Transductive Few-shot Learning for Ecosystem Classification from Satellite Time Series: A Systematic Study

Abstract

Mapping Earth's ecosystems using the IUCN Global Ecosystem Typology (GET) is increasingly vital for biodiversity and natural-capital reporting. However, GET-aligned mapping faces a persistent labeling bottleneck due to scarce reference labels and the continual need for new labels across regions or revisions. Few-shot learning is a promising solution to this challenge, but it has not been systematically evaluated for ecosystem functional classification within a globally consistent typology. This study introduces a benchmark of 24,009 georeferenced sites across South Korea for a single year, labeled with 24 GET functional groups across six realms through a rule-based integration of national thematic maps and paired with multivariate Harmonized Landsat and Sentinel-2 time series. The proposed IT-TIM first learns a transferable representation of phenological dynamics from the training classes. During inference, it refines per-class prototypes using the unlabeled query set, improving classification of previously unseen classes. With only five labeled sites per class, IT-TIM attains 0.857 accuracy on the realm-diverse partition and reaches 92.3% of the accuracy of a supervised reference under a matched six-class protocol, where the reference uses 40 times more labeled examples per target class. IT-TIM ranks above every baseline in all five class partitions, with modest margins over ProtoNet that are statistically significant in four of the five. Analysis of the objective reveals that class-balance enforcement, rather than per-query sharpening, drives the transductive gain, which is sustained only when queries are approximately balanced. The study also quantifies the accuracy cost when spectrally heterogeneous functional groups are not resolved individually.

Keywords: Few-shot learning, Transductive inference, Satellite image time series, IUCN Global Ecosystem Typology, Ecosystem mapping, Label-efficient mapping

1. Introduction

Land use and land cover are undergoing rapid transformation due to urbanization, agricultural intensification, and climate-driven shifts in ecosystem distribution. Monitoring these changes forms the basis for biodiversity assessment, carbon and ecosystem service accounting, conservation planning, and natural capital reporting within international frameworks. As the scope of land-cover information has expanded from pixel-level cover type identification to the characterization of functioning ecosystems, the demand has shifted from inventories of surface materials to consistent and functionally meaningful ecosystem classifications.

Within this context, the IUCN Global Ecosystem Typology (GET) (Keith et al., 2020b) has emerged as the international standard for ecosystem classification. It provides the first globally consistent hierarchical framework for categorizing Earth's ecosystems into functional groups across terrestrial, freshwater, marine, and transitional realms. The GET is increasingly adopted as the reference for global ecosystem accounting, Red List of Ecosystems assessments, and post-2020 biodiversity targets. However, implementing the GET at regional resolution, such as producing comprehensive maps of functional groups for a country or biome, remains considerably more complex than the terminology suggests. The global indicative maps accompanying the typology are intentionally broad in scale, and most regional products are generated by harmonizing existing national datasets rather than by directly classifying satellite imagery. In regions lacking suitable pre-existing products, expert interpretation remains necessary to address these gaps.

A further consideration is that the remote-sensing signal most relevant to GET functional groups is inherently temporal. Functional ecosystems are defined by dynamic processes such as seasonal cycles of vegetation growth, water inundation, and changes in surface state, rather than by instantaneous spectral signatures (Keith et al., 2022). For example, evergreen and deciduous forests may appear similar in a single mid-summer image but display distinct phenological patterns throughout the year. Intensive croplands, pastures, and semi-natural grasslands share broad spectral characteristics yet differ significantly in the timing and magnitude of greenup and senescence. Tidal flats, marshes, and open shorelines are best distinguished by their seasonal inundation patterns. Consequently, the discriminative information

resides in the temporal evolution of multispectral observations rather than in any single image acquisition.

Remote sensing is particularly well-suited to extracting these temporal signals at scale. Extensive research demonstrates that the frequent and globally consistent coverage provided by the Landsat and Sentinel-2 constellations, combined with deep-learning models that leverage phenological dynamics in satellite image time series, enables accurate land-cover discrimination (Pelletier et al., 2019; Rußwurm and Körner, 2020; Ismail Fawaz et al., 2020; Zhao et al., 2023). Nevertheless, two persistent obstacles hinder the operational mapping of GET-aligned functional groups. First, the deep representations underlying recent advances are data-intensive, typically requiring numerous labeled examples per class, while expert-provided ecosystem-functional labels remain inherently scarce (Zhao et al., 2023). Second, functional groups in the GET are spectrally subtle, with boundaries between ecologically distinct groups often limited to narrow climatic and spectral regions (Keith et al., 2022). Consequently, the discriminative signal is weaker than for broader cover classes, and the demand for precise labels is correspondingly higher. These factors together create a persistent labeling bottleneck, as each new region, season, or typology revision requires additional costly annotation efforts.

In response to these challenges, various strategies have been explored to alleviate the labeling bottleneck in remote sensing (Zhao et al., 2023). However, each approach relies on assumptions that the GET-aligned mapping problem does not naturally fulfill. Self-supervised pre-training (Wang et al., 2022) reduces dependence on labeled data for representation learning but still requires a significant supervised budget to adapt the representation to a specific functional typology. Transfer learning assumes the existence of a label-rich source domain from which knowledge can be transferred (Zhao et al., 2023), yet no such domain has been established for the GET. Aligned products in other regions are constructed from local datasets, and ecosystem-functional labels remain scarce across all regions. Active learning optimizes annotation budgets but still accumulates a substantial number of labels over time (Zhao et al., 2023), which is incompatible with a typology that is hierarchical, revisable, and continually extended to new regions. The GET-aligned mapping regime is therefore distinctive: each new region or revision introduces a new set of functional groups unseen during prior training, and each must be classified from a limited set of expert labels.

Within this regime, few-shot learning is particularly well-suited to the

task, as it aims to classify genuinely novel classes from only a small number of labeled examples by leveraging a transferable representation. Recent research in remote sensing has demonstrated promising results for few-shot scene classification and crop-type mapping (Rußwurm et al., 2020; Mohammadi et al., 2024; Li et al., 2024; Santiago Júnior, 2024). Additionally, transductive inference, which utilizes the unlabeled query set during prediction, has consistently outperformed inductive alternatives in related domains such as hyperspectral classification (Xi et al., 2025; Chen et al., 2025). However, to the best of our knowledge, no prior work has applied few-shot learning to ecosystem functional classification under a globally consistent typology, systematically evaluated it across the spectrum from no-representation baselines to transductive meta-learners, or investigated which component of the transductive objective is responsible for observed performance gains. The following three questions motivate this study: (i) whether few-shot learning can achieve near-supervised accuracy on GET-aligned classification at realistic label budgets, (ii) how competing inference rules perform under varying shot, class-count, spatial, and per-class conditions, and (iii) what mechanism drives the transductive advantage.

To address these questions, this study introduces a systematic evaluation using a newly developed benchmark of 24,009 georeferenced sites across South Korea. Each site is assigned to one of 24 IUCN GET functional groups spanning six realms, with labels inherited from a national GET-aligned map produced by a rule-based integration of thematic products (Lee et al., 2025), and is associated with a multivariate Harmonized Landsat and Sentinel-2 time series for the 2022 calendar year (Claverie et al., 2018). Leveraging this benchmark, the IT-TIM transductive few-shot framework is developed and assessed. IT-TIM first meta-learns a transferable representation of phenological dynamics from the observed class pool, then refines per-class prototypes at inference using the unlabeled query set to classify previously unobserved ecosystem classes. IT-TIM is evaluated against six reference methods, including no-representation classifiers, a linear probe, an inductive meta-learner, and two transductive meta-learners, with a fully supervised reference serving as a well-labeled reference. The analysis investigates performance across shot count, class count, spatial provenance, individual classes, and the weighting of the transductive objective.

The contributions of this study are as follows.

- A new few-shot benchmark for IUCN GET ecosystem functional clas-

sification from satellite time series is introduced, with labels derived from a rule-based integration of national thematic maps, along with a reproducible episodic evaluation protocol.

- Under a shared backbone and identical episode protocol, the spectrum from no-representation baselines to inductive and transductive meta-learners is systematically evaluated, with a supervised reference, and their behavior is characterized across shot, class-count, spatial, and per-class axes.

- Decomposition of the transductive objective identifies class-balance enforcement, rather than prediction sharpening, as the principal source of transductive gain on this task.

The remainder of the paper is organized as follows. Section 2 reviews related work in few-shot learning, satellite-imagery land-cover classification, and few-shot remote sensing. Section 3 describes the dataset and harmonized satellite time series. Section 4 formalizes the few-shot problem and the IT-TIM framework, and Section 5 details the evaluation protocol. Section 6 reports label efficiency and the comparison against baselines. Section 7 examines robustness, per-class behavior, and the decomposition of the transductive objective. Section 8 discusses the principal findings, practical implications, and limitations.

