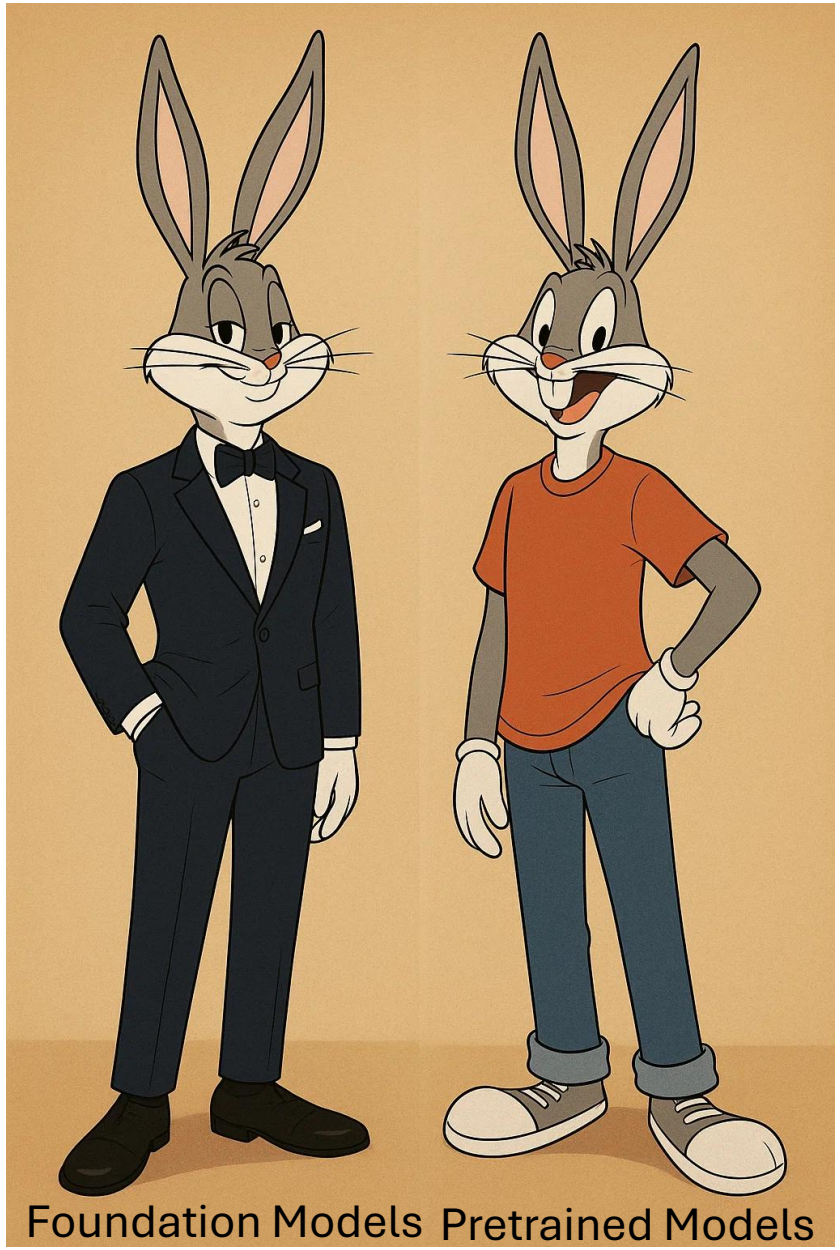


Rotation + Color Distortion

Jigsaw Puzzle Solving

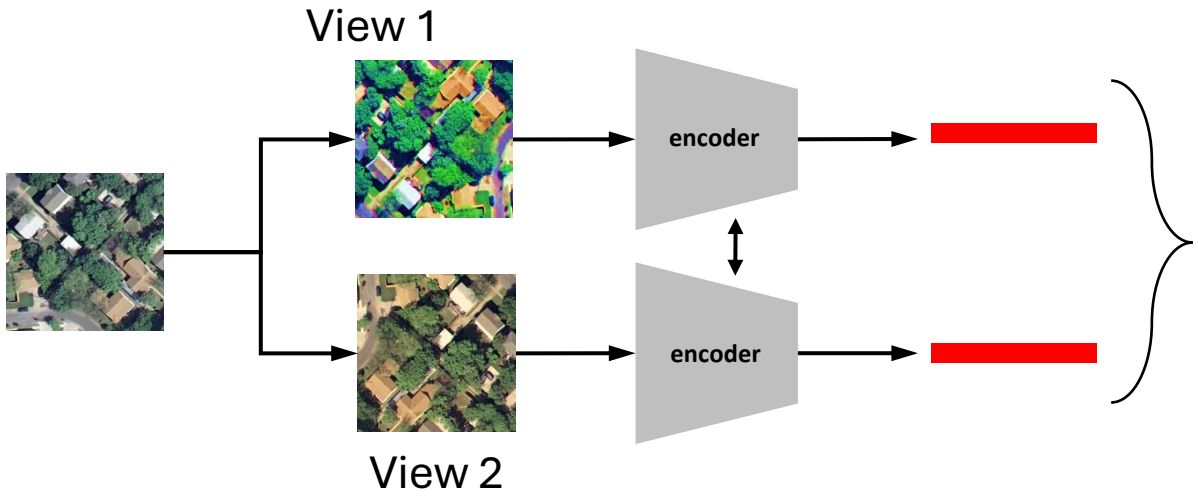
What is a Foundation Model?

- Foundation Model (FM): A (large-scale AI pretrained) model that gives us a foundation to do further tasks
- An FM is often trained on big unlabeled data using SSL techniques
- Geospatial FMs: large-scale AI models that are pretrained on geospatial data and can be adapted for a wide range of downstream Earth observation and spatial analysis tasks

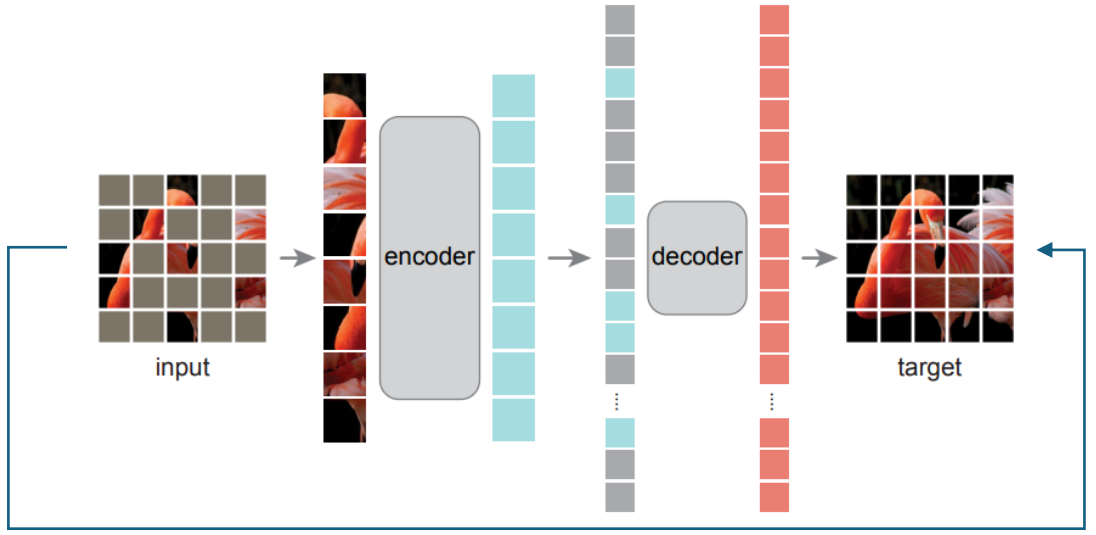


Three Main Paradigms for Self-Supervised Learning

1. Joint Embedding



2. Masked Image Modeling (MIM)

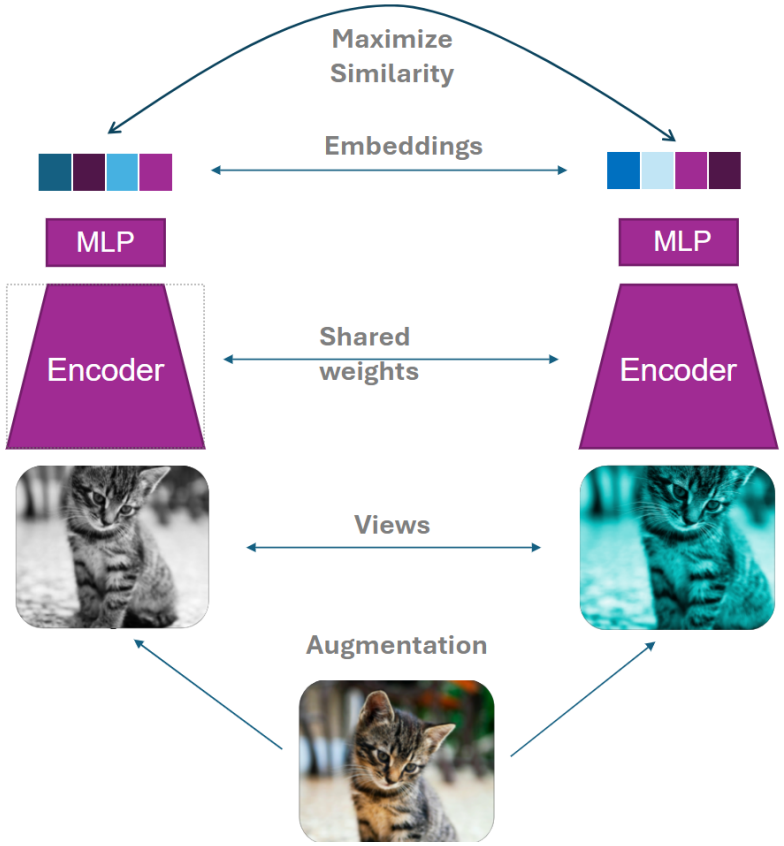


3. Hybrid (Masking+ Joint Embedding)

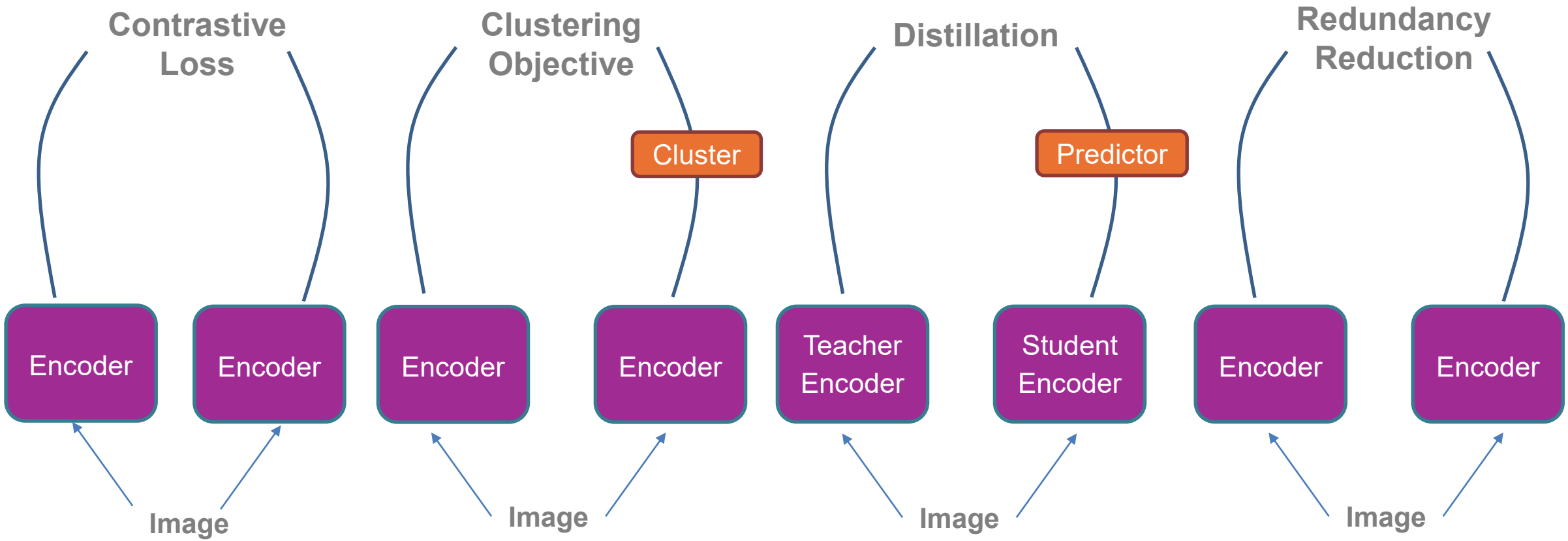
He, K., et al., "Masked autoencoders are scalable vision learners." *IEEE/CVF conference on computer vision and pattern recognition*, 2022.

Joint Embedding

- General idea
  - Siamese architecture with shared parameters
  - Similar images (views) are generated using **data augmentation**
  - Enforce **invariance** to the augmentations
- **Problem:** Collapsing means all inputs map to (almost) the same vector
- **Solution**?



Self Supervised Learning Methods: Joint Embedding



Contrastive Methods:Pull positive pairs close in the embedding space, push apart negatives.

- SimCLR
- Moco

Clustering Methods: Learn embeddings by grouping similar samples into clusters without explicit negatives

- SwAV

Distillation Methods: A "student" encoder matches the output distribution of a "teacher" encoder across augmentations, often with EMA-updated teacher.

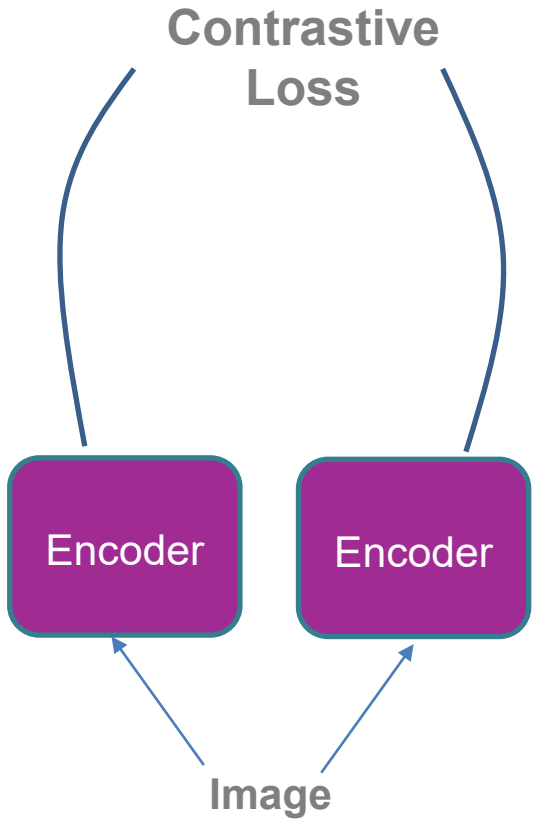
- BYOL
- DINO

Regularization Methods: Encourage embeddings to be decorrelated across dimensions.

- Barlow Twins
- VICReg

Contrastive Methods

- Core idea: Pull positive pairs (different views/augmentations of the same image) close in the embedding space, push apart negatives (different images).
- Contrastive loss
- Key property: Requires negative samples or large batch sizes
- Good for multimodal data (e.g., CLIP)
- Examples: SimCLR, MoCo

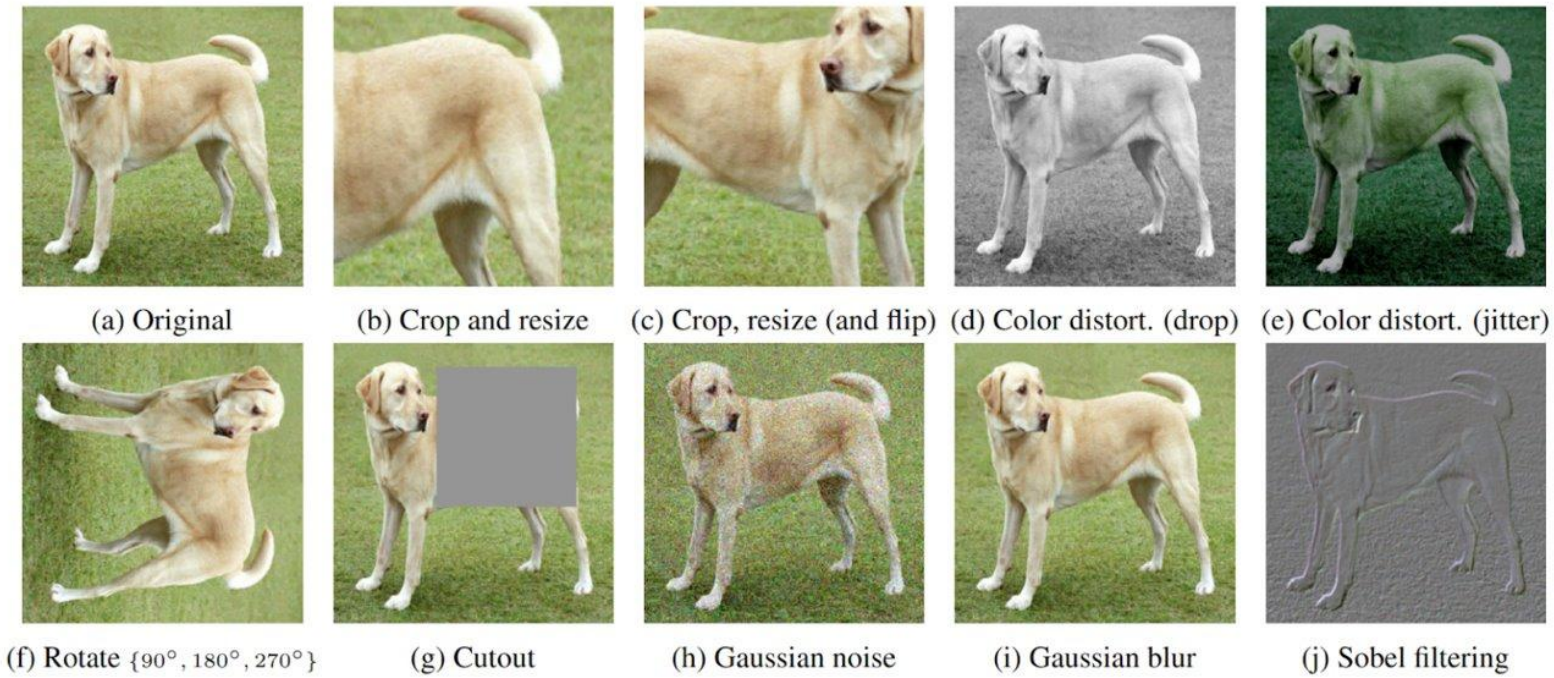


Contrastive Methods

- SimCLR
- Moco

Common Data Augmentation

- Random Crop and Resize (Two very important transforms)



T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," in *Proc. Int. Conf. Machine Learning (ICML)*, PMLR, 2020.

SimCLR

- Core idea: Learn by pulling two augmented views of the same image together and pushing them away from all other images in the batch (negatives).
- Two views x_1, x_2 →shared encoder f→ features y→ MLP projector g→ $z = \frac{g(y)}{\|g(y)\|_2} \in \mathbb{R}^D$.
- **NT-Xent loss**: For a batch of B images → $2B$ views and normalized embedding $\{z_k\}_{k=1}^{2B}$. For a positive pair (i, j) (two views of the same image):

$$\ell_{i,j} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1,\, k\neq i}^{2B} \exp(\text{sim}(z_i, z_k)/\tau)}$$

Negative samples

- sim(·,·)= cosine similarity (dot product)
- τ = temperature (controls sharpness of softmax).

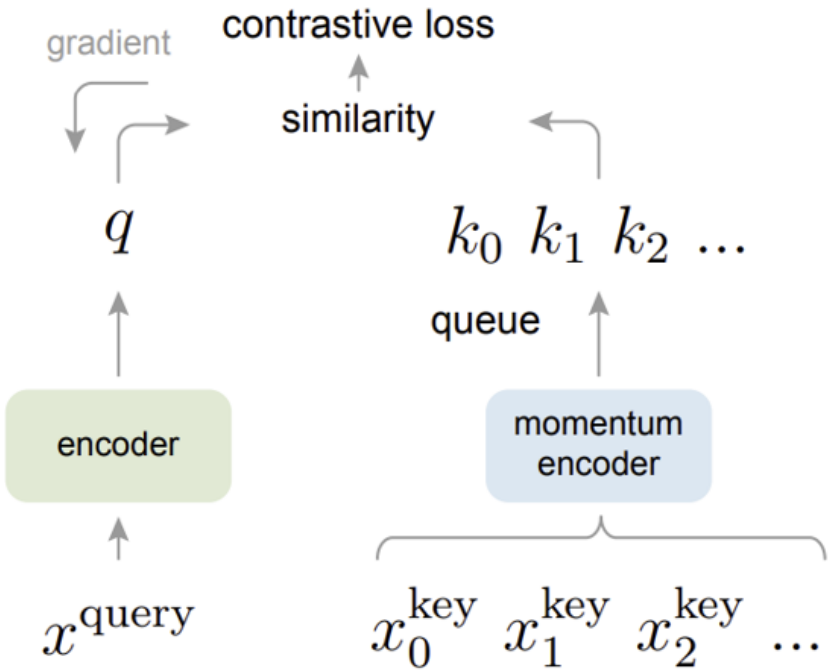
- Pros: Simple and very effective, Cons: Requires a very large batch size, false negatives

T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," in *Proc. Int. Conf. Machine Learning (ICML)*, PMLR, 2020.

MoCo: Momentum Contrast

- **Core idea:** Contrastive learning with a **dynamic dictionary**
- Positive pair: Two augmented views of the same image (query–key)
- **Negative pairs:** Keys from other images in the **queue (dictionary)**
- Query encoder (f_{θ_q}): updated by backpropagation
- Key encoder (f_{θ_k}): updated by *momentum* from query encoder (EMA)

$$\theta_k \leftarrow m\theta_k + (1-m)\theta_q$$



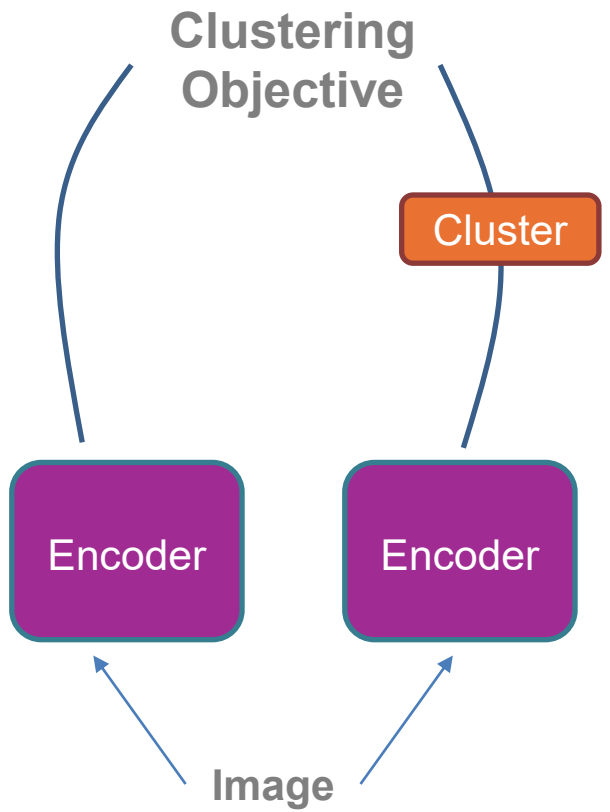
- **InfoNCE loss**: $\mathcal{L}_{\text{InfoNCE}} = -\log \frac{\exp(\text{sim}(q, k^+)/\tau)}{\sum_{i=0}^{N} \exp(\text{sim}(q, k_i)/\tau)}$

- q= query vector (embedding)
- k+= positive key (embedding from same instance)
- ki = negatives (embeddings from other instances+positive).
- Pros **:** Large and consistent set of negatives without needing huge batch sizes.- Cons : Extra Memory & Computation to maintain/ update dictionary
- MoCo v2: Stronger data augmentations + MLP projection head (like SimCLR).
- MoCo v3: Adapts MoCo framework to ViTs, dropping the explicit dictionary, negatives from minibatch.

K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum Contrast for Unsupervised Visual Representation Learning," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 9726–9735, 2020.

Clustering Methods

- Core idea: Learn embeddings by grouping similar samples into clusters without explicit negatives.
- **Key property**: Jointly learns representations and cluster assignments.
- Examples: SwAV, Deep Cluster (not joint Embedding)



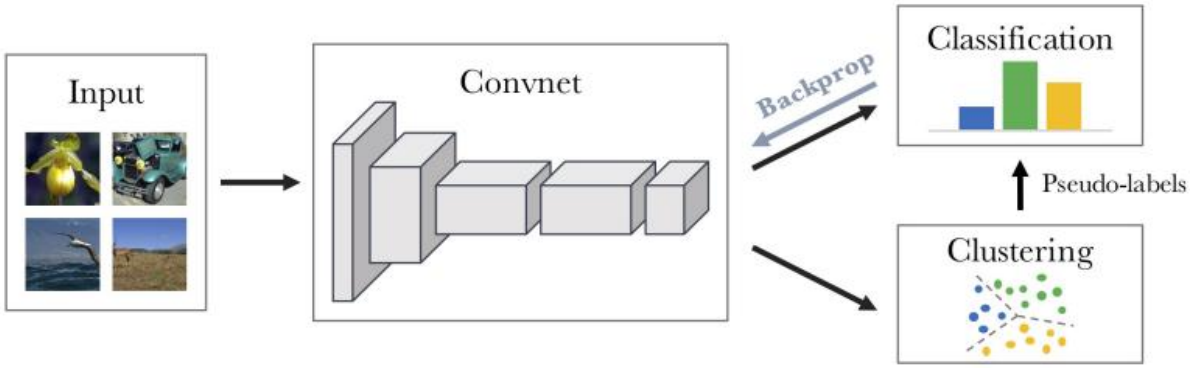
Clustering Methods

Deep Cluster

- Core idea: Learn through clustering

- Deep Cluster (Not joint embedding)
  - Iteratively
    - Cluster the dataset (K-means)
    - Train a classifier on the pseudo labels

  - Pros: Simplicity and effectiveness
  - Cons: Does not scale well with the dataset



Caron, Mathilde, et al. "Deep clustering for unsupervised learning of visual features." *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018.

SwAV

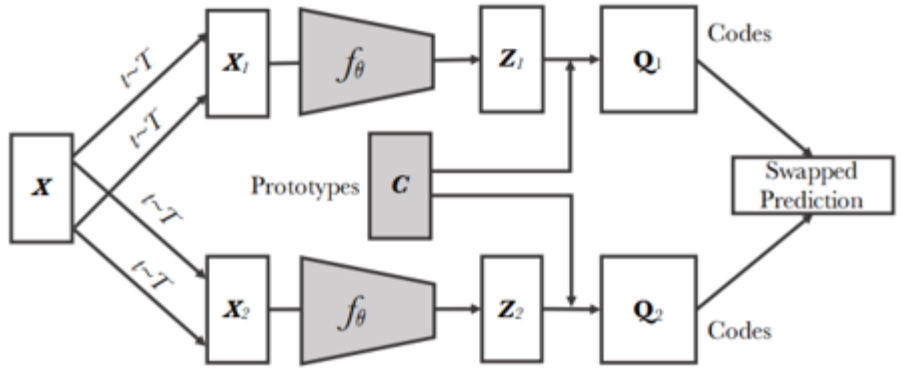
- **SwAV** is based on **Clustering**
- General Principle:
  - Compute « codes » by assigning features to prototypes
  - The codes are the soft cluster-assignment vectors
  - Prototypes are learnt during training

- Swapping assignments between views
  - The **code** q (**online clustering**) from one view is used as the **target** for another view. For two views x_1, x_2
  - Predict p_1 for v_1, target = q_2 (assignment of v_2)
  - Predict p_2 for v_2, target = q_1 (assignment of v_1)

$$\mathcal{L} = \frac{1}{2}\Big(\ell(p_1, q_2) + \ell(p_2, q_1)\Big) \qquad \ell(p, q) = -\sum_{i=1}^{K} q(i) \log p(i)$$

  - K = number of prototypes (cluster centers, $\{c_1, \ldots, c_k\}$).

- Pros: No negative samples- Cons: May collapse

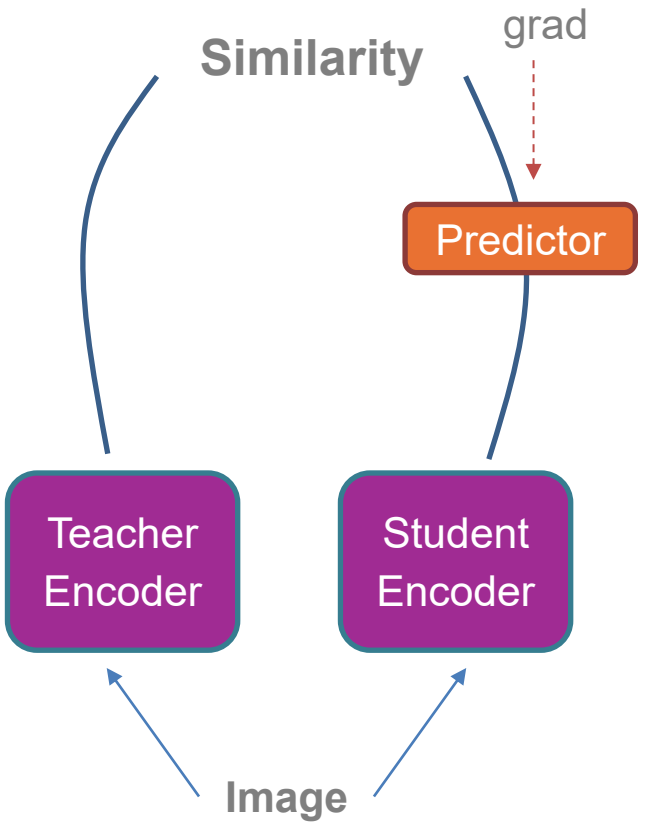


Swapping Assignments between Views

M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, **"Unsupervised Learning of Visual Features by Contrasting Cluster Assignments,"** *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2020, pp. 9912-9924.

Distillation Methods

- Core idea: A "student" network matches the output distribution of a "teacher" network across augmentations, often with a momentum-updated teacher.
- Examples: BYOL, DINO
- Key property: Works without negatives, relies on asymmetric architecture (teacher vs. student) to avoid collapse.
- Pros:
  - Simple and powerful
  - No need for large batches of negative samples
- Cons:
  - More prone to collapsing



Distillation Methods

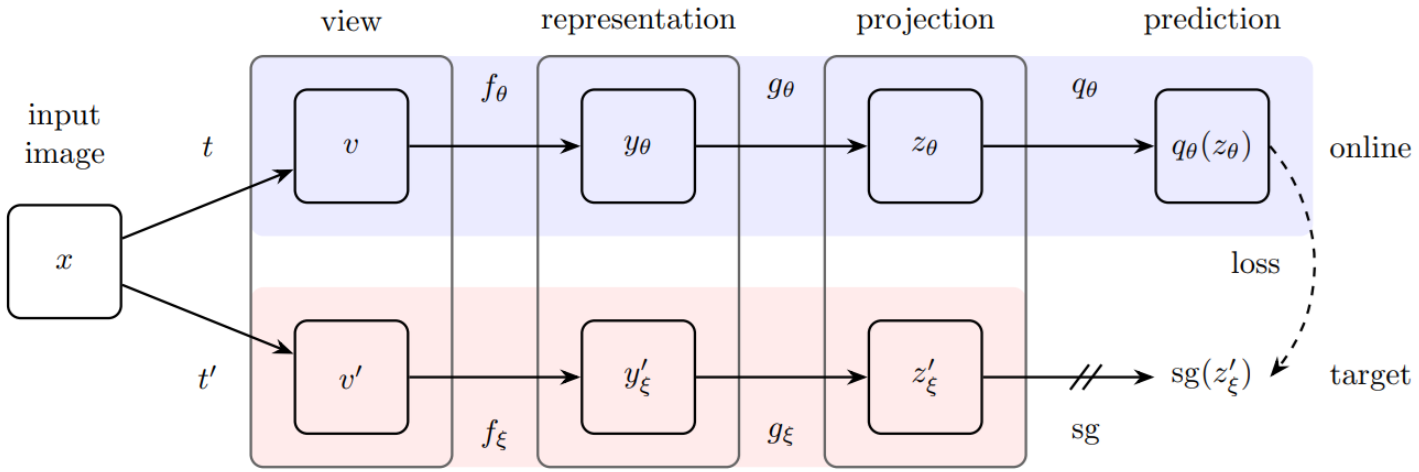
BYOL

- BYOL components
  - A **student encoder (online)**
  - A teacher **momentum encoder (target)**
    - An exponentially moving average (**EMA**) of the values of the student from previous iterations

$$\xi \leftarrow \tau\xi + (1-\tau)\theta$$

$$\tau \in [0, 1]$$

  - A **prediction** MLP head only for the student branch



- Optimize the L_2 norm between the prediction of the student and the teacher

$$L_{\theta,\xi} = \| q_\theta(z_\theta) - z'_\xi \|_2^2$$

Grill, Jean-Bastien, et al. "Bootstrap Your Own Latent: A new approach to self-supervised learning." *Neural Information Processing Systems*. 2020.

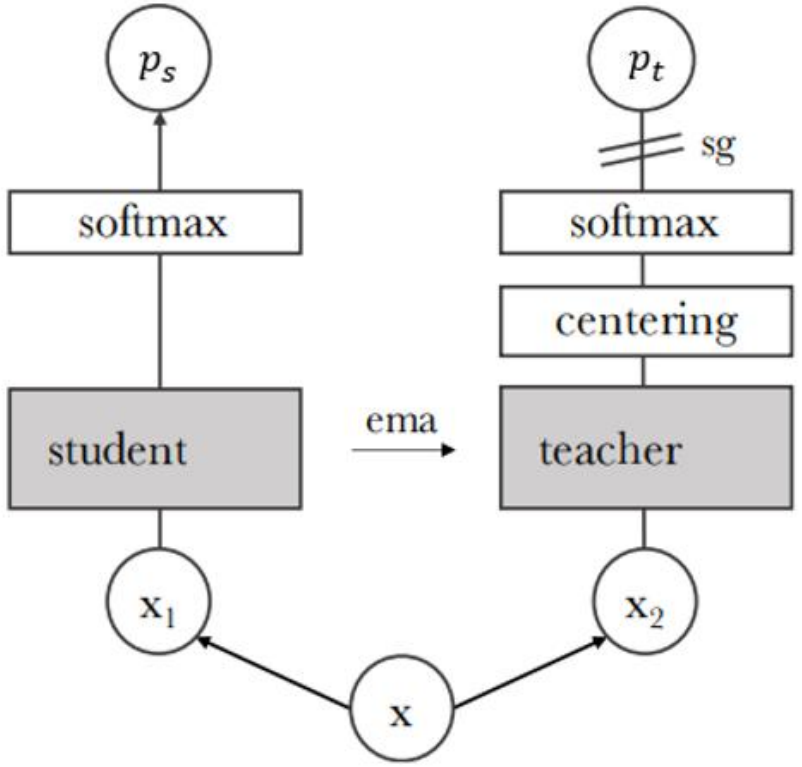
DINO: Self-Distillation without Labels

- **Core idea:** Self-distillation between student and teacher encoders.
- Teacher updated via **EMA** of student.
- Two augmented views, x_1, x_2
- **Student encoder f_θ** + projection head g_θ → produces probability distribution $p_s \in R^K$
- **Teacher encoder f_ζ** + projection head g_ζ → produces probability distribution $p_t \in R^K$
- Introduced **sharpened teacher outputs (**with low temperature T_t**)** + centering → avoids collapse.

$$p_t = \text{softmax}\left(\frac{g_{\theta_t}(f_{\theta_t}(x_2)) - c}{T_t}\right), \quad p_s = \text{softmax}\left(\frac{g_{\theta_s}(f_{\theta_s}(x_1))}{T_s}\right)$$

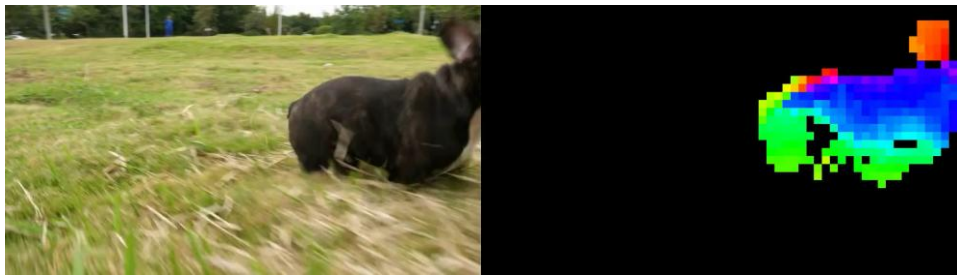
- The center c is updated $c \leftarrow mc + (1-m)\frac{1}{B}\sum_{i=1}^{B} g_{\theta_t}(x_i),$
- Loss: cross-entropy

$$\mathcal{L}(x_1, x_2) = -\sum_{i=1}^{K} p_t(i) \log p_s(i),$$



M. Caron et al., "Emerging Properties in Self-Supervised Vision Transformers," Proc. IEEE Int. Conf. Comput. Vis. (ICCV), pp. 9630–9640, 2021.