2. Related Work

2.1. Few-shot and transductive meta-learning

Few-shot learning aims to classify novel categories using only a limited number of labeled examples per class, typically by meta-learning a transferable representation across a distribution of episodic tasks. Metric-based approaches are predominant in this field. For example, Matching Networks (Vinyals et al., 2016) introduced the episodic training protocol and attention-weighted nearest-neighbor classification, while Prototypical Networks (Snell et al., 2017) further simplified this approach to nearest-centroid classification in a learned embedding space, computing a single prototype per class as the mean of its support embeddings. Optimization-based alternatives, such as MAML (Finn et al., 2017), instead meta-learn an initialization that facilitates rapid gradient-based adaptation. All of these methods share an

inductive assumption: each query is classified independently, relying solely on the labeled support set.

Transductive few-shot methods relax the inductive assumption by leveraging the statistical structure of the entire unlabeled query set during inference. For instance, SimpleShot (Wang et al., 2019) demonstrated that feature centering and ℓ_2 normalization of support and query embeddings can achieve competitive accuracy, underscoring the importance of feature transformations during inference. LaplacianShot (Ziko et al., 2020) constructs a nearest-neighbor graph over query embeddings and minimizes a Laplacian-regularized objective to encourage neighboring queries to share labels. Transductive Information Maximization (TIM) (Boudiaf et al., 2020) maximizes the mutual information between query features and predictions by combining a marginal-entropy term to prevent single-class collapse, a conditional-entropy term to sharpen predictions, and a support cross-entropy anchor. This work adopts the TIM objective and investigates its application in a setting where transductive few-shot learning has received limited attention: multivariate satellite time series of ecosystem types.

2.2. Land-cover classification from satellite imagery

Land-use and land-cover (LULC) classification from satellite imagery has been a persistent challenge in remote sensing, historically addressed using shallow machine-learning classifiers. Among these, random forests (Belgiu and Drăguţ, 2016) and support vector machines (Mountrakis et al., 2011) are the most widely used. Support vector machines are particularly valued for their ability to generalize from limited training samples, a common constraint in remote sensing applications (Mountrakis et al., 2011). The free, global, and frequent coverage of the Sentinel-2 and Landsat constellations has established medium-resolution multispectral data as the standard for operational LULC mapping.

However, single-date imagery often conflates classes with similar instantaneous spectral signatures. Incorporating the temporal dimension through satellite image time series (SITS) reduces this ambiguity by capturing phenological dynamics. As a result, deep learning has become the dominant paradigm for SITS classification. Temporal convolutional networks (Pelletier et al., 2019) apply one-dimensional convolutions along the time axis for crop mapping. Self-attention encoders (Rußwurm and Körner, 2020) model long-range temporal dependencies in raw, cloud-affected optical sequences. General-purpose architectures such as InceptionTime (Ismail Fawaz et al.,

2020) use parallel convolutions at multiple kernel sizes to capture patterns across temporal scales. InceptionTime is adopted as the backbone in this study because phenological signatures unfold over both short and long horizons, and multi-scale receptive fields are well-suited to this structure.

These deep models, however, require large amounts of data and often transfer poorly. Strong benchmark accuracy frequently does not generalize to regional or global deployment due to spectral similarity, spatial and scale variability, and seasonal inconsistency. One approach to address these limitations is to enrich satellite representations during training with auxiliary sources that are not available at deployment. For example, CarbonCLIP incorporates street-view semantics and month-level temporal context into satellite features through contrastive pretraining, enabling urban carbon emission prediction from satellite imagery alone at inference (Yang et al., 2026). While such methods reduce dependence on ground-level data during inference, they still require a large number of labeled examples for the target task. These challenges increase the cost of obtaining labels for each new region and class, precisely the scenario in which few-shot methods offer significant value.

2.3. Few-shot learning for ecosystem and land-cover mapping

The scarcity of labeled data has driven significant interest in few-shot learning within the field of remote sensing. The majority of research has concentrated on scene classification using single-date images, with methods including contrastive pre-training (Wu et al., 2023), masked reconstruction, and metric learning. These approaches are typically evaluated on established benchmarks such as EuroSAT and PatternNet. Few-shot learning has also been extensively explored for hyperspectral classification, where transductive methods that leverage query-set statistics often outperform inductive techniques (Xi et al., 2025; Chen et al., 2025). Recent studies in remote-sensing scene classification have begun to systematically compare inductive and transductive inference (Santiago Júnior, 2024).

In the context of land-cover mapping, Rußwurm et al. (2020) demonstrated that meta-learned models applied to Sentinel-2 imagery yield more accurate and consistent labels than manual annotation, establishing meta-learning as a viable approach for scenarios with limited labels. Recent benchmarks, such as the OpenEarthMap few-shot land-cover challenge (Li et al., 2024), prioritize identifying novel land-cover classes from limited labeled data, highlighting the need for label-efficient mapping. Notably, Mohammadi

et al. (2024) adapted few-shot methods for crop mapping using satellite image time series and found that evaluating on class-balanced query sets from the target domain can significantly overestimate performance. This finding indicates the importance of rigorous experimental design and transductive evaluation protocols that account for the structure of the unlabeled query set.

Recent research has increasingly focused on the IUCN Global Ecosystem Typology (GET) (Keith et al., 2022), which provides the first globally consistent framework for classifying Earth's ecosystems into functional groups spanning terrestrial, freshwater, marine, and transitional realms. The global maps associated with this typology illustrate broad distribution patterns rather than detailed, fine-scale occurrences (Keith et al., 2020a). Most large-scale GET-aligned products are created by harmonizing existing datasets, rather than through direct classification of satellite imagery. Remote sensing applications targeting GET functional groups have primarily addressed individual, spectrally distinctive groups, such as the continental-scale mapping of tidal flats using the Landsat archive (Murray et al., 2012), or have utilized region-specific, expert-guided pipelines to differentiate functional ecosystems with similar climatic and spectral characteristics (Wells et al., 2025). These studies indicate that operationalizing the GET with remote sensing is possible, but remains resource-intensive. Functional types frequently display subtle spectral differences, and medium-resolution sensors often lack the capacity to resolve fine-grained distinctions. As a result, GET-aligned mapping in regions lacking suitable products still relies on expert labeling, which perpetuates the labeling bottleneck. Recently, Lee et al. (2025) produced a national-scale GET-aligned EFG map for the Republic of Korea by integrating authoritative national thematic datasets. However, this assessment assumes that sufficient labels are available for each class. The challenge of identifying the minimum number of labels necessary when functional groups are previously unobserved remains unresolved. Few-shot classification of ecosystem functional types under this typology, as well as analysis of the factors influencing transductive performance on such data, has not, to our knowledge, been systematically addressed in the context of satellite time series.

3. Data

3.1. Study area and ecosystem labels

The study area comprises the South Korean peninsula. Within this region, a dataset of 24,009 georeferenced sample sites was compiled, with each site assigned to one of 24 ecosystem functional groups adapted from the IUCN Global Ecosystem Typology (GET) v2.1 (Keith et al., 2022).

Label provenance. The labels originate from a national-scale IUCN GET ecosystem functional group map for the Republic of Korea (Lee et al., 2025), developed in two stages. The first stage involves selecting functional groups relevant to the national context through expert consultation and establishing an explicit classification framework for each group, specifying land-cover correspondences and functional criteria such as climatic thresholds and hydrological conditions. In the second stage, this framework is applied to authoritative national geospatial products: the national Land Cover Map (Ministry of Environment, 2022; 41 classes), the Forest Type Map (Korea Forest Service, 2022), the FarmMap cropland layer (Ministry of Agriculture, Food and Rural Affairs, 2022), and the National Wetlands Inventory (National Institute of Ecology, 2022). Functional attribution is based on gridded mean air temperature (MK-PRISM, Korea Meteorological Administration, 2015–2019), surface-water occurrence (Pekel et al., 2016), river geometry (Linke et al., 2019), and a national digital elevation model. Sample sites are selected from the resulting map using stratified random sampling within each class. The labels used in this study are taken from this rule-based integration product. Complete classification criteria and source details for all 24 classes are provided in Appendix A.

Nature of the labels. The labels are map-derived rather than independently adjudicated at the site level. Expert judgment is incorporated through the design of the classification framework rather than through per-site interpretation. This approach has two main consequences. First, label consistency is high because identical rules are applied throughout; however, any systematic error in the underlying thematic maps propagates to the labels. The resulting uncertainty is greatest in transitional and mixed-use zones, where the source products themselves are least certain. Second, and importantly for the present evaluation, the reference layers are based on aerial orthophotography, field surveys, cadastral records, and ground meteorological observations, rather than intra-annual medium-resolution optical time series. The labels

are therefore methodologically independent of the HLS predictors used in this study, ensuring that the benchmark does not simply reproduce an existing Landsat- or Sentinel-derived product. The only exception is a layer used to distinguish permanent from seasonal water bodies, which incorporates long-term water-occurrence frequency derived from the Landsat archive (1984–2021) rather than the intra-annual phenological signal on which the classifier operates.