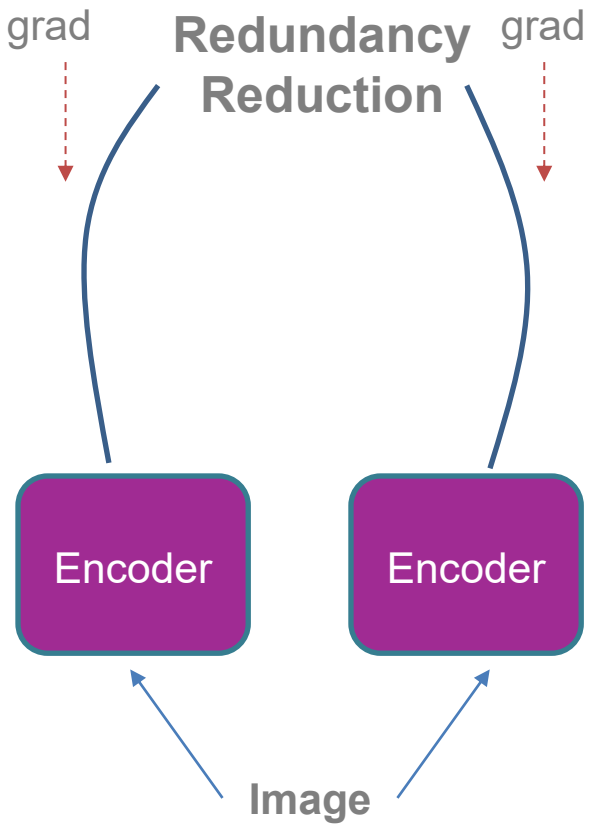
DINOv2 and DINOv3



M. Oquab et al., “DINOv2: Learning Robust Visual Features without Supervision,” arXiv preprint arXiv:2304.07193, 2023.
H. Touvron et al., “DINOv3: Scaling Self-Supervised Learning for Vision Transformers,” arXiv preprint arXiv:2405.16445, 2024

Regularization Methods

- Core idea: Encourage embeddings to be decorrelated across dimensions (maximize information per feature).
- Key property: Avoids collapse by penalizing correlation between embedding dimensions, no negatives needed.
- Examples: Barlow Twins, VICReg



Regularization Methods

Barlow Twins

- **Core idea**. Learn invariances while minimizing redundancy: make the cross-correlation between two augmented views' embeddings close to identity → diagonal ≈ 1 (invariance), off-diagonals ≈ 0 (redundancy reduction).
- Compute the embeddings of the augmented views, $Z_A, Z_B \in \mathbb{R}^{B \times D}$
- standardized **along the batch:**

$$\tilde{Z}_* = (Z_* - \text{mean}_B)/\text{std}_B$$

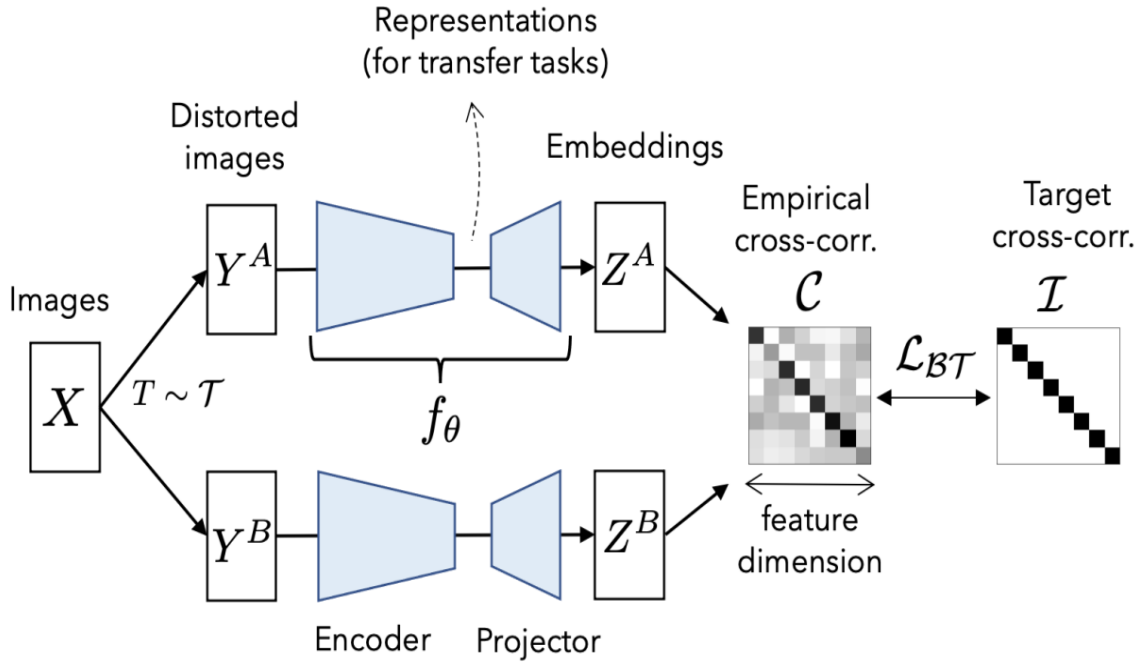
- Compute the Cross Correlation

$$C = \frac{1}{B} \tilde{Z}_A^\top \tilde{Z}_B \in \mathbb{R}^{D \times D}$$

- loss function:

$$\mathcal{L}_{\text{BT}} = \sum_{i=1}^{D} (1 - C_{ii})^2 + \lambda \sum_{i=1}^{D} \sum_{\substack{j=1 \\ j \neq i}}^{D} C_{ij}^2$$

- "invariance" term + "redundancy-reduction" term



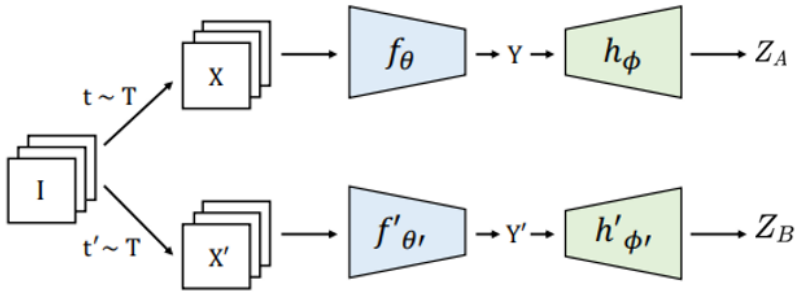
- Pros: Simple objective, no negatives/queues/EMA- Cons: Requires high dimensional embeddings

J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow Twins: Self-Supervised Learning via Redundancy Reduction," in *Proc. 38th Int. Conf. Mach. Learn. (ICML)*, PMLR 139, pp. 12310–12320, 2021.

VICReg

- **Core idea.** Learn view-invariant features **without negatives** by combining three terms:
- **Invariance** — align paired views (MSE).
- **Variance** — keep per-dimension batch std ≥ γ to avoid collapse.
- **Covariance** — decorrelate feature dimensions (off-diagonals → 0).
- For the be projector outputs for two views $Z_A, Z_B \in \mathbb{R}^{B \times D}$

$$\mathcal{L}_{\text{VICReg}} = \lambda \underbrace{\frac{1}{B} \|Z_A - Z_B\|_F^2}_{\text{Invariance}} + \mu \underbrace{\frac{1}{D} \sum_{j=1}^{D} \left([\gamma - \sigma_j(Z_A)]_+^2 + [\gamma - \sigma_j(Z_B)]_+^2 \right)}_{\text{Variance}} + \nu \underbrace{\frac{1}{D} \left(\| \operatorname{offdiag}(\operatorname{Cov}(Z_A)) \|_F^2 + \| \operatorname{offdiag}(\operatorname{Cov}(Z_B)) \|_F^2 \right)}_{\text{Covariance}}$$



- Covariance: $\operatorname{Cov}(Z) = \frac{1}{B-1} \hat{Z}^\top \hat{Z}$
- Per-dim std:

$$\sigma_j(Z) = \sqrt{\frac{1}{B-1} \sum_{b=1}^{B} \left(Z_{b,j} - \bar{z}_j(Z) \right)^2 + \varepsilon}$$

- Batch-centering: $\hat{Z} = Z - \mathbf{1}_B \, \bar{z}(Z)^\top$, with $\bar{z}(Z) = \frac{1}{B} \sum_{b=1}^{B} Z_{b,:}$
- and $[x]_+ = \max(0, x)$

A. Bardes, J.-P. Ponce, and Y. LeCun, “VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning,” in Proc. Int. Conf. Learn. Representations (ICLR), 2022.

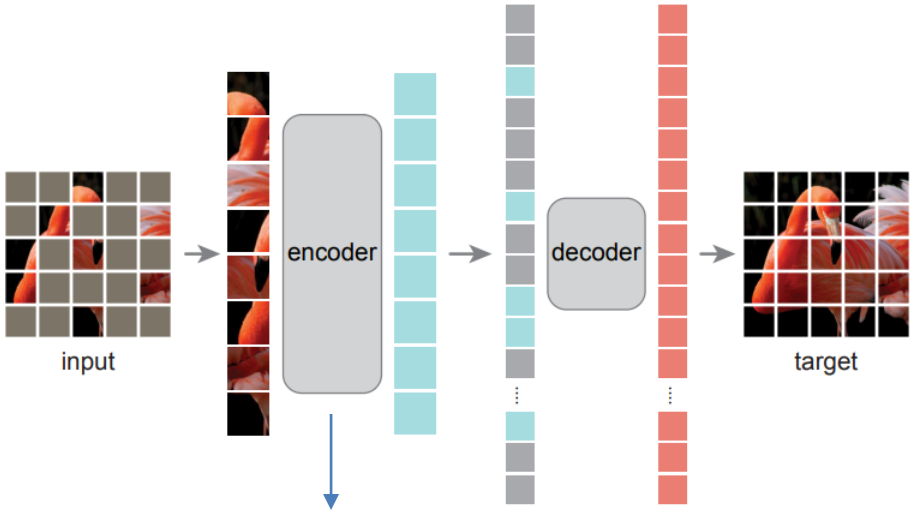
Masked Image Modeling (MIM)

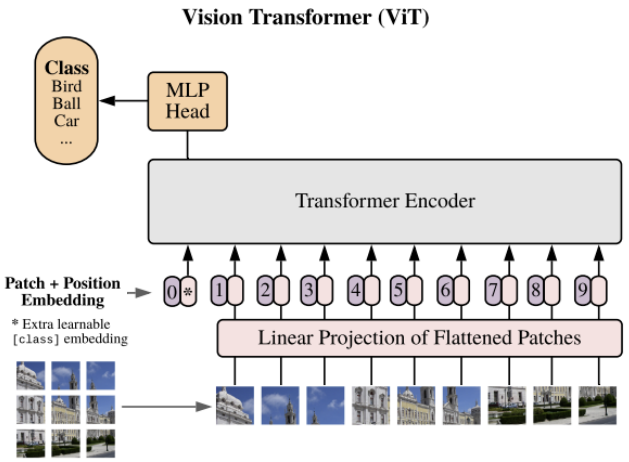
- **Core idea.**: Split an image x into patches; mask a subset $\mathcal{M}$ and learn from the visible context $x_{\sim\mathcal{M}}$.
- Loss: ℓ_2 reconstruction

$$\hat{x} = D(E(x_{\sim\mathcal{M}})), \qquad \mathcal{L}_{\text{pix}} = \frac{1}{|\mathcal{M}|} \sum_{p \in \mathcal{M}} \left\| \hat{x}_p - x_p \right\|_2^2$$

- MIM generally uses ViT backbones because it accepts the sequences of the patch.
- Pros: Simple, efficient, suitable for low-level downstream tasks
- Cons: Weaker for high-level tasks

Masked Autoencoders (MAE)*



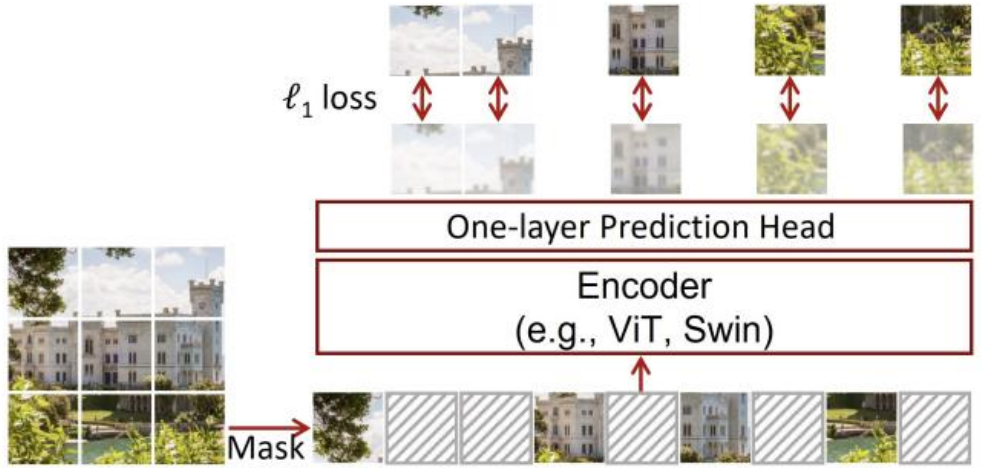


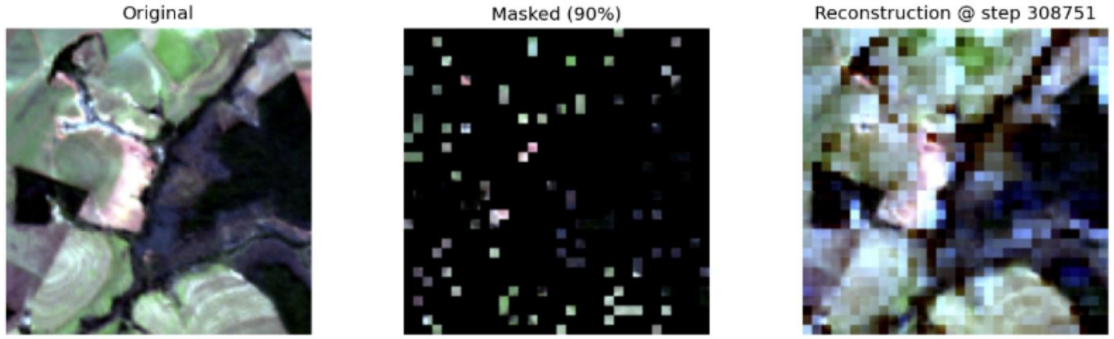
*He, Kaiming, et al. "Masked autoencoders are scalable vision learners." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022.

Masked Image Modeling

- **General idea**
  - Predict missing patches from visible ones
  - Typically, high masking ratio (~75%)
- **Prediction targets**
  - Pixel reconstruction: MAE, SimMIM
  - Feature regression: MaskFeat**
  - Token prediction: BEiT***
- Reconstruction Example (MAE)- Enmap Data from SpectralEarth

Simple Masked Image Modeling*





*Z. Xie *et al.*, "SimMIM: A Simple Framework for Masked Image Modeling," *CVPR*, 2022.
**C. Wei *et al.*, "Masked Feature Prediction for Self-Supervised Visual Pre-Training," *CVPR*, 2022.
***H. Bao, L. Dong, S. Piao, and F. Wei, "BEiT: BERT Pre-Training of Image Transformers," *ICLR*, 2022.

Joint Embedding vs Masked Image Modeling

- Joint Embedding
  - + Highly semantic features, great for classification tasks
  - + Architecture agnostic
  - + Competitive results in linear probing
  - - May require a large batch size
  - - Require tuning the augmentations
  - - Special care for negative samples/collapse
  - - Not fit for low-level tasks

- **Masked Image Modeling**
  - + Conceptually simple, no positive/negative pairs
  - + Masking generally reduces pre-training time
  - + Competitive results with fine-tuning
  - + Stronger fit for low-level tasks (denoising, superresolution)
  - - Requires ViT backbone
  - - Weaker for abstract (higher-level) tasks (classification)

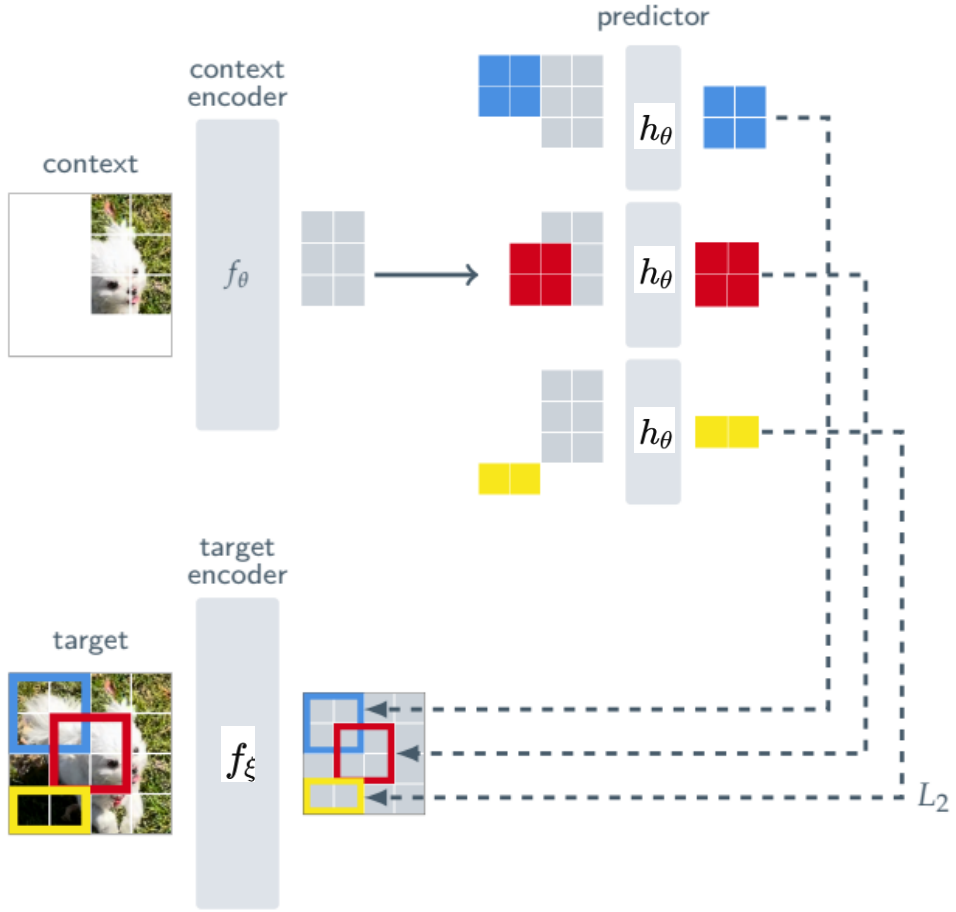
I-JEPA: Image-based Joint Embedding Predictive Architecture (Masking+ Joint Embedding)

- **Core idea:** Predict (reconstruct) *abstract representations* instead of pixels.
- Context encoder (Student, ViT): encodes the visible context patches (mask m_c), $y = f_\theta(x;\, m_c)$
- Predictor (narrow ViT): predicts **patch-level** embeddings for each target block $B_k (k = 1, \ldots, K)$ with mask m_k

$$\hat{Z}_k = h_\theta\left(y, r_k\right) \in \mathbb{R}^{|B_k| \times d}$$

- r_k positional tokens
- Target encoder (teacher, stop-grad, Update→EMA): encodes full image no masking $Z^\star = f_\xi(x)$
- take the ground-truth target block $Z_k^\star = Z^\star[B_k]$
- Loss: ℓ_2 loss in representation space, averaged over all target blocks and their patches

$$\mathcal{L}(x) = \frac{1}{\sum_{k=1}^{K} |B_k|} \sum_{k=1}^{K} \sum_{p \in B_k} \left\| \hat{z}_{k,p} - z_{k,p}^\star \right\|_2^2 = \frac{1}{\sum_k |B_k|} \sum_{k=1}^{K} \left\| \hat{Z}_k - Z_k^\star \right\|_F^2$$



- $|B_k|$: Number of patches in target block k

– Pros: Simple objective, no negatives/queues- Cons: Weaker fit for low-level tasks (denoising, superresolution)

J. Assran *et al.*, "Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 9, pp. 5831–5844, Sep. 2024, doi: 10.1109/TPAMI.2024.3385136

Geospatial Foundation Models

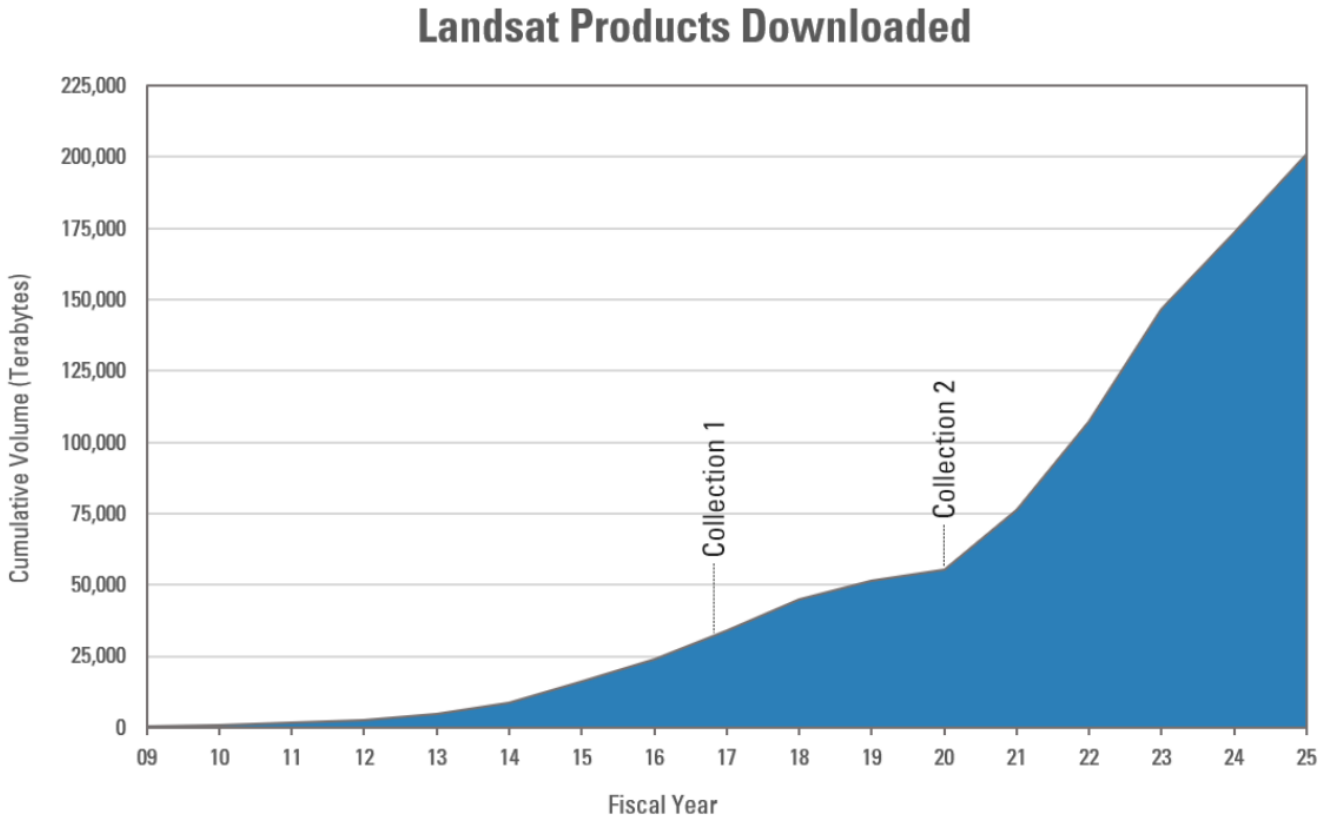
More data than you can digest

- Ground truth in Remote Sensing is rare and costly

- Archives with Peta Bytes of unlabeled data from various satellites

How to exploit unlabeled data for deep learning in EO?

⇒ Self-Supervised Learning!

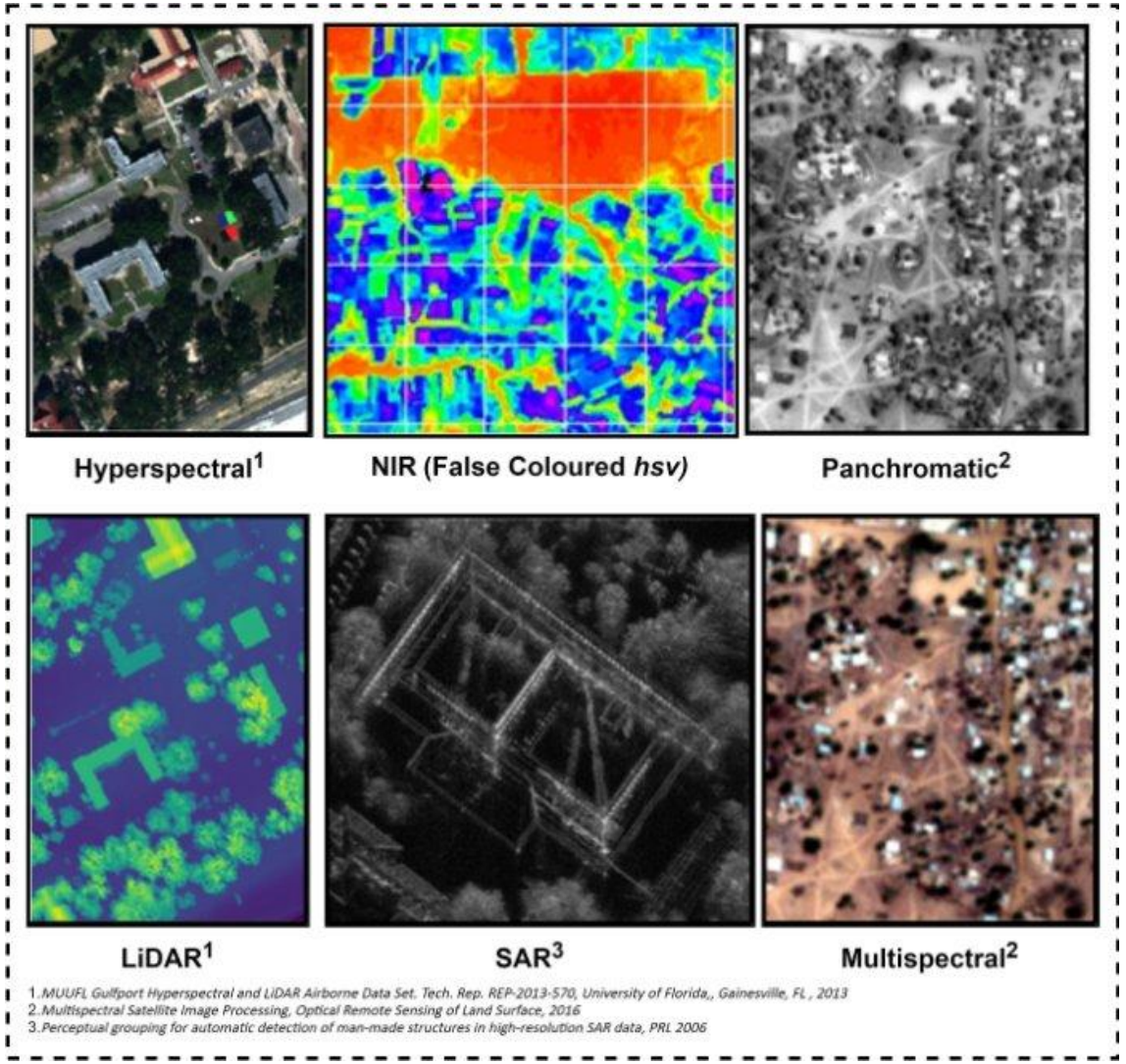


Cumulated Landsat downloads since 2008

Landsat Project Statistics | U.S. Geological Survey

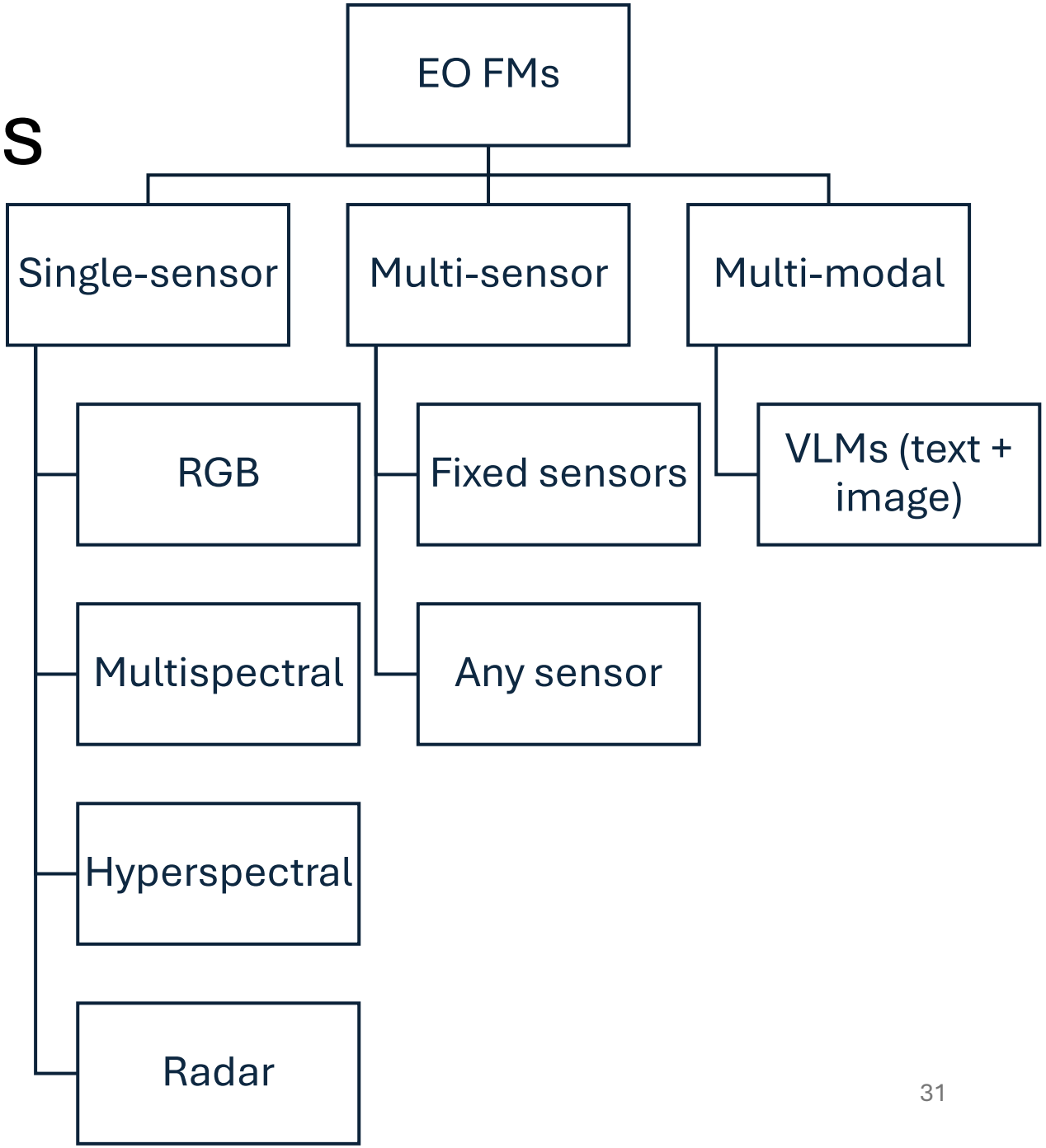
Remote Sensing Data

- Geolocated
- Timestamped
- Domain specific meta-data
- Multi-sensor
- Varying spatial resolution
- Varying spectral resolution
- Many pre-processing pipelines
- Data quality issues (clouds, noise..)



EO Foundation Models

- **A lot** of research in SSL for Earth Observation
- Early works (~2020-2023) focused on **small uni-sensor** models
- Recent literature has shifted towards **large multi-sensor/modal** networks



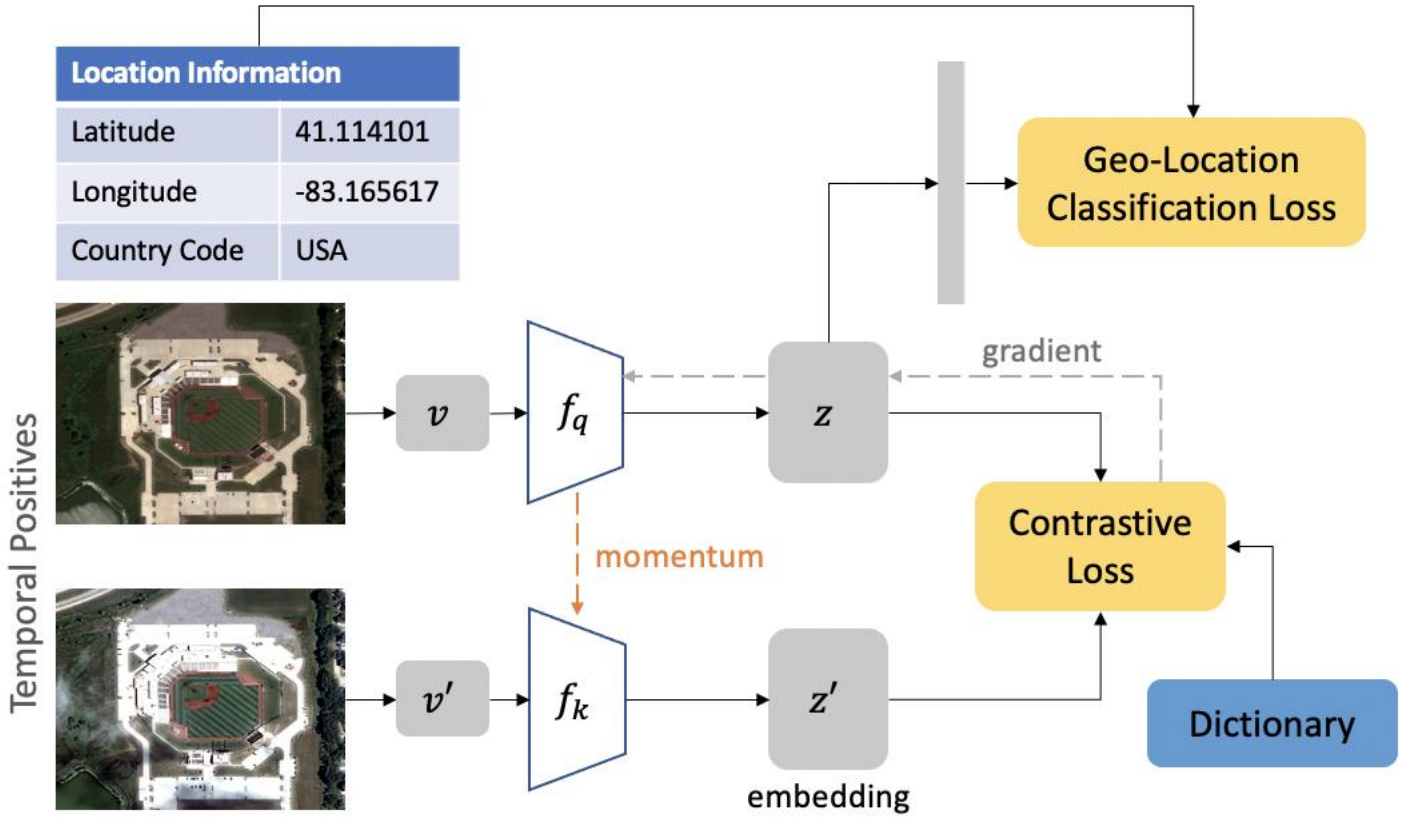
RGB/Multispectral FMs

Geo-Moco (ICCV, 2021)

- **Idea:** Exploit **geo-location** during contrastive pre-training

- Uses **temporal positives** as a *natural augmentation*

- Adds a **Geo-location** pretext task => learn location-aware representations



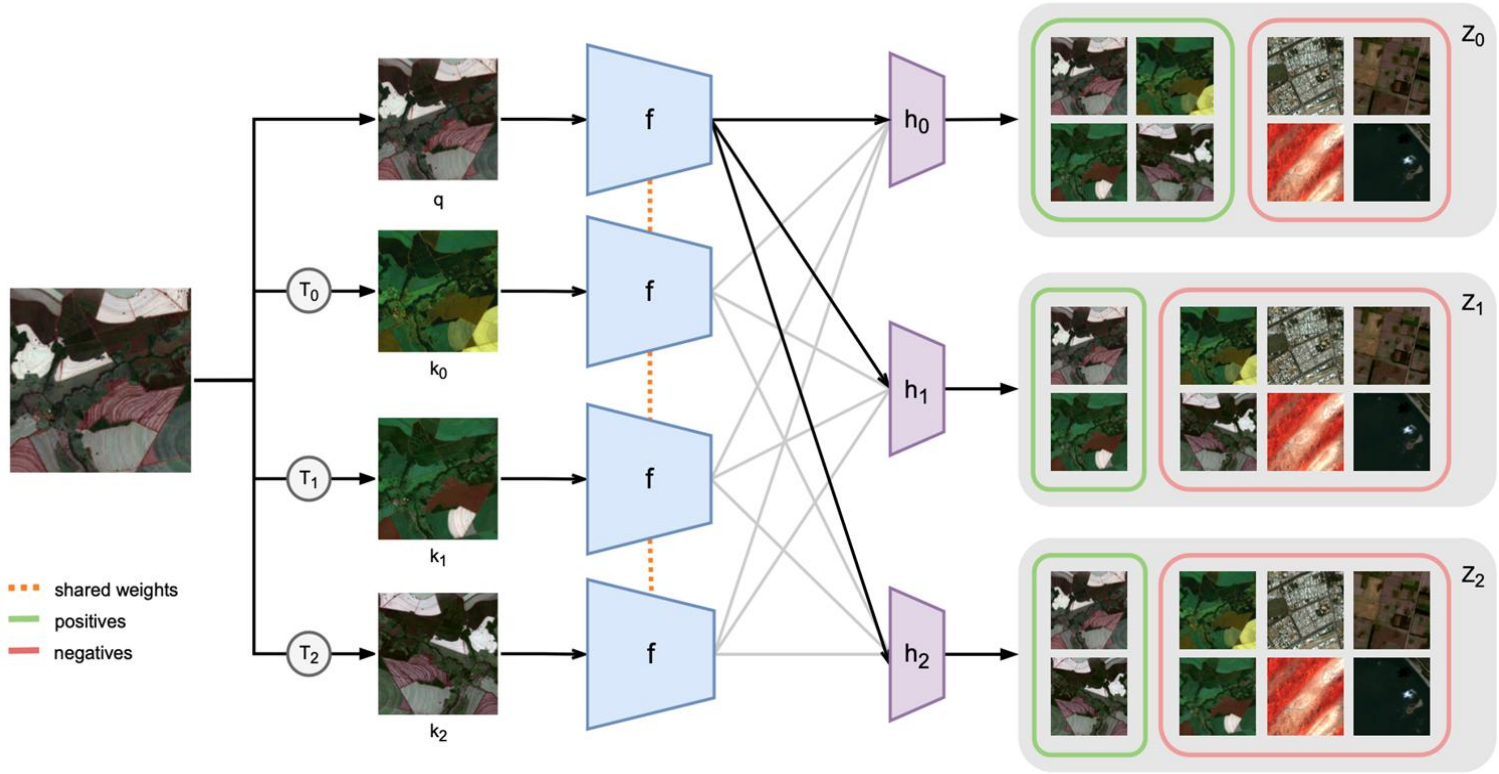
Ayush, Kumar, et al. "Geography-aware self-supervised learning." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