Regional adaptation of the typology. The source framework defines functional groups at IUCN GET Level 3, which are adapted in three ways. First, two Level-3 groups are subdivided using distinctions available in the source layers: montane forests (T2.1) are separated by stand type from the Forest Type Map, and shorelines (MT1) are divided into tidal flats (MT1.2), defined as the intersection of the tidal-flat class from the detailed Land Cover Map with mudflat substrate in the national coastal sediment map, and the remaining shoreline types. Second, the marine open ocean (M2.1) is included from the marine class of the national Land Cover Map, extending beyond the terrestrial scope of the original framework. Third, the benchmark consists of 24 classes, restricted to functional groups for which national extent supports episodic evaluation. Two of these, freshwater lakes and the remaining coastal shorelines, are labeled at the parent level because the source layers do not resolve their constituent groups individually. The retained set covers all six IUCN GET realms present in the Republic of Korea.

Class composition. The 24 classes are distributed across six IUCN GET realms present in the region: Terrestrial (6 classes), Anthropogenic (5), Freshwater (7), Marine (1), Marine–Terrestrial (2), and Transitional Freshwater (3). The complete class list, including IUCN GET codes and descriptions, is provided in Table 1. 22 classes contain exactly 1,000 sites, and the remaining two contain 1,003 and 1,006. Figure 1 presents the spatial distribution of the sample sites.

3.2. Satellite time series

For each sample site, a multivariate time series was constructed from the Harmonized Landsat and Sentinel-2 (HLS) product, version 2 (Claverie et al., 2018), for the 2022 calendar year. The Landsat-8/9 OLI collection (HLSL30) and Sentinel-2A/B MSI collection (HLSS30), both providing 30 m nadir BRDF-adjusted surface reflectance (NBAR), were harmonized into

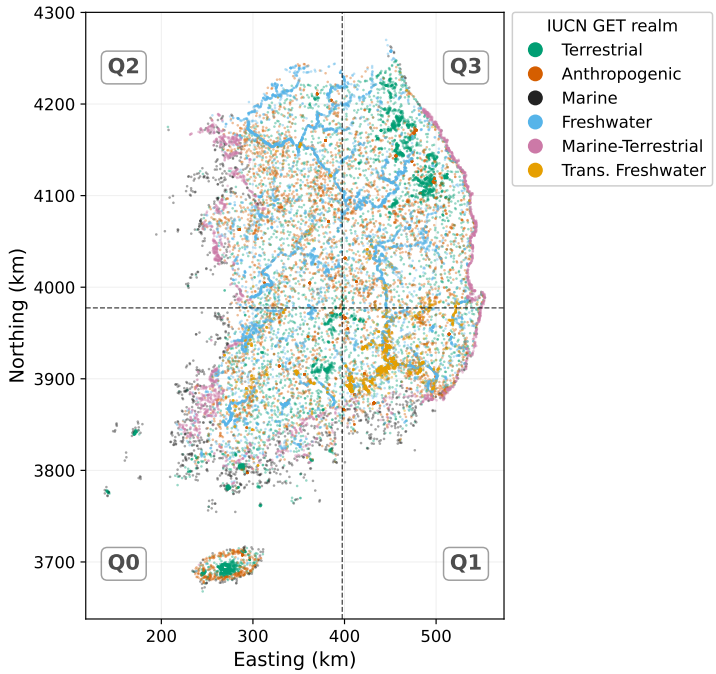


Figure 1: Spatial distribution of the 24,009 sample sites across South Korea, colored by IUCN GET realm. Coordinates are in EPSG:32652 (UTM zone 52N). Dashed lines indicate the four spatial quadrants (Q0–Q3) used in the spatial-transfer analysis.

a single, cross-sensor-consistent record with an effective revisit interval of approximately 2 to 3 days. Sentinel-2 bands B8A, B11, and B12 were mapped to the near-infrared and two shortwave-infrared bands (SR_{B5} to SR_{B7}) to align with the Landsat band designations.

Scenes with reported cloud coverage exceeding 80% were excluded. For each retained scene, pixels identified by the HLS Fmask quality band as cloud, cloud-adjacent, cloud-shadow, or snow/ice were masked, while water-flagged pixels were retained. Multiple same-day observations at a site were aggregated by computing the pixel-wise median to produce a single daily value. To address irregular temporal spacing, daily values were grouped into temporal cycles and resampled to a uniform grid of $T = 122$ time steps over the year (approximately 3-day intervals). Missing values at time steps without valid observations were imputed using linear interpolation between neighboring cycles, ensuring no remaining missing data.

At each time step, the observation consists of 11 spectral features: seven

Table 1: The 24 ecosystem functional groups, adapted from the IUCN Global Ecosystem Typology v2.1 (Keith et al., 2022) for South Korea.

Idx	GET code	Realm	Description
0	T2.1.1	Terrestrial	Subalpine coniferous forest
1	T2.1.2	Terrestrial	Subalpine mixed forest
2	T2.2.1	Terrestrial	Temperate deciduous broadleaf forest
3	T2.2.2	Terrestrial	Temperate evergreen coniferous forest
4	T2.2.3	Terrestrial	Temperate mixed forest
5	T2.4	Terrestrial	Warm-temperate evergreen broadleaf forest
6	T7.1	Anthropogenic	Croplands (dryland)
7	T7.2	Anthropogenic	Intensive pasture
8	T7.3	Anthropogenic	Orchards
9	T7.4	Anthropogenic	Urban / built-up
10	T7.5	Anthropogenic	Semi-natural grasslands and fallows
11	M2.1	Marine	Marine open ocean
12	F1.1	Freshwater	Permanent upland streams
13	F1.2	Freshwater	Permanent lowland streams
14	F1.3	Freshwater	Freeze–thaw rivers
15	F1.4	Freshwater	Seasonal upland streams
16	F1.5	Freshwater	Seasonal lowland streams
17	F2.2/4/5	Freshwater	Freshwater lakes (combined)
18	F3.3	Freshwater	Rice paddies
19	MT1.1/3/4	Marine–Terrestrial	Coastal shorelines (aggregated)
20	MT1.2	Marine–Terrestrial	Tidal flats
21	TF1.2	Trans. Freshwater	Forested wetlands
22	TF1.3	Trans. Freshwater	Marshes
23	TF1.7	Trans. Freshwater	Mountain wetlands

surface-reflectance bands (coastal-aerosol, blue, green, red, near-infrared, and two shortwave-infrared bands, denoted SR_{B1} to SR_{B7}) and four spectral indices: the Normalized Difference Vegetation Index (NDVI), Normalized Difference Built-up Index (NDBI), Modified Normalized Difference Water Index (MNDWI), and an Urban Index (UI). To incorporate local spatial context, these 11 features are computed both for the center pixel and as the mean over its surrounding 3×3 pixel neighborhood, yielding $11 \times 2 = 22$ time-varying channels per time step. Each sample is therefore represented as a matrix $\mathbf{x} \in \mathbb{R}^{T \times C}$ with $T = 122$ and $C = 22$. Three static (time-invariant) features are also available for each site: elevation (DEM) and the two projected coordinates. In the main experiments, the no-static variant is reported, in which the encoder receives only the 22 time-varying channels, since several of the classification criteria above are themselves defined by elevation and location.

3.3. Preprocessing and normalization

The seven surface-reflectance bands are retained on their native reflectance scale, with values predominantly in the range $[0, 1]$; a small fraction falls slightly below zero due to atmospheric correction artifacts. The vegetation, built-up, and water indices (NDVI, NDBI, MNDWI) are bounded in $[-1, 1]$ by construction. The Urban Index is reported in its raw, unclipped form according to the conventional index formula; its values span approximately $[-24.9, +6.1]$, with large-magnitude outliers resulting from division by near-zero denominators, and are passed to the encoder without further preprocessing. Preliminary experiments confirmed that removing or rescaling the UI channel does not materially affect results, consistent with the encoder's per-block normalization. The static elevation and coordinate features are standardized to zero mean and unit variance, while the 22 time-varying channels are passed to the encoder at their recorded scale. After the HLS compositing and gap-filling step, no missing values remain in the time-varying channels.

4. Method

This section introduces IT-TIM, a transductive few-shot framework for satellite time-series classification. Figure 2 provides an overview of the framework. Subsection 4.1 formalizes the few-shot episodic problem. Subsection 4.2 describes the multivariate InceptionTime encoder, which serves as the representation backbone. Subsection 4.3 outlines the episodic meta-training procedure. Subsection 4.4 introduces transductive information maximization as the inference-time refinement objective.