SeCo: Seasonal contrast (ICCV, 2021)

- **Idea:** Contrastive learning with views from different seasons, **Capturing seasonal changes**

- Draw images from Gaussians around **most populated cities**

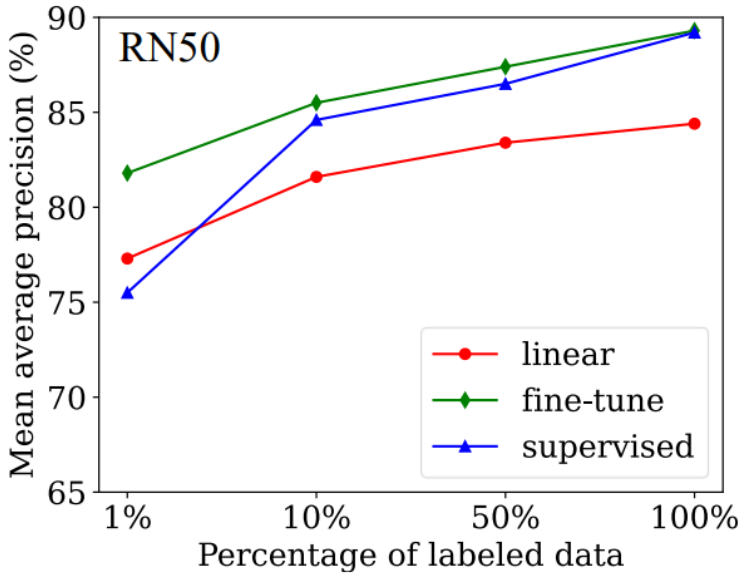
- Multiple subspaces:
  - Z_0: invariant to seasonal variation and data augmentation
  - Z_1: invariant to seasonal variation
  - Z_2: invariant to data augmentation



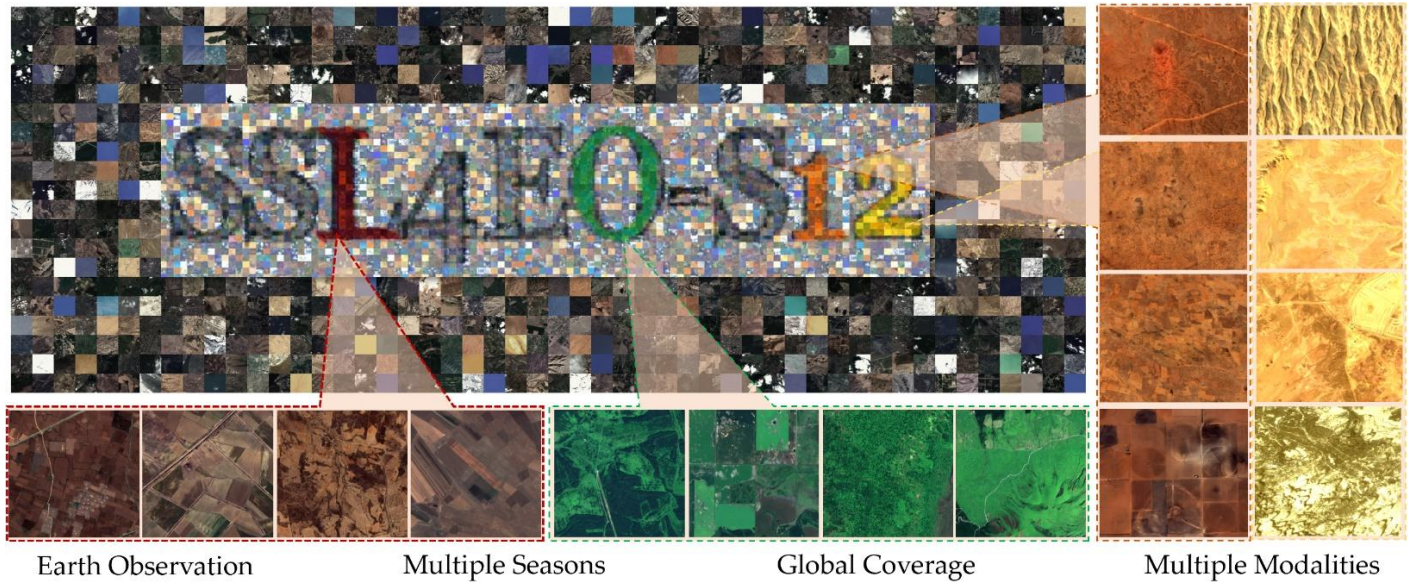
Manas, Oscar, et al. "Seasonal contrast: Unsupervised pre-training from uncurated remote sensing data." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

SSL4EO-S12 (GRSM 2023)

- ~250,000 S2-S1 patches
- 264x264 pixels
- 1.5TB of data
- 4 timestamps per location



Results on BigEarthNet: Pre-training improves performance and label efficiency



Wang, Y., Braham, N. A. A., Xiong, Z., Liu, C., Albrecht, C. M., & Zhu, X. X. (2023). SSL4EO-S12: A large-scale multimodal, multitemporal dataset for self-supervised learning in Earth observation [Software and Data Sets]. *IEEE Geoscience and Remote Sensing Magazine*, *11*(3), 98-106.

SSL4EO-L (NeurIPS 2023 - Dataset Track)

- **Data**
  - 5 Landsat Sensors
  - 4 timestamps per location
- **Models**
  - SimCLR and MoCo
  - ResNet-18/50 and ViT-S

(a) OLI/TIRS

(b) TM

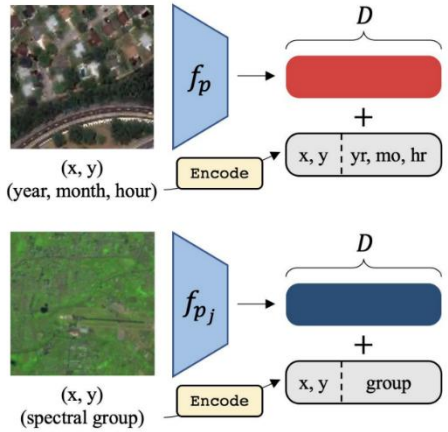
(c) ETM+

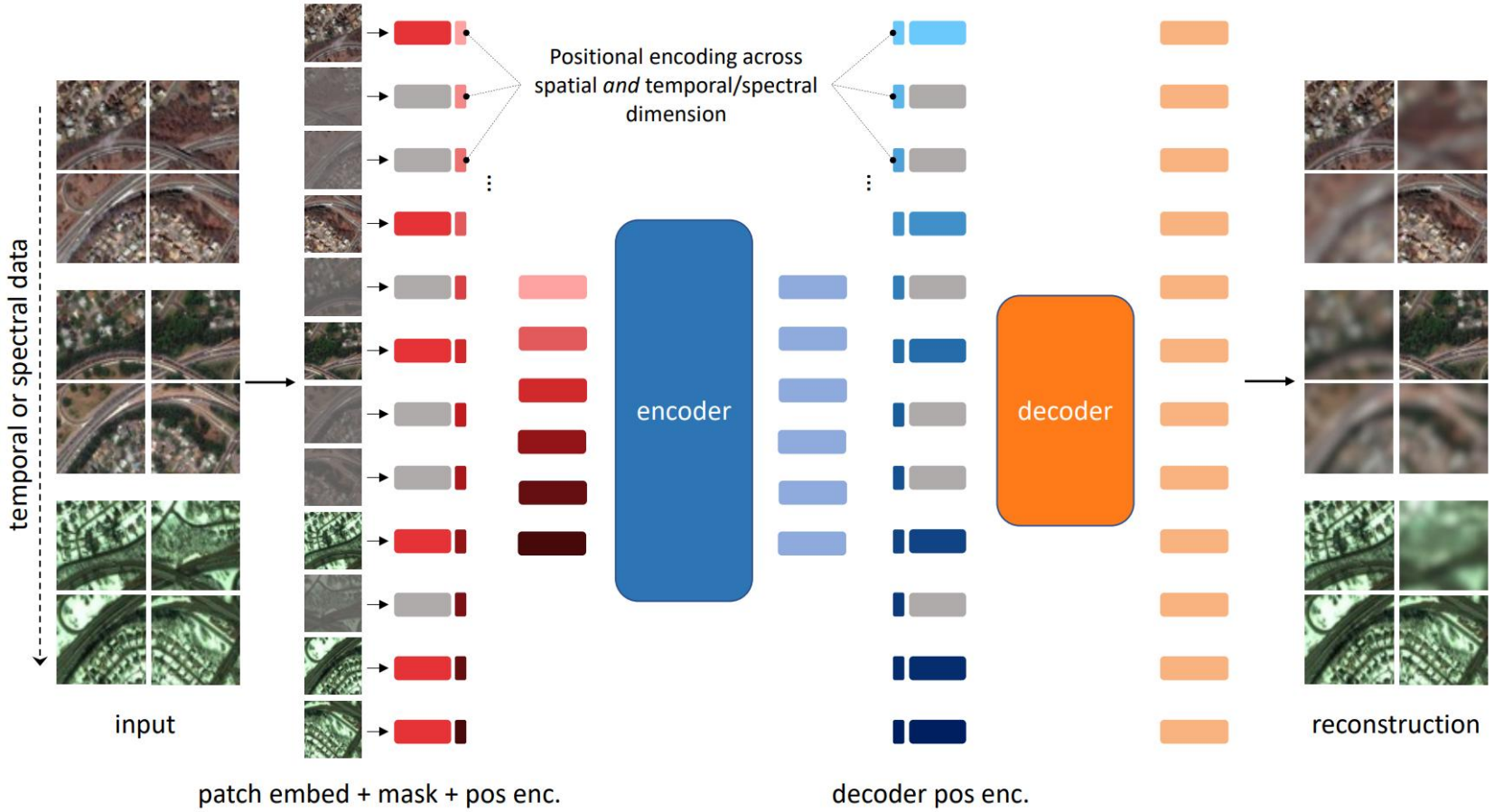
Stewart, Adam, et al. "Ssl4eo-l: Datasets and foundation models for landsat imagery." *Advances in Neural Information Processing Systems* 36 (2023): 59787-59807.

SatMAE (NeurIPS 2022)

- Adapt MAE to **Multispectral Multitemporal inputs**

- **Temporal Encoding** and **Spectral Encoding** help improve performance



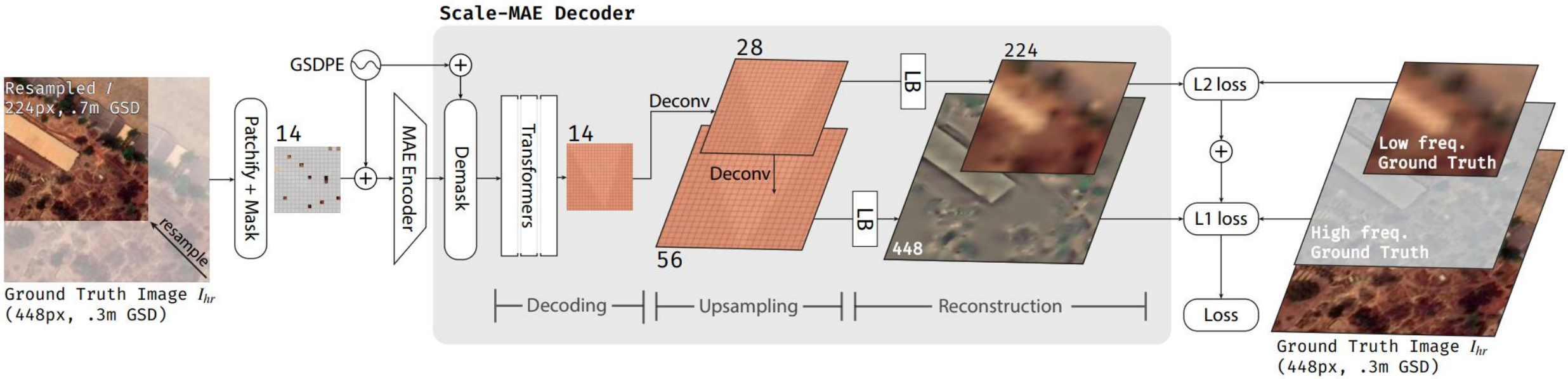


Cong, Yezhen, et al. "Satmae: Pre-training transformers for temporal and multi-spectral satellite imagery." Advances in Neural Information Processing Systems 35 (2022): 197-211.

ScaleMAE (ICCV, 2023)

- Introduce scale information in MAE

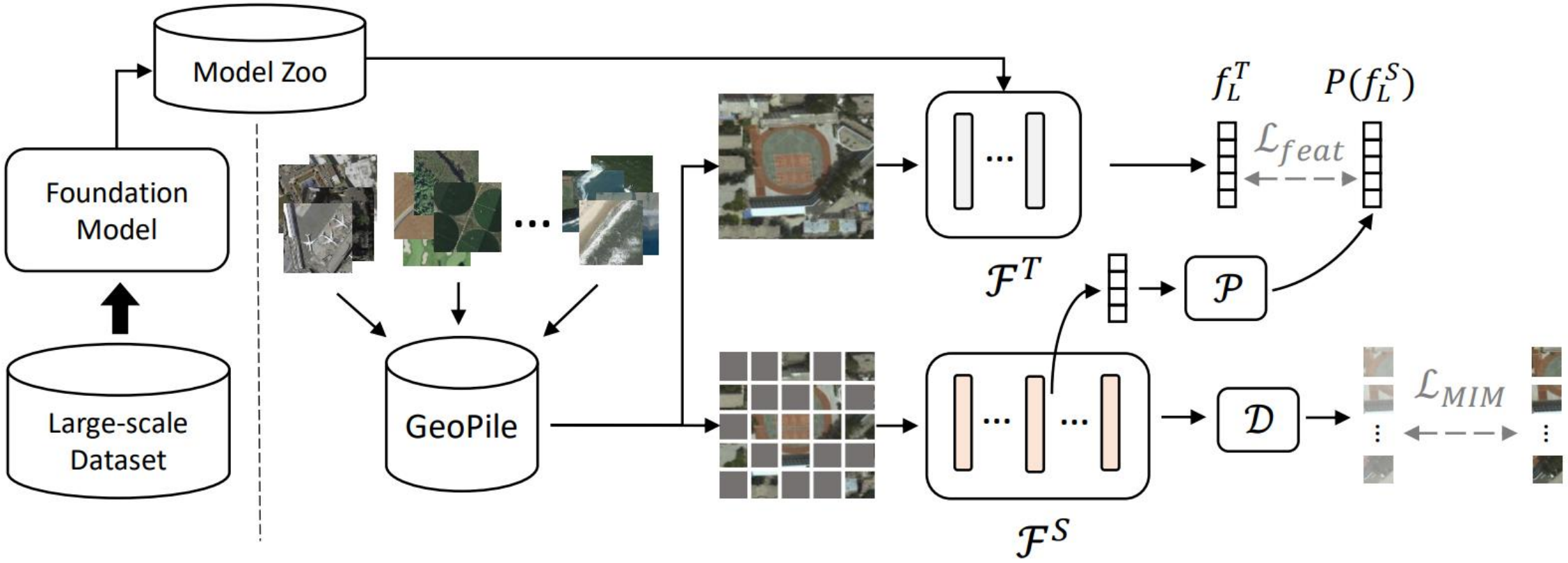
- Ground Sampling Distance Positional Encodings
- Decode high frequency and low frequency parts of the input



Reed, Colorado J., et al. "Scale-mae: A scale-aware masked autoencoder for multiscale geospatial representation learning." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023.

GFM (ICCV, 2023)

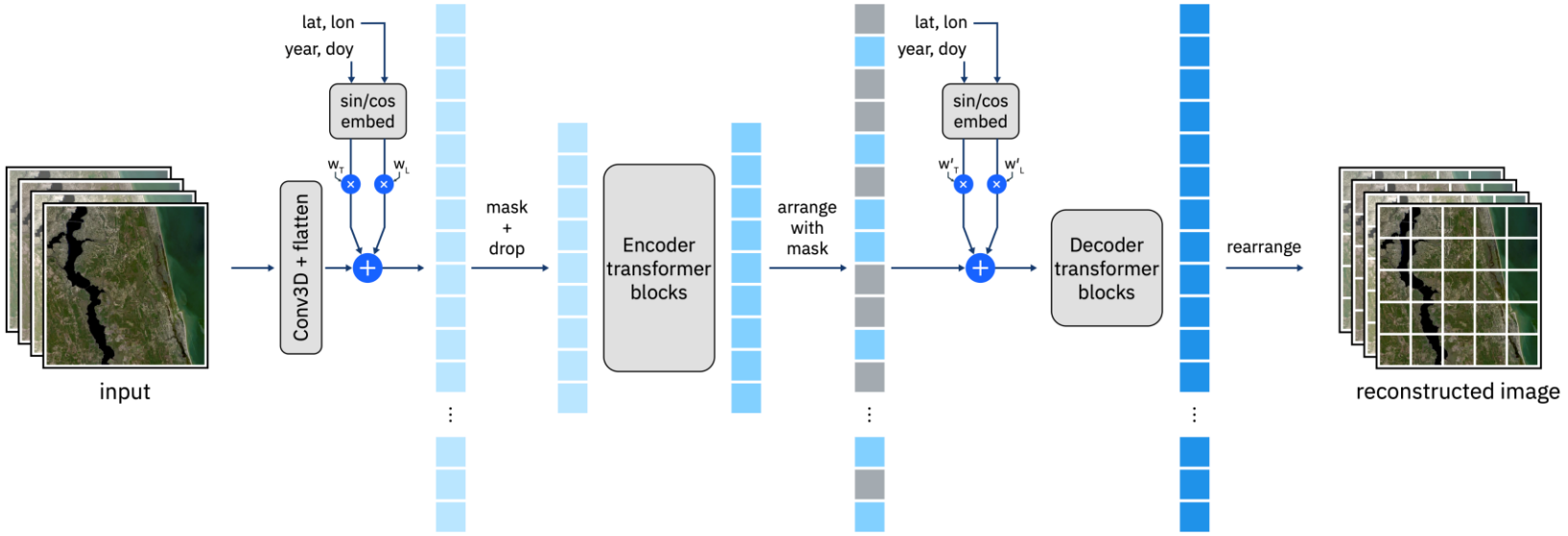
- MAE + Distillation from CV model pre-trained on ImagNet-22k



Mendieta, Matías, et al. "Towards geospatial foundation models via continual pretraining." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023.