4.1. Problem formulation

This work addresses the few-shot classification of multivariate satellite time series. Each input sample is represented as a matrix $\mathbf{x} \in \mathbb{R}^{T \times C}$, where T denotes the number of time steps and C indicates the number of spectral-temporal channels observed at a single ecosystem site. The classification target $y \in \mathcal{Y}$ corresponds to one of $|\mathcal{Y}|$ ecosystem classes defined by a hierarchical typology.

The class set $\mathcal{Y}$ is divided into a meta-train pool $\mathcal{Y}_{\text{train}}$ and a meta-test pool $\mathcal{Y}_{\text{test}}$, where $\mathcal{Y}_{\text{train}} \cap \mathcal{Y}_{\text{test}} = \emptyset$. The meta-train pool is utilized to learn a generic time-series representation. The meta-test pool comprises held-out classes, and their samples are used exclusively during evaluation.

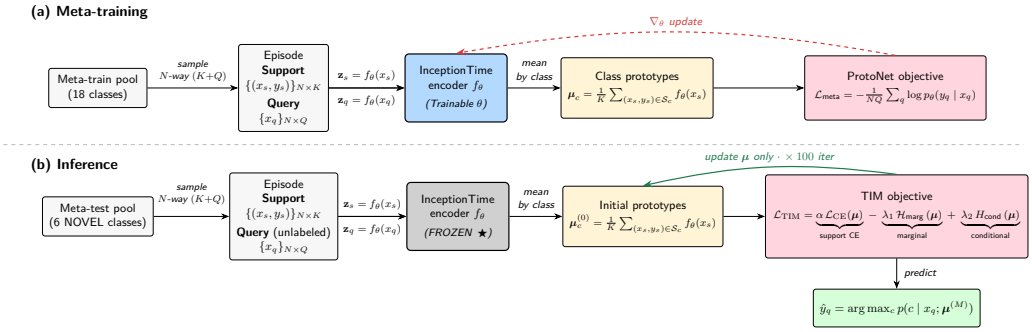


Figure 2: Overview of the IT-TIM framework. The InceptionTime encoder is meta-trained on the seen-class pool with the prototypical-network objective (a); at inference, the encoder is frozen and class prototypes are refined by the TIM objective (b).

Each evaluation episode $\mathcal{E} = (\mathcal{S}, \mathcal{Q})$ is constructed as follows. We first sample N classes uniformly at random from $\mathcal{Y}_{\text{test}}$. For each sampled class $c \in \{1, \ldots, N\}$, K labeled support examples and Q query examples are drawn from previously unseen sample indices, resulting in the support set $\mathcal{S} = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{NK}$ and the query set $\mathcal{Q} = \{(\mathbf{x}_j^q, y_j^q)\}_{j=1}^{NQ}$. The few-shot classifier receives only the inputs and labels in $\mathcal{S}$, along with the query inputs in $\mathcal{Q}$, and is required to predict the query labels $\{y_j^q\}_{j=1}^{NQ}$. This setup is referred to as the N-way K-shot Q-query episode.

In the context of this study, an N-way episode refers to classification among N ecosystem functional groups. The parameter K shots denotes the number of labeled sample sites available for each group, while Q queries correspond to the unlabeled sample sites requiring IUCN GET class prediction. The primary experimental configuration employs $N = 5$, $K = 5$, and $Q = 15$ per episode (Subsection 5.1), which simulates a realistic labeling budget in which five labeled sites are available per functional group and the classifier assigns labels to an additional fifteen unlabeled sites.

The few-shot regime is defined by the small support count K (typically $K \leq 50$ per class) and the disjoint class partition between meta-train and meta-test, which together require that the learned representation generalizes to classes never seen during training.

4.2. Multivariate InceptionTime encoder

We utilize the InceptionTime architecture (Ismail Fawaz et al., 2020) as the representation backbone, extending it to support multivariate inputs.

The encoder $f_\theta : \mathbb{R}^{T\times C} \to \mathbb{R}^D$ transforms a multichannel time series of length T into a fixed-dimensional embedding in $\mathbb{R}^D$.

The encoder comprises a stack of Inception blocks, each performing parallel convolutions across multiple temporal scales. Each block processes its input $\mathbf{h} \in \mathbb{R}^{T\times C'}$ by first applying a 1×1 bottleneck convolution to reduce the channel dimension. This is followed by three parallel one-dimensional convolutions with kernel sizes $\{k_1, k_2, k_3\}$ and a 1×1 max-pooled branch. The outputs from all four branches are concatenated along the channel dimension to produce the block output. This multi-scale approach enables the encoder to capture phenological patterns at various temporal resolutions. Short-term variations, such as cloud-cover anomalies and weekly leaf-flush events, are detected by the smallest kernel (k_1). The medium kernel (k_2) captures monthly seasonal transitions, while the largest kernel (k_3) identifies quarterly phenological cycles, including greenup, peak greenness, senescence, and dormancy.

A residual connection is incorporated after every three Inception blocks, consistent with the design of Ismail Fawaz et al. (2020), to facilitate gradient flow in deep architectures. The final block output $\mathbf{h}^{(L)} \in \mathbb{R}^{T\times C^{(L)}}$ is then reduced to a single D-dimensional embedding using global average pooling along the temporal axis:

$$f_\theta(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T} \mathbf{h}_t^{(L)} \in \mathbb{R}^D. \tag{1}$$

Global average pooling confers three key properties for time-series classification. First, it ensures invariance to the temporal alignment of input observations, as each of the T observations contributes equally to the embedding. Second, it serves as a regularizer by discouraging dependence on specific time steps and promoting the extraction of patterns distributed across the observation window. Third, it produces a fixed-length representation regardless of the input length T, allowing the encoder to process time series of varying durations. The resulting embedding $f_\theta(\mathbf{x})$ defines the representation space for all subsequent few-shot operations, including prototype computation, transductive refinement, and prediction.

4.3. Episodic meta-training

Prototypical Networks (Snell et al., 2017) constitute a metric-based method for few-shot classification. For each class in an episode, a single prototype

vector is computed as the mean of the embeddings of its labeled support samples. Classification of a query involves assigning it to the class whose prototype is nearest in Euclidean distance within the embedding space. The episodic training procedure encourages the encoder to produce embeddings such that the support-mean prototype of each class is located near the embeddings of all samples from that class and far from those of other classes. This nearest-prototype rule is computationally efficient during inference and, when paired with a well-trained encoder, provides a robust baseline for few-shot classification (Wang et al., 2019; Chen et al., 2019).

The InceptionTime encoder f_θ is meta-trained using the Prototypical Networks objective (Snell et al., 2017) on episodes sampled from the meta-training class pool. This objective exploits the structure of few-shot episodes to learn representations in which samples from the same class form clusters in $\mathbb{R}^D$ and samples from different classes are distinctly separated.

For each meta-training episode $\mathcal{E}_{\text{train}} = (\mathcal{S}_{\text{train}}, \mathcal{Q}_{\text{train}})$, a prototype $\boldsymbol{\mu}_c \in \mathbb{R}^D$ is computed for each of the N episode classes as the mean of its support embeddings:

$$\boldsymbol{\mu}_c = \frac{1}{K} \sum_{(\mathbf{x}_i^s, y_i^s) \in \mathcal{S}_c} f_\theta(\mathbf{x}_i^s), \quad \mathcal{S}_c = \{(\mathbf{x}_i^s, y_i^s) \in \mathcal{S}_{\text{train}} : y_i^s = c\}. \tag{2}$$

The selection of the support mean as the prototype relies on the assumption that, in a well-structured embedding space, samples from the same class cluster around their class centroid. The empirical mean serves as a maximum-likelihood estimator of this centroid under isotropic Gaussian intra-class noise (Snell et al., 2017). While this estimator is unbiased, its variance is inversely proportional to K. As a result, the support-mean prototype exhibits increased noise as K decreases, thereby motivating the prototype refinement discussed in Subsection 4.4.

The classification probability for a query $\mathbf{x}_j^q$ assigned to class c is defined using a softmax function over the Euclidean distances to all N prototypes:

$$p_\theta(y_j^q = c \,|\, \mathbf{x}_j^q, \mathcal{S}_{\text{train}}) = \frac{\exp(-\|f_\theta(\mathbf{x}_j^q) - \boldsymbol{\mu}_c\|_2^2)}{\sum_{c'=1}^{N} \exp(-\|f_\theta(\mathbf{x}_j^q) - \boldsymbol{\mu}_{c'}\|_2^2)}. \tag{3}$$

The meta-training loss is defined as the negative log-likelihood of the true query labels, summed over all queries within the episode:

$$\mathcal{L}_{\text{meta}}(\theta; \mathcal{E}_{\text{train}}) = -\frac{1}{NQ} \sum_{(\mathbf{x}_j^q, y_j^q) \in \mathcal{Q}_{\text{train}}} \log p_\theta(y_j^q \,|\, \mathbf{x}_j^q, \mathcal{S}_{\text{train}}). \tag{4}$$

The encoder parameters θ are updated using stochastic gradient descent on $\mathcal{L}_{\text{meta}}$ across multiple meta-training episodes. Each episode is constructed by sampling a new set of N classes from $\mathcal{Y}_{\text{train}}$ with disjoint support and query splits. The resulting representation captures inter-class separability that generalizes from $\mathcal{Y}_{\text{train}}$ to $\mathcal{Y}_{\text{test}}$, assuming consistency in class taxonomy across both pools.