Prithvi & Prithvi-EO-2.0 (arXiv 2023/2024)

- IBM/NASA foundation model, based on MAE
- Trained on Harmonized Landsat/S2 imagery
- Cool insights on the technical challenges when training a FM



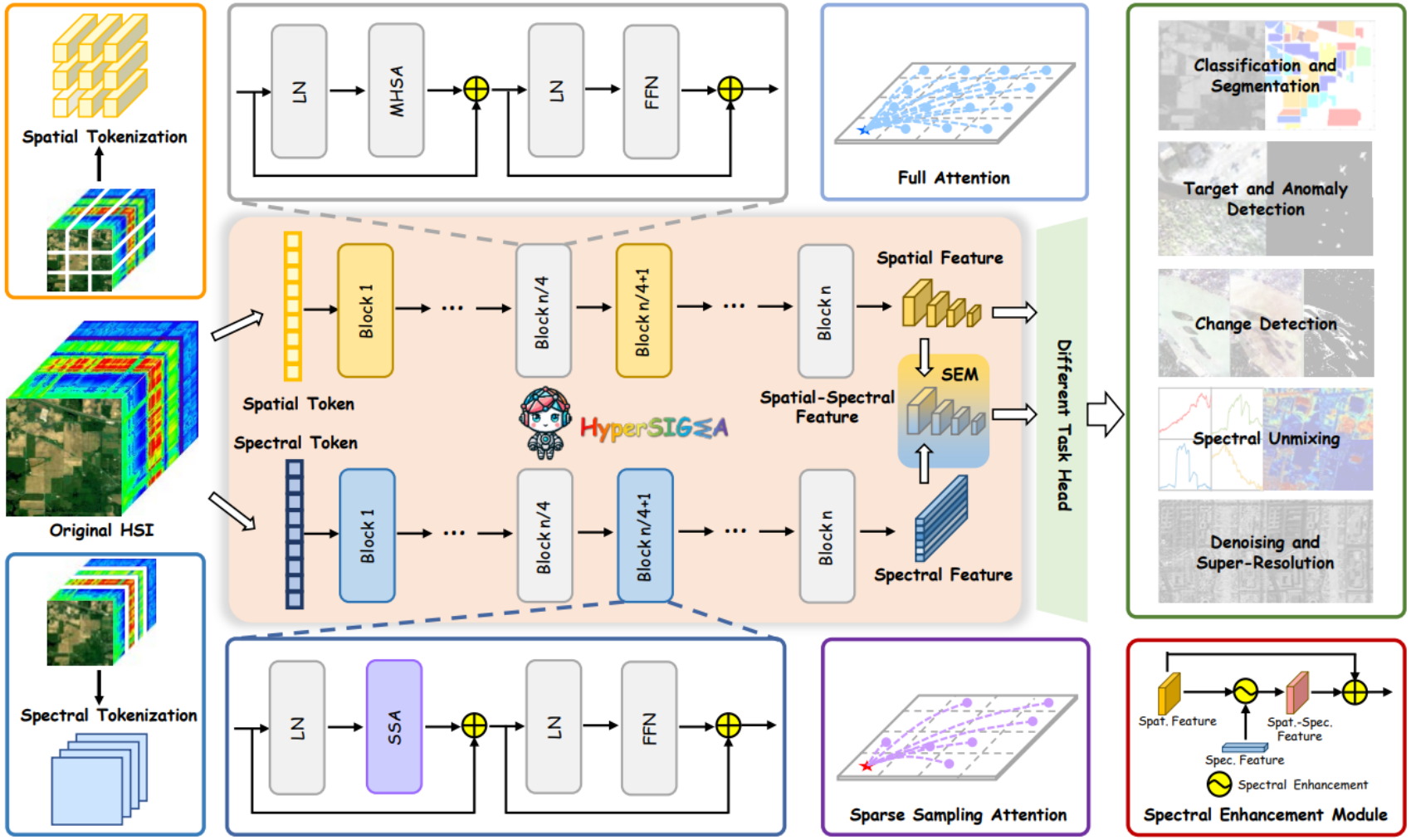
Szwarcman, Daniela, et al. "Prithvi-eo-2.0: A versatile multi-temporal foundation model for earth observation applications." *arXiv preprint arXiv:2412.02732* (2024).

Jakubik, Johannes, et al. "Foundation Models for Generalist Geospatial Artificial Intelligence." arXiv, 2023, https://arxiv.org/abs/2310.18660.

Hyperspectral FMs

HyperSigma (TPAMI, 2025)

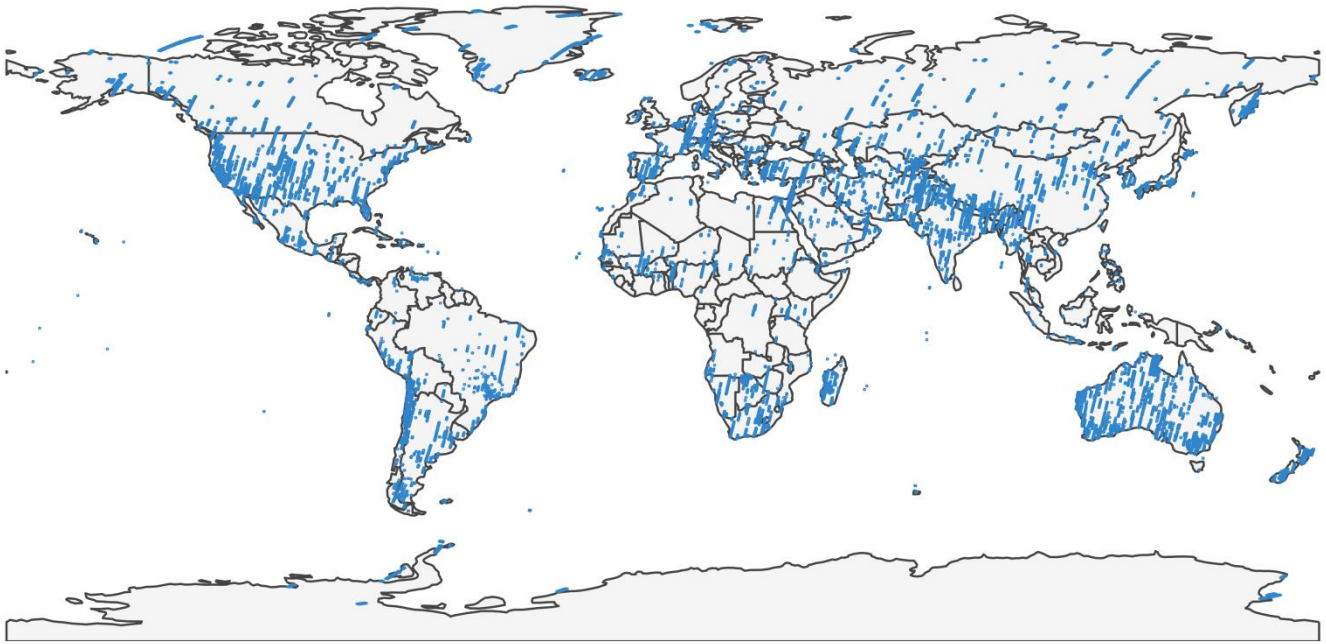
- A hyperspectral FM based on MAE
- Trained on Hyperion EO-1 and Gaofen-5 imagery
- Includes a spatial and a spectral transformer

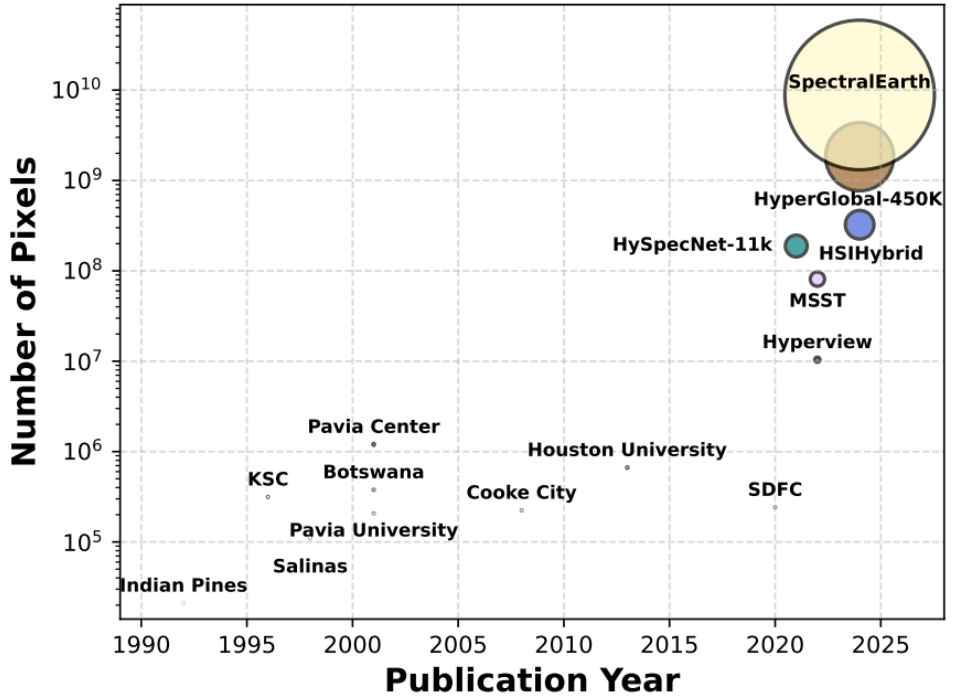


Wang, Di, et al. "Hypersigma: Hyperspectral intelligence comprehension foundation model." *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2025).

SpectralEarth (JSTARS, 2025)

- **Motivation:** A lot of research in RGB/multispectral foundation models
  - Much less in the hyperspectral domain
- **Contribution:** A large-scale dataset + hyperspectral foundation models



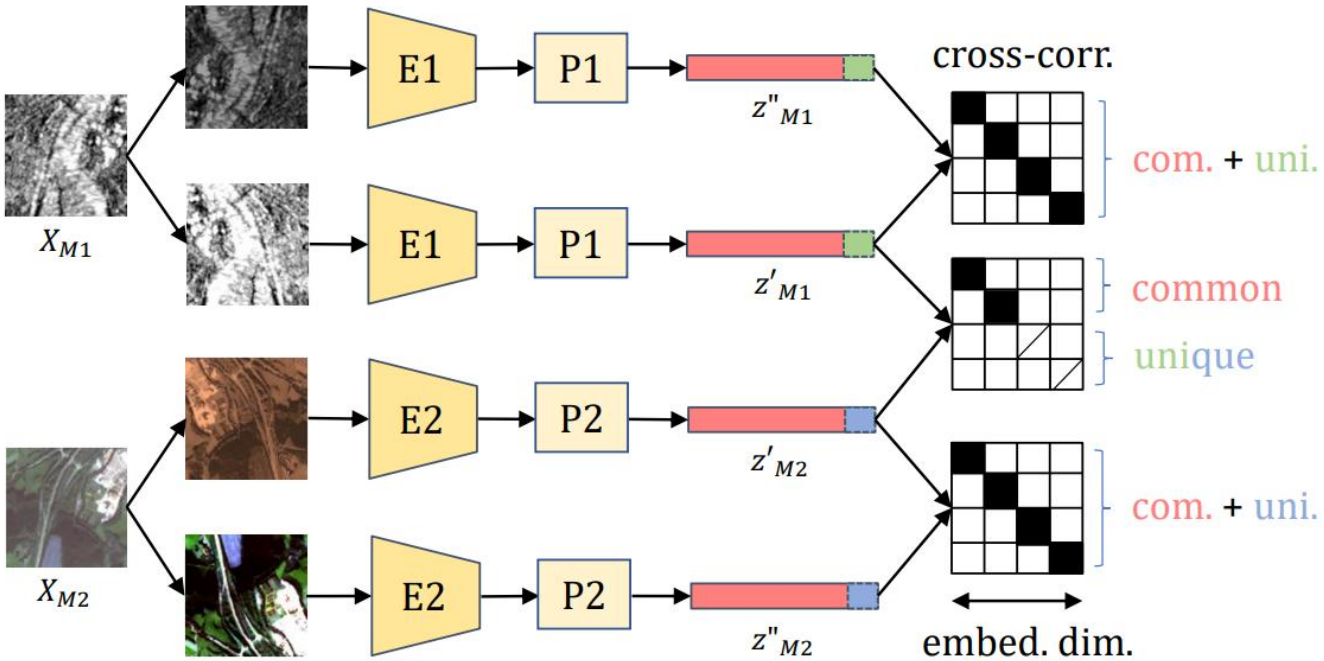


Braham, Nassim Ait Ali, et al. "Spectralearth: Training hyperspectral foundation models at scale." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* (2025).

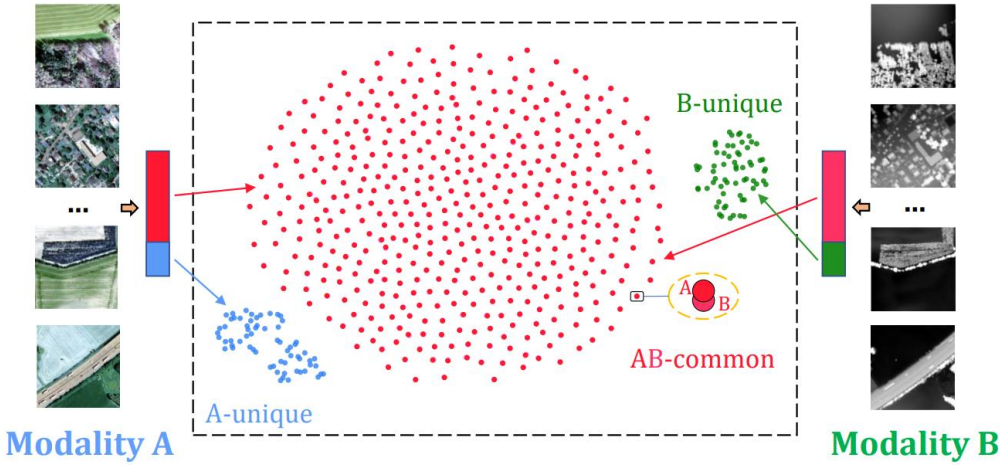
Multi-Sensor & Multi-Modal FMs

DeCUR (ECCV, 2024)

- **Motivation:** How to leverage complementary information from different sensors?
- **Contribution:** Sensor fusion in SSL using Barlow-Twins preserving both **common** and **unique** features from both modalities



A schematic overview of DeCUR

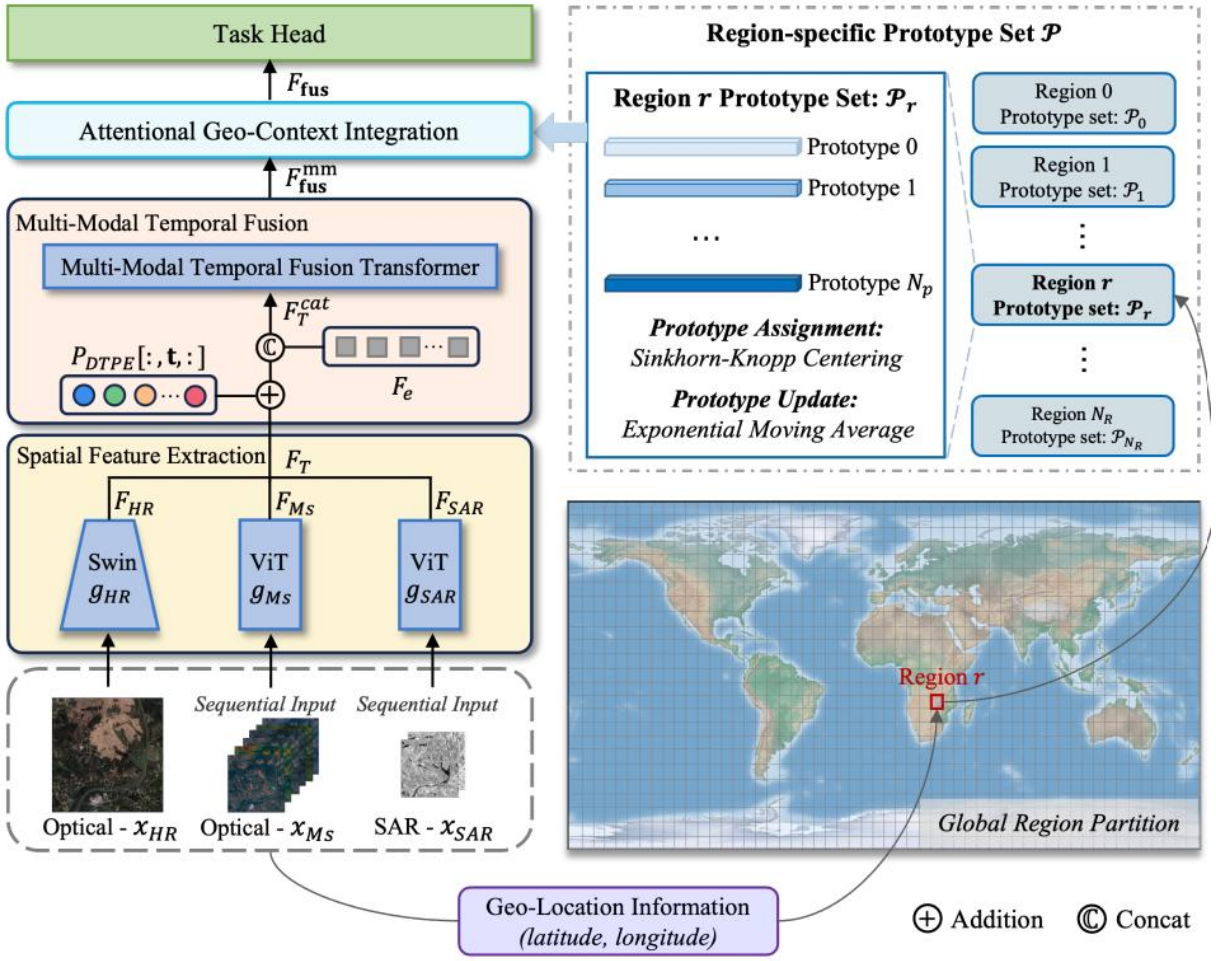


T-SNE projection of learned features using DeCUR

Wang, Yi, et al. "Decoupling common and unique representations for multimodal self-supervised learning." *18th European Conference on Computer Vision, ECCV 2024*. 2024.

SkySense (CVPR, 2024)

- Trained on a large dataset of 21M World View images, Sentinel-1/2 time-series
- Based on joint-embedding architecture
- Exploits geolocations with learnable region specific prototypes

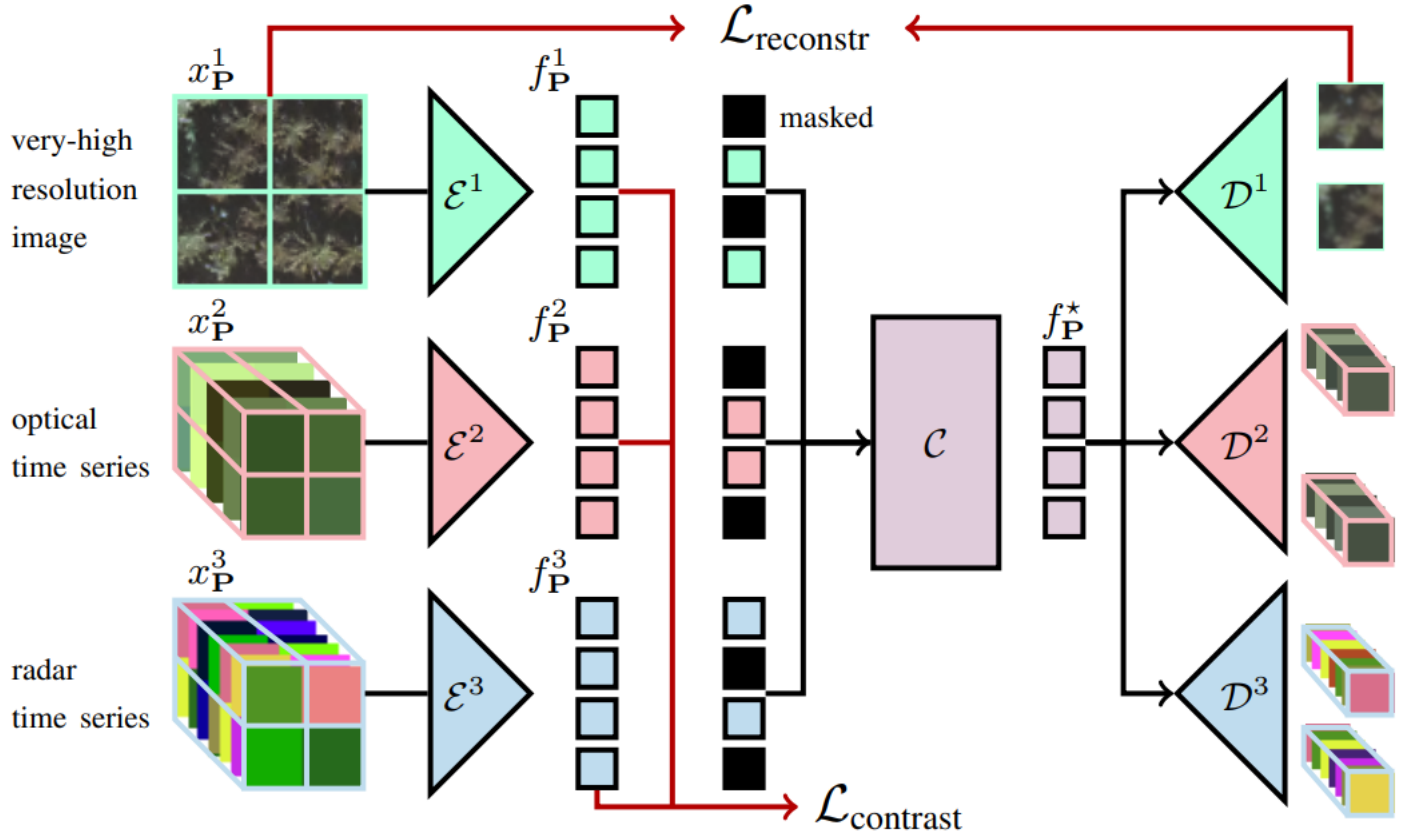


Guo, Xin, et al. "Skysense: A multi-modal remote sensing foundation model towards universal interpretation for earth observation imagery." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024.

OmniSat (ECCV, 2024)

- **Motivation:** A lot of mono-sensor foundation models. How to exploit the complementarity of different sensors?

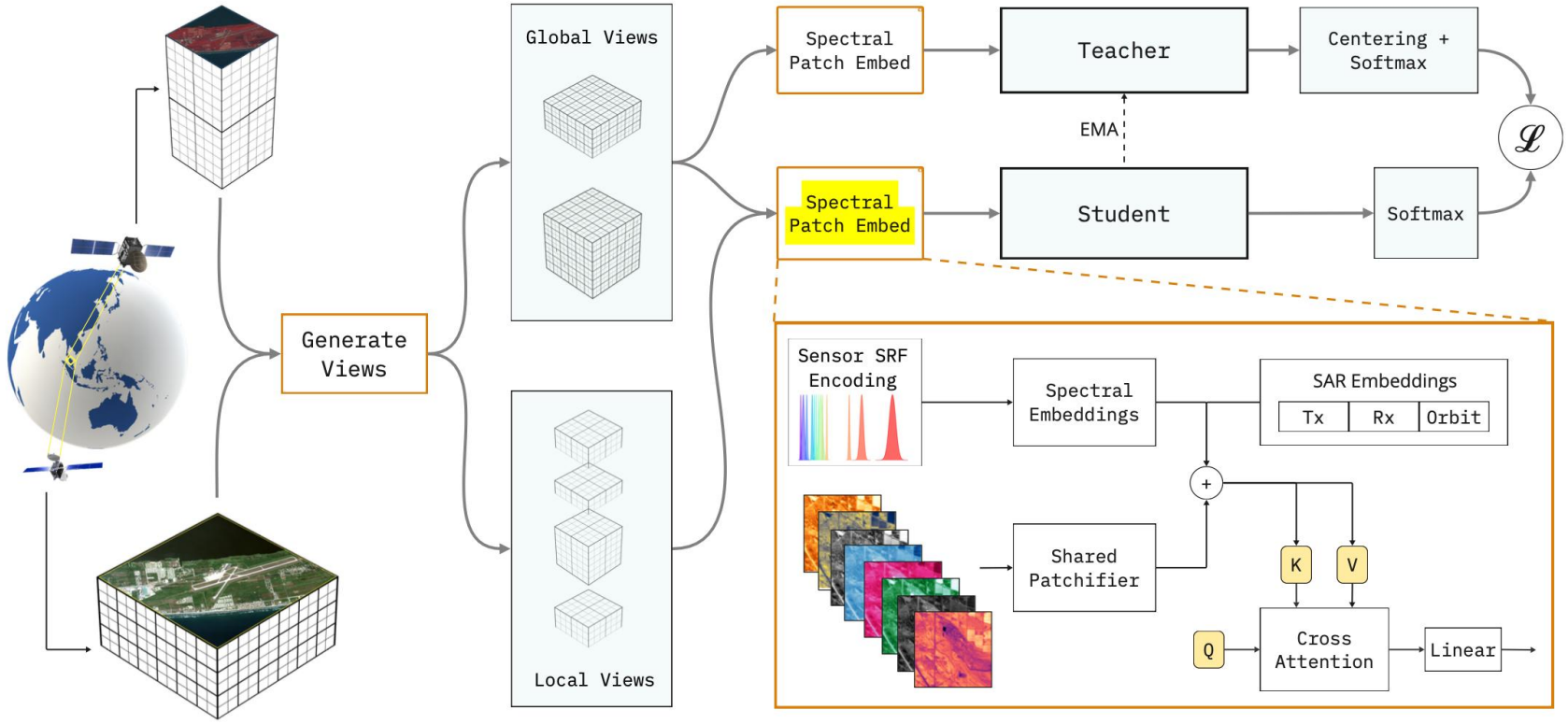
- **Contribution:** a pre-training algorithm that can handle
  - An arbitrary number of sensors
  - Timeseries data



Astruc, Guillaume, et al. "Omnisat: Self-supervised modality fusion for earth observation." *European Conference on Computer Vision*. Springer, Cham, 2025.

Panopticon (CVPR EarthVision, 2025)

- Any-sensor model: integrates the spectral response function
- Pre-trained on a mix datasets: SpectralEarth, FMoW, MMEarth and SatlasPretrain
  - EnMAP, Multispectral (Landsat-9, Sentinel-2), High Res RGB and Radar (Sentinel-1)

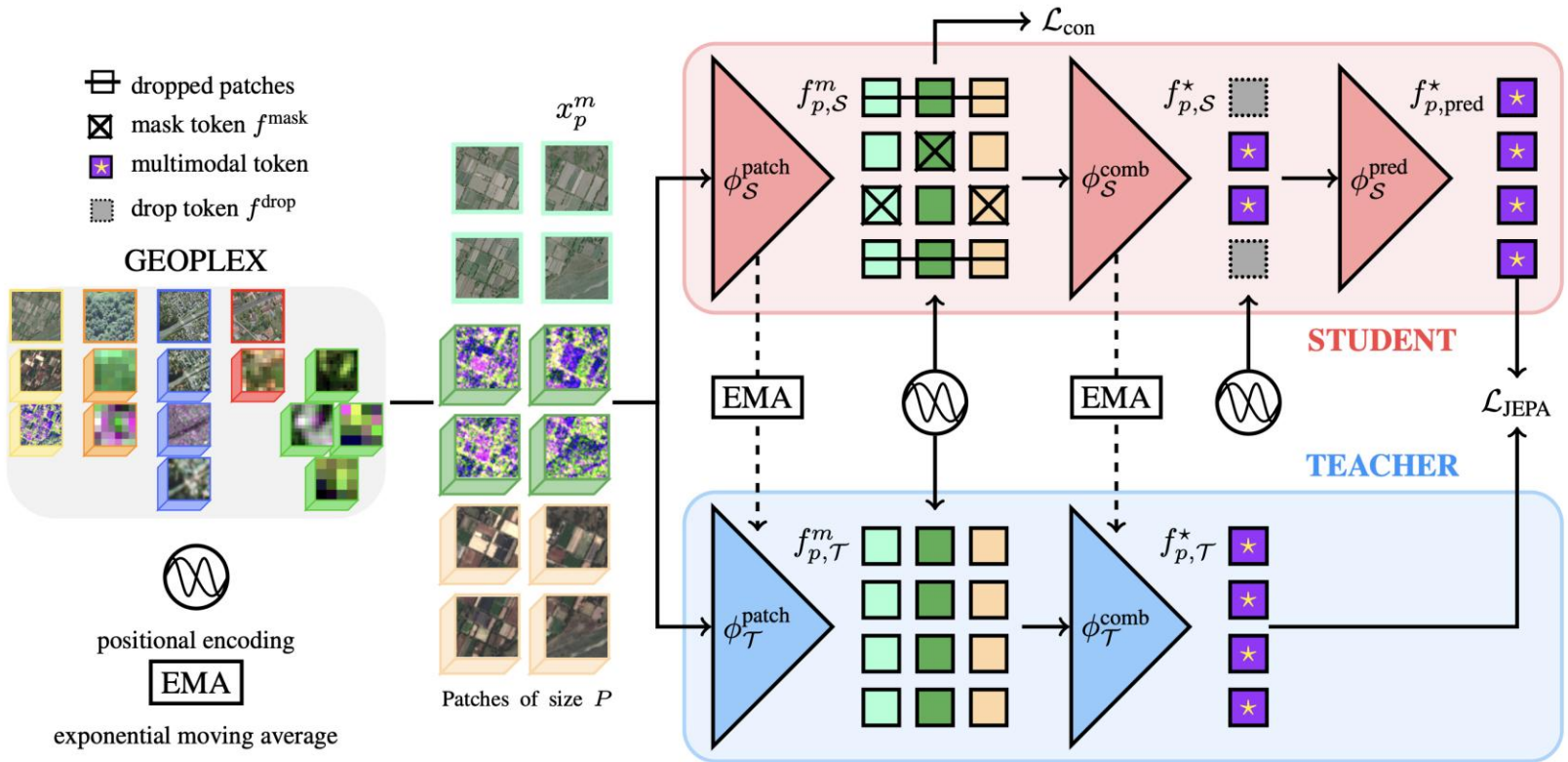


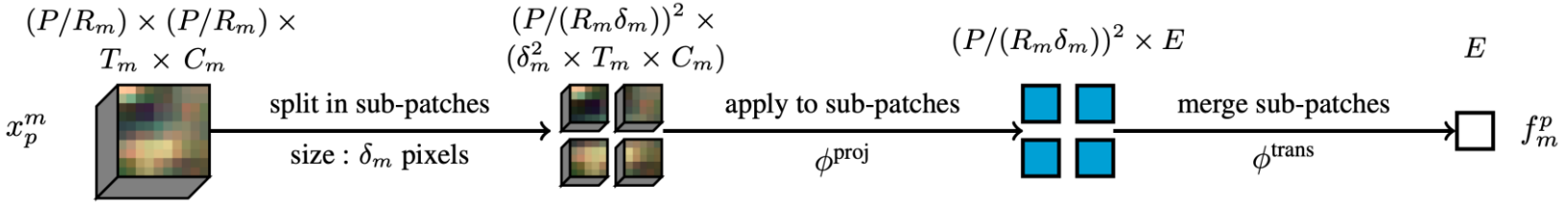
Waldmann, Leonard, et al. "Panopticon: Advancing any-sensor foundation models for earth observation." *Proceedings of the Computer Vision and Pattern Recognition Conference*. 2025.

AnySat (CVPR, 2025)

- Handles A wide range of sensors by combining existing datasets (High Res RGB, Sentinel-2, Sentinel-1, Landsat 7-9, MODIS)

- Tricks to handle the large gaps in spatial

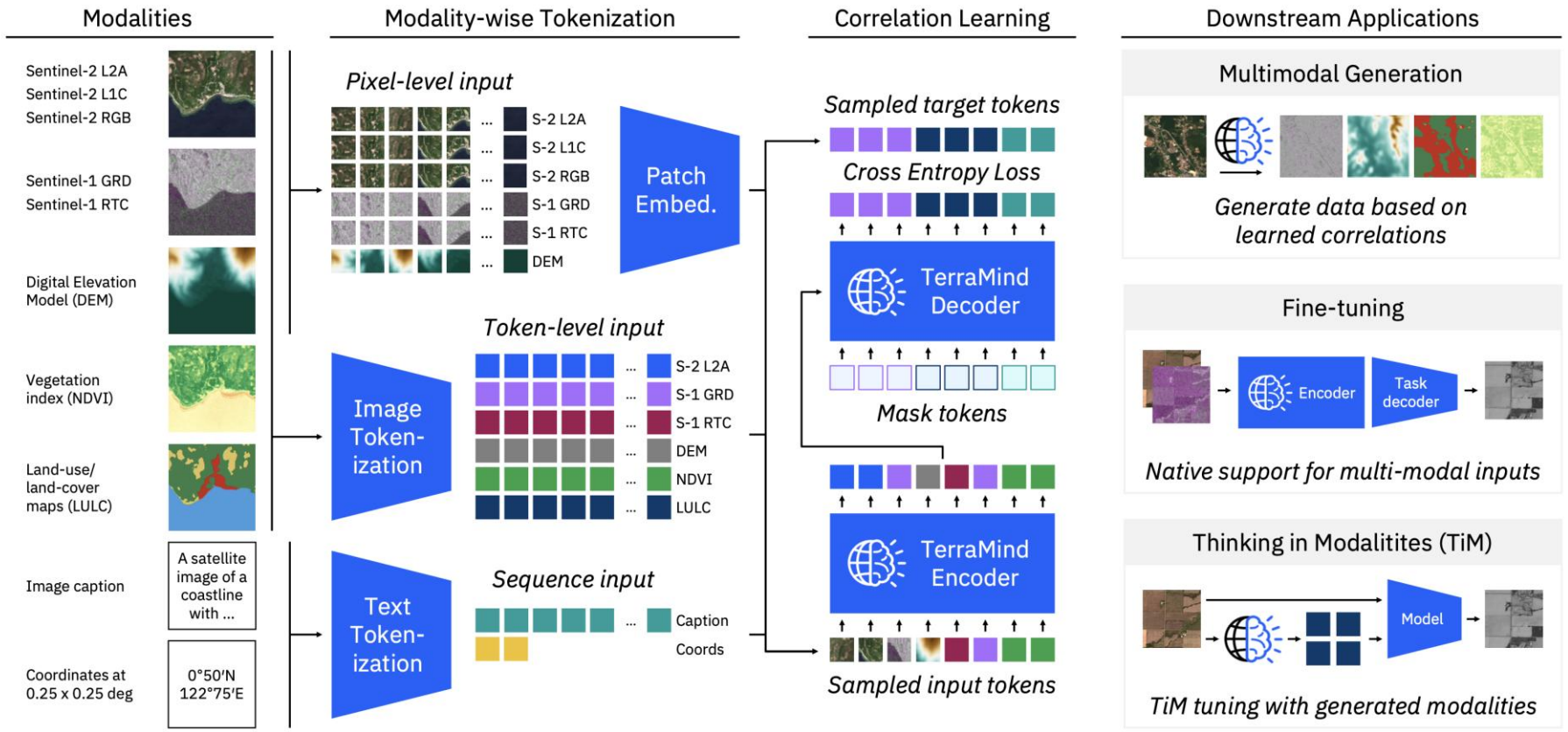




[6] Astruc, G., Gonthier, N., Mallet, C., & Landrieu, L. (2024). AnySat: An Earth Observation Model for Any Resolutions, Scales, and Modalities. *arXiv preprint arXiv:2412.14123*.