During inference, f_θ is fixed and used solely as a feature extractor. The subsequent transductive refinement procedure operates exclusively within the resulting embedding space.

4.4. Transductive Information Maximization (TIM) inference

The IT-TIM inference process proceeds as follows. After meta-training, the encoder f_θ generates embeddings for each support and query sample. Initial class prototypes are computed as support-mean centroids, as detailed in Subsection 4.3. When the number of labeled samples is limited (for example, $K \leq 5$), these prototypes often represent noisy approximations of the true class centroids within the embedding space. To mitigate this issue, IT-TIM treats the prototypes as trainable parameters and refines them during each evaluation episode by utilizing both the labeled support set and the geometric structure of the unlabeled query embeddings. Although the classifier does not access query labels, the spatial arrangement of query embeddings reveals information about the underlying class structure. Queries belonging to the same class tend to cluster in compact regions, and the joint distribution of all queries helps define class boundaries that noisy support prototypes alone cannot capture. Throughout this procedure, the encoder remains frozen, and only the prototype vectors are updated during each episode.

The IT-TIM objective formalizes this idea through three components: a support cross-entropy anchor that ties the prototypes to the labeled support set, and two query-side entropy regularizers that exploit the query distribution. Formally, let $\boldsymbol{\mu} = \{\boldsymbol{\mu}_c\}_{c=1}^{N}$ denote the prototype parameters being optimized. For any support or query sample $\mathbf{x}$, define the soft assignment over the N classes via a tempered softmax:

$$p(c \,|\, \mathbf{x}; \boldsymbol{\mu}) = \frac{\exp(-\tau \cdot \|f_\theta(\mathbf{x}) - \boldsymbol{\mu}_c\|_2^2)}{\sum_{c'=1}^{N} \exp(-\tau \cdot \|f_\theta(\mathbf{x}) - \boldsymbol{\mu}_{c'}\|_2^2)}, \tag{5}$$

where $\tau > 0$ denotes a temperature parameter that controls the sharpness of the assignment. The TIM objective consists of three components:

$$\mathcal{L}_{\text{TIM}}(\boldsymbol{\mu}) = \alpha \cdot \mathcal{L}_{\text{CE}}(\boldsymbol{\mu}) - \lambda_1 \cdot \mathcal{H}_{\text{marg}}(\boldsymbol{\mu}) + \lambda_2 \cdot \mathcal{H}_{\text{cond}}(\boldsymbol{\mu}), \tag{6}$$

where each term is defined as follows.

The first term anchors the prototypes to the support labels using cross-entropy:

$$\mathcal{L}_{\mathrm{CE}}(\boldsymbol{\mu}) = -\frac{1}{NK} \sum_{(\mathbf{x}_i^s, y_i^s) \in \mathcal{S}} \log p(y_i^s \mid \mathbf{x}_i^s; \boldsymbol{\mu}). \tag{7}$$

This term ensures that the refined prototypes are consistent with the labeled support information.

The second term is the marginal (Shannon) entropy of the query predictions, which enters the objective in Equation 6 with a negative sign. This term encourages the prototype configuration to assign queries to all N classes in approximately balanced proportions:

$$\mathcal{H}_{\mathrm{marg}}(\boldsymbol{\mu}) = -\sum_{c=1}^{N} \bar{p}_c(\boldsymbol{\mu}) \log \bar{p}_c(\boldsymbol{\mu}), \quad \bar{p}_c(\boldsymbol{\mu}) = \frac{1}{NQ} \sum_{\mathbf{x}_j^q \in \mathcal{Q}} p(c \mid \mathbf{x}_j^q; \boldsymbol{\mu}). \tag{8}$$

Maximizing this term discourages prototype configurations that concentrate query predictions on a limited subset of classes. In the original formulation, this approach is motivated as a safeguard against degenerate single-class solutions (Boudiaf et al., 2020). This effect is most pronounced when the support count K is small, and the support-mean prototypes are poorly calibrated.

The third term is the mean conditional entropy of individual query predictions. This term encourages confident classification of each query:

$$\mathcal{H}_{\mathrm{cond}}(\boldsymbol{\mu}) = -\frac{1}{NQ} \sum_{\mathbf{x}_j^q \in \mathcal{Q}} \sum_{c=1}^{N} p(c \mid \mathbf{x}_j^q; \boldsymbol{\mu}) \log p(c \mid \mathbf{x}_j^q; \boldsymbol{\mu}). \tag{9}$$

Minimizing this term, as indicated by the sign in Equation 6, drives the query predictions toward one-hot assignments and sharpens class boundaries in regions of the embedding space populated by queries.

The combined objective in Equation 6 leverages query-side structure in two complementary ways. The marginal-entropy term ensures global class balance across the query set, while the conditional-entropy term ensures local sharpness of individual query assignments. The support cross-entropy anchors both to the labeled support information.

The prototype parameters $\boldsymbol{\mu}$ are initialized to the support-class centroids,

$$\boldsymbol{\mu}_c^{(0)} = \frac{1}{K} \sum_{(\mathbf{x}_i^s, y_i^s) \in \mathcal{S}_c} f_\theta(\mathbf{x}_i^s), \tag{10}$$

The prototypes are then iteratively refined using Adam gradient updates on $\mathcal{L}_{\text{TIM}}$ over a fixed number of iterations M. The encoder weights θ remain unchanged. After M iterations, the final prototypes $\boldsymbol{\mu}^{(M)}$ are used to compute query predictions with the same softmax formulation:

$$\hat{y}_j^q = \arg \max_{c \in \{1,\ldots,N\}} p(c \mid \mathbf{x}_j^q; \boldsymbol{\mu}^{(M)}). \tag{11}$$

5. Experimental Setup

5.1. Evaluation protocol

Class splits. The 24 IUCN GET classes in the dataset are divided into 18 meta-train and 6 meta-test classes using uniform random selection. To evaluate sensitivity to this partitioning, five distinct partitions are generated using random seeds 0-4. Although each split is randomly assigned, the resulting compositions represent a range of ecologically meaningful scenarios. These include forest-rich combinations dominated by spectrally similar canopy types, water-dominated combinations of spectrally confusable hydrological ecosystems, and realm-balanced compositions spanning all six IUCN GET realms in the dataset. For clarity, each split is labeled according to its dominant ecological character, as summarized in Table 2. These partitions are not drawn from a broader set of candidates, and the descriptive labels are assigned retrospectively.

Table 2: The five meta-test class partitions used in this study.

Split	Name	Realm composition	Meta-test classes (IUCN GET codes)
A	Forest-rich	T:3, A:1, F:1, MT:1	T2.2.1, T2.2.2, T2.4, T7.2, F1.1, MT1.2
B	Agricultural	T:1, A:2, F:2, MT:1	T2.2.1, T7.1, T7.2, F1.2, F3.3, MT1.2
C	Realm-diverse	All 6 realms (1 each)	T2.1.1, T7.2, M2.1, F1.1, MT1.1/MT1.3/MT1.4, TF1.3
D	Human-modified	T:2, A:2, MT:1, TF:1	T2.2.2, T2.2.3, T7.4, T7.5, MT1.2, TF1.2
E	Water-dominated	A:1, F:3, MT:1, TF:1	T7.5, F1.2, F1.3, F2.2/F2.4/F2.5, MT1.2, TF1.2

Realm abbreviations: T (Terrestrial), A (Anthropogenic), M (Marine), F (Freshwater), MT (Marine-Terrestrial), TF (Transitional Freshwater).

Each meta-test class contains approximately 1,000 labeled sites, from which disjoint support and query subsets are selected for each episode. The primary results presented in subsequent sections are reported as the mean across all five splits, ensuring that conclusions are not restricted to any single class subset. Notably, Split C is the only partition that contains one class from each of the six IUCN GET realms in the dataset, resulting in

the most ecologically balanced composition. Therefore, Split C serves as the representative reference for analyses requiring per-class detail.

Episode protocol. In each evaluation episode (Vinyals et al., 2016), $N = 5$ classes are drawn uniformly at random from the 6-class meta-test pool. For each selected class, K labeled support examples and $Q = 15$ query examples are sampled from previously unseen sample indices within that class. Support and query subsets within an episode remain disjoint through permutation-based sampling. The model is provided with the NK support labels and generates predictions for the $NQ = 75$ queries. Results are reported for $K \in \{1, 5, 10, 20, 50\}$, with evaluation conducted over 600 random episodes per (split-seed, model-seed, K) combination.

Statistical aggregation. For each split, five model seeds (0 to 4) are used, with each seed generating a separately meta-trained backbone characterized by distinct weight initialization and episode sampling order. Consequently, each cell in the main results tables aggregates a total of $5 \times 5 \times 600 \times 75 = 1{,}125{,}000$ classification decisions. These classification decisions do not represent an equivalent number of independent observations. Accuracy is first averaged over the 600 episodes for each seed, and because the five seeds share a class partition, they are considered nested within splits. Within a single split, the statistical unit is the model seed ($n = 5$); across splits, seed-level accuracies are first averaged within each split, and the statistical unit is the split ($n = 5$). Paired t-tests compare methods on the corresponding unit. Confidence intervals are 95% intervals based on the t distribution, and effect sizes d_z are computed on the same unit.

5.2. Models and training details

Backbone meta-training. The InceptionTime encoder is configured with a depth of 6, 32 filters per layer, kernel sizes of $\{10, 20, 40\}$, and produces a 128-dimensional output embedding following global average pooling along the temporal axis. Meta-training is conducted using the Prototypical Networks objective (Snell et al., 2017) on $N = 5$-way, $K = 5$-shot, $Q = 15$-query episodes sampled from the 18-class meta-training pool, over 10,000 episodes with the Adam optimizer at a learning rate of 10^{-3}. During inference, the encoder weights are frozen and remain consistent across all subsequent inference methods.

Transductive inference (IT-TIM). During evaluation, prototypes are initialized as support-class centroids within the frozen embedding space and subsequently refined using the Transductive Information Maximization objective for 100 gradient steps at a learning rate of 10^{-4}. This objective integrates a support cross-entropy term (weight $\alpha = 1$), a marginal-entropy term that promotes balanced class assignment across the query set (weight $\lambda_1 = 1$), and a conditional-entropy term that encourages confident classification of queries (weight $\lambda_2 = 0.1$). The softmax temperature is fixed at $\tau = 15$. All hyperparameter values are consistent with the defaults reported by Boudiaf et al. (2020).

Baselines. Six baselines are used as references for IT-TIM. Both the baselines and IT-TIM utilize the same backbone variant and follow identical episode protocols. This design ensures that any observed differences in accuracy are attributable to methodological factors rather than to data or architectural variations.

- *Nearest Centroid (NC)* (Tibshirani et al., 2002) computes prototypes as the mean of support samples in the raw $T \times C$ input space, which is flattened to a $T \cdot C$-dimensional vector. Each query is then assigned to the nearest centroid based on Euclidean distance.

- *1-Nearest Neighbor with Dynamic Time Warping (DTW-1NN)* (Sakoe and Chiba, 1978) limits comparison to the NDVI univariate time series, using a Sakoe-Chiba band with a width of 10.

- *Linear probe* (Chen et al., 2019) fits an Adam-optimized linear head on the same frozen meta-trained backbone as ProtoNet and IT-TIM (200 epochs, learning rate 10^{-2}, ℓ_2 weight decay 10^{-4}).

- *Prototypical Networks (ProtoNet)* (Snell et al., 2017) utilize the same meta-trained backbone as IT-TIM, with predictions generated by nearest-prototype assignment in the frozen embedding space.

- *SimpleShot* (Wang et al., 2019) applies feature centering and ℓ_2 normalization to both support and query embeddings prior to nearest-prototype prediction.

- *LaplacianShot* (Ziko et al., 2020) constructs a k-nearest-neighbor graph over query embeddings ($k = 3$) and refines predictions using a Laplacian regularizer ($\lambda_{\text{Lap}} = 1.0$, 20 iterations) that encourages neighboring queries to share class assignments.

All baselines use the same 22-channel HLS input and identical episode protocols, allowing direct attribution of accuracy differences to method design rather than data or evaluation setup. The three static features are excluded from this input throughout.

5.3. Supervised reference

A fully supervised classifier, trained on 200 labeled sites per meta-test class (approximately 1,200 labels per split), provides a K-agnostic reference for the accuracy achievable under a well-labeled regime. For each split and model seed, these 200 sites per class are drawn once at random and permanently removed from the episode pool; all support and query samples used in evaluation, for both the few-shot methods and the reference, are drawn only from the remaining sites.

The supervised reference utilizes the same InceptionTime architecture as IT-TIM and is trained from random initialization using cross-entropy loss for 100 epochs with the Adam optimizer (learning rate 10^{-3}, weight decay 10^{-4}, batch size 128), without employing an episodic structure. This supervised reference represents a standard supervised model trained with abundant per-class labels, in contrast to an episodic few-shot classifier.

The factor of 40 between the two regimes refers only to the number of labeled examples per target class; IT-TIM additionally relies on a backbone meta-trained on the 18 meta-training classes, which the reference does not use. The reference does not use episode-level support sets and, during evaluation, classifies query examples directly with its trained classifier. Therefore, the percentage of supervised reference performance reported in subsequent sections should be interpreted as the proportion of attainable performance that few-shot inference achieves relative to this well-labeled reference, rather than as a strict upper bound on the identical episodic task. Differences in learning setup, such as the absence of episodic training, result from this distinction in learning regime.

5.4. Metric reporting

All metrics reported in the main text are derived from cumulative confusion matrices computed across the entire query set for each evaluation cell,

rather than being averaged per episode. This cumulative approach provides greater stability compared to per-episode aggregation, especially at low shot counts where individual episodes may include few representatives of certain classes. The main tables present accuracy, macro-F1, and Cohen's κ (chance-corrected agreement; Landis and Koch, 1977).

6. Main Results

This section evaluates the accuracy of IT-TIM and analyzes the sources of that accuracy. Subsection 6.1 examines the K-shot accuracy curve to determine a practical label budget. Subsection 6.2 compares the resulting accuracy to a supervised reference under two protocols, as the reference and the episodic classifiers are not scored on the same episodes. Subsection 6.3 benchmarks IT-TIM against six reference methods at the selected operating point, initially on the realm-diverse partition and subsequently across all five partitions, including the inference cost associated with each method. Subsection 6.4 isolates the contributions of the learned representation and the inference rule through a controlled ablation. All results are based on the frozen backbones and episode protocol described in Section 5.

6.1. Label efficiency

The label efficiency of IT-TIM is assessed using the K-shot accuracy curve, evaluated on the 5-way meta-test pool for $K \in \{1, 5, 10, 20, 50\}$. Table 3 presents the mean results across the five class splits introduced in Table 2.

Also, Table 4 provides the corresponding results for the realm-diverse split for detailed reference.

First, a small number of labeled examples per class yields substantial classification accuracy. At $K = 5$, IT-TIM reaches 0.827 averaged over the five partitions and 0.857 on the realm-diverse partition, using five labeled sites per class, with the supervised reference trained on 200. The Cohen's κ values are within the substantial agreement range (Landis and Koch, 1977).

Also, performance gains demonstrate significant diminishing returns beyond $K = 5$. The marginal accuracy gain per additional support example decreases by approximately 30-fold between the $K \leq 5$ and $K > 5$ regimes (from 2.3 to 0.08 percentage points per label), a trend also observed on Split

Table 3: Label efficiency of IT-TIM averaged across the five class partitions. Mean ± SD across the five partitions, after averaging the five model seeds within each partition. The supervised reference is trained on 200 labeled sites per meta-test class.

K	Accuracy	Macro-F1	Cohen's κ
1	0.734 ± 0.069	0.734 ± 0.067	0.681 ± 0.083
5	0.827 ± 0.066	0.827 ± 0.064	0.792 ± 0.079
10	0.846 ± 0.061	0.845 ± 0.060	0.815 ± 0.073
20	0.856 ± 0.058	0.856 ± 0.058	0.828 ± 0.069
50	0.862 ± 0.057	0.862 ± 0.056	0.835 ± 0.068
Supervised	0.902 ± 0.035	0.899 ± 0.037[a]	—

[a]Averaged over evaluation episodes.

Table 4: Label efficiency of IT-TIM on the realm-diverse partition (Split C). Mean ± 95% CI over five model seeds (t distribution). The supervised reference is trained on 200 labeled sites per meta-test class.

K	Accuracy	Macro-F1	Cohen's κ
1	0.790 ± 0.020	0.790 ± 0.018	0.748 ± 0.024
5	0.857 ± 0.013	0.854 ± 0.013	0.828 ± 0.015
10	0.869 ± 0.008	0.867 ± 0.007	0.843 ± 0.009
20	0.875 ± 0.008	0.874 ± 0.008	0.850 ± 0.009
50	0.880 ± 0.005	0.880 ± 0.006	0.857 ± 0.006
Supervised	0.913 ± 0.013	0.913 ± 0.012[a]	—

[a]Averaged over evaluation episodes.