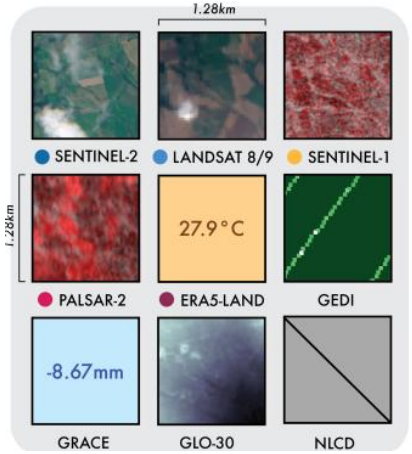
Terramind (ICCV, 2025)

- Trained on 9M locations with Sentinel-2, Sentinel-1, DEM, LULC maps and text
- Enables sensor-to-sensor generation through cross-modality reconstruction

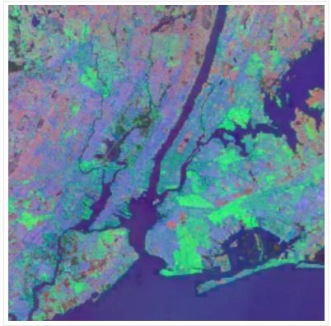


AlphaEarth (arXiv, 2025)

- A foundation model trained on annual time-series
- Released in the form of global embeddings



Satellite Embedding V1



Dataset Availability

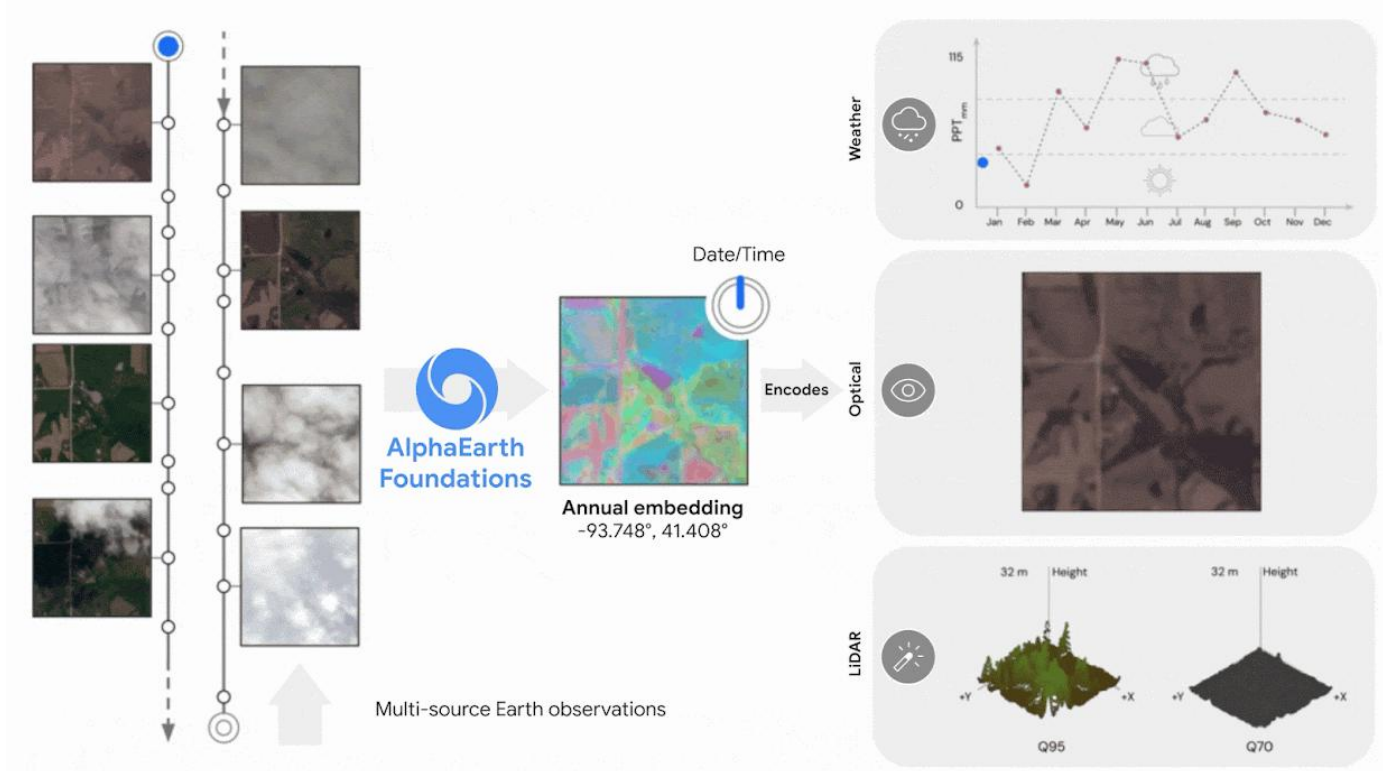
2017-01-01T00:00:00Z–2024-01-01T00:00:00Z

Dataset Provider

Google Earth Engine Google DeepMind

Earth Engine Snippet

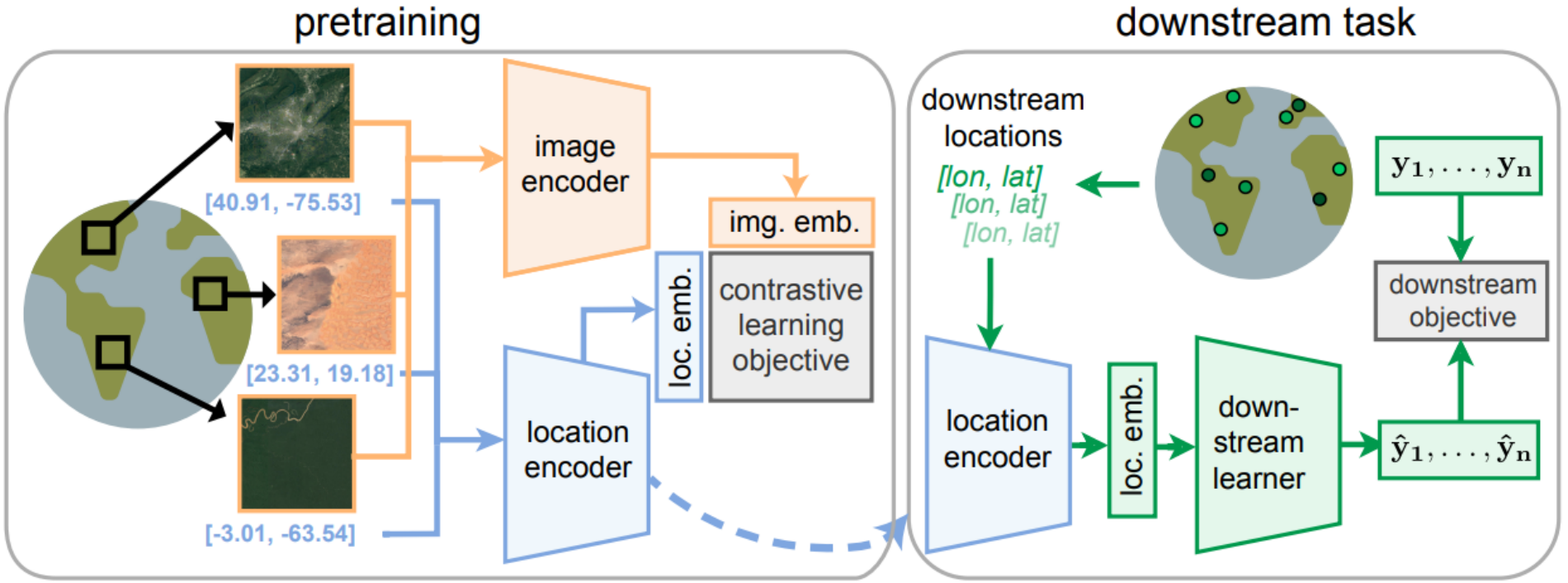
```
ee.ImageCollection("GOOGLE/SATELLITE_EMBEDDING/V1/ANNUAL")
```



Brown, Christopher F., et al. "AlphaEarth Foundations: An embedding field model for accurate and efficient global mapping from sparse label data." *arXiv preprint arXiv:2507.22291* (2025).

SatCLIP (AAAI, 2025)

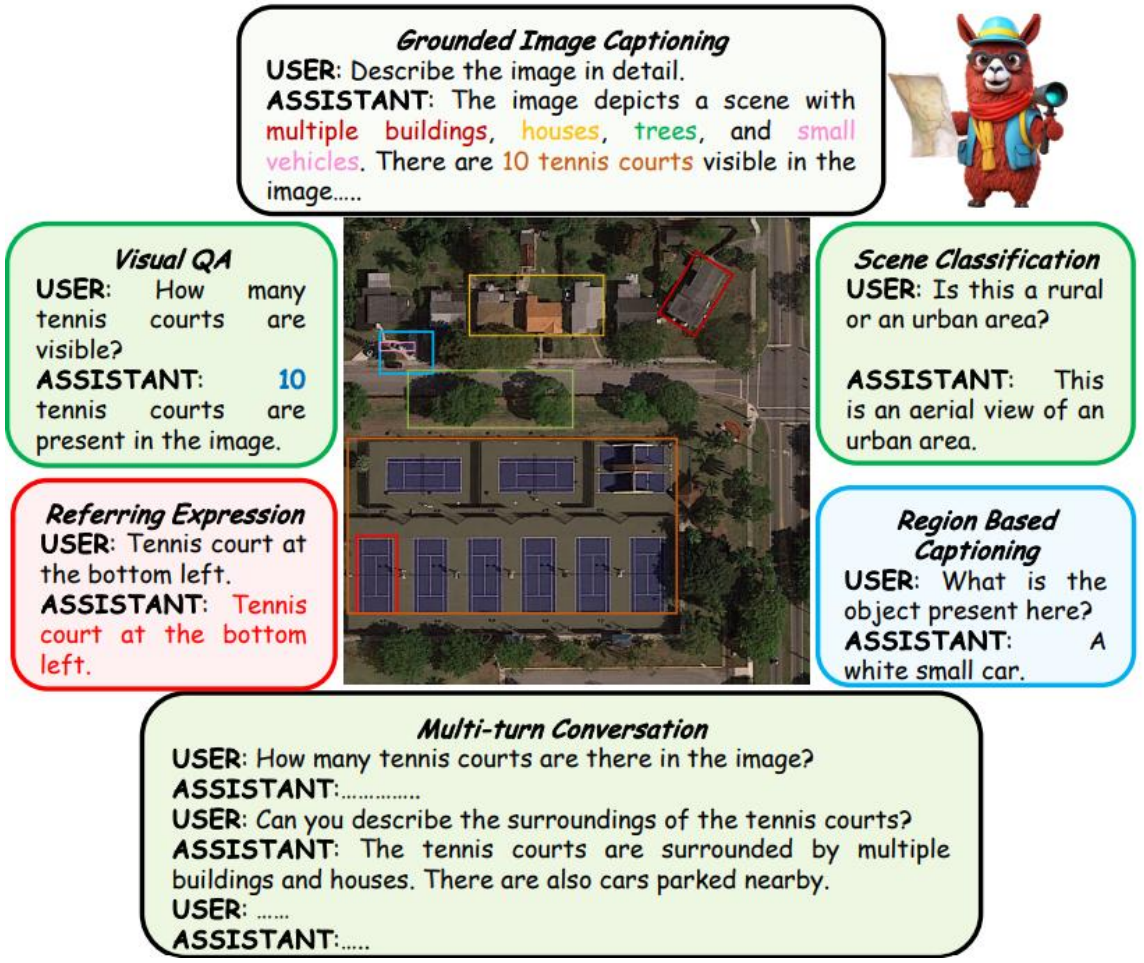
- **Motivation:** incorporate meta-data as additional information
- **Idea:** use a neural net to encode location and perform location-image contrastive learning



Klemmer, Konstantin, et al. "Satclip: Global, general-purpose location embeddings with satellite imagery." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 39. No. 4. 2025.

GeoChat (CVPR 2024)

- Vision/text models allow prompting
- Often rely on vision-text contrastive learning
- Combines a vision encoder and an LLM



Kuckreja, Kartik, et al. "Geochat: Grounded large vision-language model for remote sensing." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024.

Evaluating Geospatial FMs

Using a Foundation Model

Frozen Encoder

+ Fewer parameters to train
+ Computationally cheaper
+ Less memory intensive
+ Less prone to overfitting

- Frozen generic features can be suboptimal

Full Fine-tuning

+ Can yield best results with enough labels

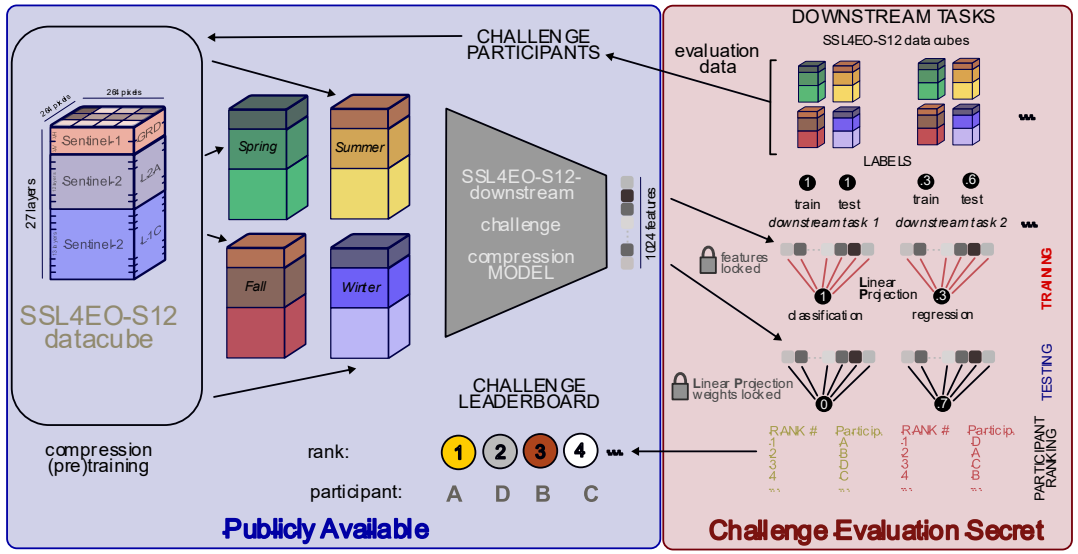
- Prone to overfitting
- Computationally & memory demanding
- Catastrophic forgetting

Efficient Adaptation (e.g., LoRA)

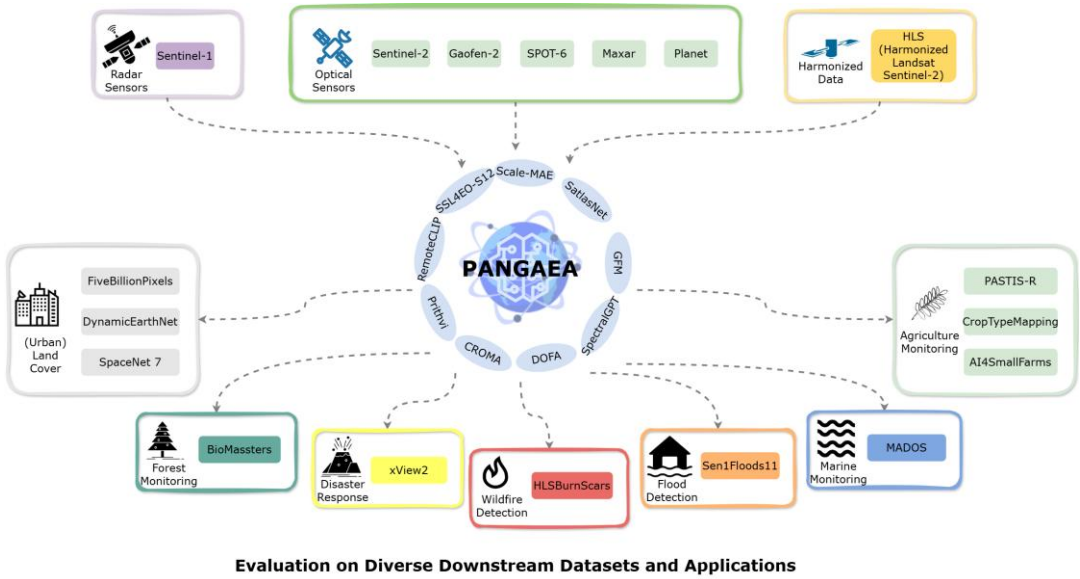
+ Memory & compute efficient
+ Enables partial adaptation of the pre-trained model

- Adds extra implementation complexity
- Adds extra hyperparameters

FM Evaluation Benchmarks

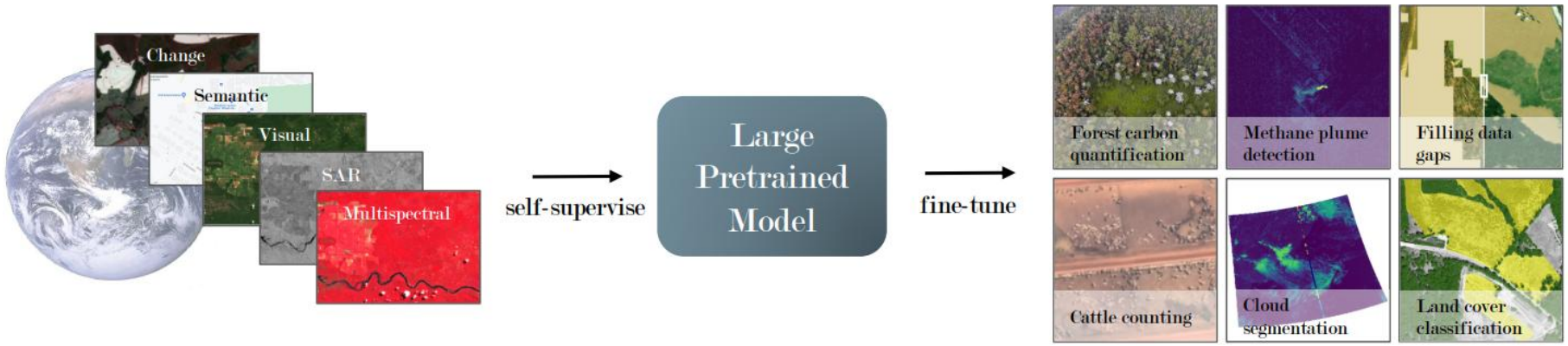


CVPR EarthVision Challenge



PANGAEA

GeoBench



How well do GeoFMs perform?

- **UNet is not dead yet!**

Model	BurnSr*	MADOS*	PASTIS	Sen1Fl11	FBP*	DEN*	CTM-SS	SN7*	AI4Farms*	Avg. mIoU	Avg. Rank
CROMA	82.42	67.55	32.32	90.89	51.83	38.29	49.38	59.28	25.65	55.29	6.61
DOFA	80.63	59.58	30.02	89.37	43.18	39.29	51.33	61.84	27.07	53.59	8.22
GFM-Swin	76.90	64.71	21.24	72.60	67.18	34.09	46.98	60.89	27.19	52.42	10.00
Prithvi 1.0 100M	83.62	49.98	33.93	90.37	46.81	27.86	43.07	56.54	26.86	51.00	11.00
RemoteCLIP	76.59	60.00	18.23	74.26	**69.19**	31.78	52.05	57.76	25.12	51.66	11.22
SatlasNet	79.96	55.86	17.51	90.30	50.97	36.31	46.97	61.88	25.13	51.65	10.67
Scale-MAE	76.68	57.32	24.55	74.13	67.19	35.11	25.42	**62.96**	21.47	49.43	11.44
SpectralGPT	80.47	57.99	35.44	89.07	33.42	37.85	46.95	58.86	26.75	51.87	10.11
S.-S12-MoCo	81.58	51.76	34.49	89.26	53.02	35.44	48.58	57.64	25.38	53.02	10.06
S.-S12-DINO	81.72	49.37	36.18	88.61	51.15	34.81	48.66	56.47	25.62	52.51	10.89
S.-S12-MAE	81.91	49.90	32.03	87.79	51.92	34.08	45.80	57.13	24.69	51.69	12.39
S.-S12-Data2Vec	81.91	44.36	34.32	88.15	48.82	35.90	54.03	58.23	24.23	52.22	10.72
UNet Baseline	**84.51**	54.79	31.60	**91.42**	60.47	**39.46**	47.57	62.09	**46.34**	57.58	4.89
ViT Baseline	81.58	48.19	38.53	87.66	59.32	36.83	44.08	52.57	38.37	54.13	10.28
TerraMindv1-B	82.42	69.52	40.51	90.62	59.72	37.87	**55.80**	60.61	28.12	58.35	3.94
TerraMindv1-L	82.93	**75.57**	**43.13**	90.78	63.38	37.89	55.04	59.98	27.47	**59.57**	**3.44**

Conclusion

Foundation Models: Pros & Cons

Advantages

- Strong generalization capabilities
- Little to no fine-tuning needed, works out of the box
- Off-shelf embeddings
- Label efficiency
- Cool branding

Limitations

- High inference cost
- High memory cost
- Good in many tasks, but not necessarily SOTA
- ViT limitations for pixel-level tasks
- Still require some labels

Conclusion

- **> 100 foundation model papers** in the past few years
- A trend towards **multi-modal foundation models**
  - Several sensors, vision/text models, etc.
- Interesting research ongoing in **climate foundation models**
- More research on **fair evaluation and model comparison** is needed
- FMs are **not** the solution to everything (yet?)
- The development of a truly **universal FM** remains an **open challenge**