C. Thus, $K = 5$ represents a practical label-efficiency operating point, capturing most accessible accuracy with only $5 \times N = 25$ labeled examples per evaluation episode.

Furthermore, the gap between IT-TIM and the supervised reference decreases but remains present even at $K = 50$ (4.0 percentage points on the cross-split mean). This residual gap aligns with the supervised reference having approximately four times as many labels per class as IT-TIM at this stage, suggesting that additional labels continue to provide gains, though at increasing cost.

6.2. Matched comparison

Label efficiency is quantified as a proportion relative to the performance of a well-labeled classifier. This comparison requires careful consideration because, under the standard protocol, the supervised reference and the few-shot methods are scored on separately drawn episodes. As a result, the resulting ratio does not represent a recovery rate for an identical task, nor is it necessarily bounded by unity. Comparisons are reported under both protocols (Table 5). In the 5-way protocol, the reference is scored on its own set of 5-way episodes, with predictions restricted to the classes of each episode. In the matched 6-way protocol, each episode utilizes the entire meta-test set, and query samples are processed by both the few-shot methods and the reference.

Table 5: Accuracy relative to the supervised reference on the realm-diverse partition, under the 5-way episodic protocol and under a matched 6-way protocol in which the reference is evaluated on the same query samples as the few-shot methods. Mean over five seeds, 600 episodes each.

Protocol	Method	$K = 1$	$K = 5$	$K = 10$	$K = 20$	$K = 50$
5-way	ProtoNet	0.759	0.848	0.862	0.867	0.871
	IT-TIM	0.790	0.857	0.869	0.875	0.880
	% of reference	86.5	93.8	95.1	95.8	96.4
6-way	ProtoNet	0.724	0.821	0.836	0.843	0.848
	IT-TIM	0.755	0.828	0.842	0.851	0.858
	% of reference	84.1	92.3	93.8	94.9	95.8
Supervised reference (6-way)		0.897	0.897	0.897	0.896	0.895

The performance of the reference classifier varies by less than 0.2 percentage points across different shot counts, as anticipated, because it does not utilize the support set. The absolute accuracy of the few-shot methods is lower under the matched protocol than under the 5-way protocol, by 2.2 to 3.5 percentage points. This reduction is most pronounced at $K = 1$, attributable to the inclusion of an additional class in the label space. The recovered proportion also declines by 0.6 to 2.4 percentage points; for instance, at $K = 5$, IT-TIM achieves 92.3% of the reference performance, compared to 93.8% under the 5-way protocol. The ranking of methods remains unchanged, with IT-TIM ranked above ProtoNet at every shot count under both protocols.

6.3. Method comparison

We compare our method with six baseline approaches at the operating point $K = 5$. The results are presented in Table 6. All methods employ the same backbone variant and adhere to identical episode protocols. We report the accuracy difference (Δ vs IT-TIM) and paired t-test p-values.

Table 6: Method comparison at $K = 5$ on Split C. Mean ± 95% CI over five model seeds (t distribution). Significance: $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$.

Method	Accuracy	Macro-F1	Cohen's κ	Δ vs IT-TIM
NC	0.791 ± 0.005	0.784 ± 0.004	0.749 ± 0.006	-6.5^{***}
DTW-1NN	0.762 ± 0.004	0.752 ± 0.002	0.714 ± 0.004	-9.4^{***}
Linear probe	0.836 ± 0.009	0.834 ± 0.008	0.803 ± 0.011	-2.1^{***}
ProtoNet	0.848 ± 0.009	0.845 ± 0.009	0.818 ± 0.011	-0.8^{*}
SimpleShot	0.850 ± 0.014	0.847 ± 0.014	0.820 ± 0.017	-0.6^{***}
LaplacianShot	0.843 ± 0.016	0.836 ± 0.018	0.812 ± 0.019	-1.3^{***}
IT-TIM (ours)	$\mathbf{0.857 \pm 0.013}$	$\mathbf{0.854 \pm 0.013}$	$\mathbf{0.828 \pm 0.015}$	—
Supervised reference	0.913 ± 0.013	0.913 ± 0.012[a]	—	—

[a]Averaged over evaluation episodes.

As shown in the table, IT-TIM achieves the highest accuracy among all evaluated methods at $K = 5$, with statistically significant improvements over all baselines (paired t-test, $p < 0.05$). The performance advantage ranges from +0.6 percentage points over SimpleShot to +9.4 percentage points over DTW-1NN.

Additionally, accuracy increases monotonically with method enrichment. The most substantial gain is observed with the introduction of a frozen meta-trained backbone combined with a linear classifier, increasing accuracy from $0.762 - 0.791$ (NC, DTW-1NN) to 0.836 (Linear probe), representing a 4.5 percentage point improvement. Replacing the linear classifier with a meta-learned classifier (ProtoNet) provides an additional +1.2 percentage points. However, this comparison alters the representation, classifier, and input dimensionality simultaneously, and for DTW, it also changes the input channels. Subsection 6.4 presents a controlled ablation that isolates the effect of the representation. Further transductive refinement yields mixed effects relative to ProtoNet, ranging from −0.5 percentage points (LaplacianShot) to +0.9 percentage points (IT-TIM), with SimpleShot at +0.2.

Furthermore, LaplacianShot demonstrates a K-dependent performance pattern that distinguishes it from the other transductive baselines. At $K = 1$,

LaplacianShot outperforms ProtoNet by +2.5 percentage points on Split C ($p = 0.001$) and +2.1 percentage points on Split A ($p = 0.021$). For $K \geq 5$, this advantage reverses: LaplacianShot underperforms ProtoNet by -0.5 percentage points at $K = 5$ and by -1.2 to -1.8 percentage points at $K = 10$. This K-dependent crossover is consistent across both Split A and Split C, suggesting that it is a property of the kNN-based regularizer rather than a split-specific artifact.

Table 7 presents a comparison across all five class partitions. Absolute accuracy varies substantially by partition, ranging from 0.711 on the water-dominated split to 0.867 on the agricultural split for IT-TIM. This variation indicates differences in spectral separability among the class subsets. The ordering remains consistent: IT-TIM achieves the highest accuracy in all five partitions, and its advantage over ProtoNet is statistically significant in four of the five cases (mean +0.83 pp, range +0.18 to +1.23 pp). The only exception is the agricultural split, where the margin is +0.18 pp ($p = 0.63$). A comparative analysis indicates that feature normalization in SimpleShot does not consistently yield an advantage on this benchmark. The accuracy of SimpleShot is lower than that of ProtoNet in two of the five partitions, and the pooled difference is not statistically significant.

Transductive refinement incurs additional computational cost. ProtoNet computes prototypes a single time and evaluates each query against them, resulting in a cost of $O(NKD + NQND)$. In contrast, IT-TIM repeats the scoring process at each of the M optimization steps, leading to a total cost of $O(NKD + M(NK + NQ)ND)$. Measured on an otherwise idle device, this amounts to 170 ms per episode at the default $M = 100$ against 13 ms for ProtoNet, a thirteenfold difference (Table 8).

Reducing the iteration count results in a smaller decrease in accuracy than in computational cost, as marginal returns diminish with increasing M. For example, $M = 10$ preserves 36% of the improvement over ProtoNet while adding only 17 ms to ProtoNet's 13 ms, against 157 ms added at the default setting; $M = 50$ preserves 71% for 85 ms added. Early iterations therefore contribute most substantially. Nevertheless, since the total gain on this partition is only 0.84 percentage points, the absolute difference between operating points remains well below one percentage point, so the iteration count is best regarded as an explicit trade-off parameter rather than a setting with a single optimal value.

Table 7: Method comparison at $K = 5$ across all five class partitions. Mean accuracy over five model seeds, 600 episodes each. Significance marks on the IT-TIM row denote a paired t-test against ProtoNet, matched on model seed ($n = 5$): $^{*}p < 0.05$, $^{**}p < 0.01$.

	Class partition					
Method	A	B	C	D	E	Mean
Nearest centroid	0.790	0.822	0.791	0.824	0.640	0.773
DTW-1NN	0.758	0.744	0.762	0.757	0.588	0.722
Linear probe	0.848	0.843	0.836	0.812	0.698	0.807
ProtoNet	0.853	0.865	0.848	0.823	0.704	0.819
SimpleShot	0.855	0.854	0.850	0.825	0.699	0.817
LaplacianShot	0.847	0.833	0.843	0.795	0.646	0.793
IT-TIM	**0.865****	**0.867**	**0.857***	**0.835****	**0.711***	**0.827**
Supervised reference	0.924	0.934	0.913	0.886	0.850	0.902

Table 8: Inference cost per episode (Split C, $N = 5$, $K = 5$, 100 episodes after 10 warm-up episodes, measured sequentially on an otherwise idle device: macOS arm64, PyTorch 2.11.0, MPS backend, 8 threads). Accuracy is taken from the standard protocol of Table 6 (600 episodes, five seeds). Recovered fractions quoted in the text are computed from unrounded accuracies.

Method	ms/episode	SD	× ProtoNet	Accuracy
Nearest centroid	0.45	0.07	0.03	0.791
ProtoNet	13.07	0.57	1.00	0.848
SimpleShot	13.10	0.61	1.00	0.850
LaplacianShot	16.02	1.26	1.23	0.843
IT-TIM ($M = 10$)	29.85	3.13	2.28	0.851
IT-TIM ($M = 25$)	52.96	2.32	4.05	0.852
IT-TIM ($M = 50$)	97.57	5.95	7.46	0.854
IT-TIM ($M = 100$, default)	170.24	6.64	13.02	0.857
Linear probe	133.88	6.04	10.24	0.836
DTW-1NN	1789.13	15.43	136.88	0.762

6.4. Representation and inference rule

The comparison in Subsection 6.3 involves simultaneous modifications to the representation, classifier, input dimensionality, and, for DTW, the input channels. To isolate the effect of the representation, the inference rule,

classifier, input dimensionality, and input channels are held constant, and only the encoder weights are varied. Regarding this, four encoders are evaluated. The first encoder is a randomly initialized InceptionTime model that remains untrained. The second encoder is an InceptionTime model trained conventionally with cross-entropy on the eighteen meta-training classes for 50 epochs using Adam (learning rate 10^{-3}, weight decay 10^{-4}, batch size 128); the classifier head is discarded and the 128-dimensional pooled embedding is retained. The number of epochs is determined from a validation-accuracy curve on a single split-seed combination, where accuracy plateaus between epochs 30 and 40. This training schedule is applied uniformly without early stopping to ensure all encoders share identical training conditions. The third encoder is the episodically meta-trained InceptionTime model used throughout this study. The fourth encoder is a TempCNN (Pelletier et al., 2019), which consists of three one-dimensional convolutional layers with kernel sizes 5 and 64 filters each, followed by batch normalization, ReLU, and dropout, and a fully connected layer producing a 128-dimensional embedding. This model is meta-trained under the same episodic protocol, with an adaptation to use global average pooling over the temporal axis instead of the original flattening. This adjustment isolates the convolutional architecture from the temporal aggregation scheme. Results are presented in Table 9.

The encoder axis is the primary determinant of performance. When ProtoNet inference is held constant, replacing random initialization with episodic meta-training results in an improvement of 8.56 percentage points at $K = 5$ ($p = 0.007$, $d_z = 2.32$). In contrast, substituting inductive with transductive inference under the same encoder yields an increase of 0.83 percentage points ($p = 0.012$, $d_z = 1.97$). This pattern is also observed with the TempCNN encoder: the encoder axis contributes 5.80 percentage points, while the inference axis contributes 0.30 percentage points. Although InceptionTime is numerically ahead of TempCNN, the architectural difference (2.76 percentage points at $K = 5$, not significant at the partition level) is only one-third the magnitude of the encoder-axis effect. Thus, the relative importance of these axes is consistent across different architectures.

Most of the performance improvement is attributable to the use of a learned representation rather than the episodic objective itself. The conventionally supervised encoder recovers 81.8% of the gap between random initialization and episodic meta-training at $K = 5$, but only 62.2% at $K = 1$. Therefore, the episodic objective confers a disproportionately greater benefit when the support set is smallest, although the episodic-minus-supervised

Table 9: Controlled representation ablation. The inference rule, classifier, input dimensionality, and input channels are held fixed; only the encoder weights differ. Upper block: mean accuracy over the 25 split–seed runs. Supervised (CE) denotes conventional cross-entropy training on the meta-training classes. The seen-class column reports 5-way 5-shot ProtoNet accuracy on episodes drawn from the meta-training pool, computed under an identical protocol for all four encoders. Lower block: differences paired over the five class partitions after averaging the five model seeds within each partition ($n = 5$, two-sided paired t-test).

		$K = 1$		$K = 5$	
Encoder	Seen-class	ProtoNet	IT-TIM	ProtoNet	IT-TIM
Random initialization	0.747	0.641	0.635	0.733	0.726
Supervised (CE)	0.920	0.684	0.713	0.803	0.812
Episodic (InceptionTime)	0.911	0.710	**0.734**	0.819	**0.827**
Episodic (TempCNN)	0.863	0.693	0.703	0.791	0.794

	$K = 1$		$K = 5$	
Contrast	Δ (pp)	p	Δ (pp)	p
Encoder axis (ProtoNet inference held fixed)				
Supervised (CE) − random	+4.30	0.054	+7.00	0.011
Episodic − supervised (CE)	+2.62	0.042	+1.56	0.078
Episodic − random	+6.92	0.009	+8.56	0.007
Episodic TempCNN − random	+5.20	0.017	+5.80	0.007
InceptionTime − TempCNN (architecture only)	+1.72	0.237	+2.76	0.136
Inference axis (encoder held fixed)				
IT-TIM − ProtoNet, episodic InceptionTime	+2.41	0.007	+0.83	0.012
IT-TIM − ProtoNet, episodic TempCNN	+1.00	0.130	+0.30	0.327
IT-TIM − ProtoNet, random initialization	−0.59	0.321	−0.71	0.417

contrast is significant only at $K = 1$ ($p = 0.042$; $p = 0.078$ at $K = 5$).

Analysis of the seen classes clarifies the nature of this contribution. When evaluated under an identical protocol on episodes drawn from the meta-training pool, the supervised encoder separates the seen classes slightly better than the episodic encoder (0.920 compared to 0.911). However, for held-out classes, this ordering reverses at both shot counts, and the supervised representation exhibits a larger performance drop from seen to novel classes (11.7 and 9.2 percentage points, respectively). Therefore, episodic meta-training does not produce a universally superior representation, but rather one that transfers more effectively to novel classes, a property not observable

in evaluations limited to the training classes.

The crossed experimental design also reveals an interaction effect. With a randomly initialized encoder, IT-TIM does not improve on ProtoNet (-0.59 percentage points at $K = 1$ and -0.71 percentage points at $K = 5$, neither significant), in contrast to the ordering observed for all three learned representations. This result is consistent with the mechanism described in Subsection 7.4: the marginal-entropy term redirects the prototypes toward a configuration that balances the predicted class distribution, which is beneficial only when the embedding effectively separates the classes. In the absence of this separation, the redirection operates on noise. The relationship is characterized by a threshold effect rather than a continuous one: the transductive gain is absent for the untrained encoder and positive for all three learned encoders, but among learned encoders, its magnitude does not correlate with representation quality (Spearman $\rho = -0.50$ at $K = 1$ and $+0.50$ at $K = 5$, both $p > 0.6$).

Another important consideration pertains to the input features rather than the encoder. For each site, three static features are available: elevation and the two projected coordinates. The impact of including these features varies by more than an order of magnitude across partitions, reaching 7.68 percentage points on one of them. However, these features are excluded from all experiments reported in this study, because several classification criteria are defined by elevation and location. Consequently, a model provided with these inputs could infer part of the label from the assignment rule instead of the observed time series. Appendix B presents the comparison and supporting evidence for this interpretation.

7. Robustness and Mechanism

Following the demonstration that IT-TIM achieves the highest accuracy at the $K = 5$ operating point, this section investigates the robustness and underlying factors contributing to this performance. The analysis encompasses varying the number of classes per episode (Subsection 7.1), quantifying spatial autocorrelation in the benchmark and re-evaluating under spatial controls (Subsection 7.2), decomposing accuracy by individual ecosystem classes (Subsection 7.3), decomposing the transductive objective to identify which of its terms drives the gain (Subsection 7.4), establishing the condition under which that gain holds (Subsection 7.5), and evaluating a partition in which six of the seven freshwater classes are withheld from meta-training (Subsec-

tion 7.6). Unless stated otherwise, all analyses employ the same frozen Split C backbone and $K = 5$ protocol as described in Section 6.

7.1. Scaling to more classes

The number of classes sampled per episode is varied, with $N \in \{2, 3, 5, 6\}$, selected from the six-class meta-test pool. Each episode contains $K = 5$ support and $Q = 15$ query examples per class, evaluated over 600 episodes per seed and 5 model seeds. Table 10 reports results on the held-out meta-test classes across all three metrics. With the number of queries per class fixed at 15, increasing N in proportion raises the total number of queries from 30 to 90. As a result, this analysis does not fully disentangle the effects of label-space size from the quantity of unlabeled query information available for transductive refinement.

Table 10: N-way scaling on meta-test classes of Split C at $K = 5$. Mean ± 95% CI over five model seeds (t distribution).

N	Method	Accuracy	Macro-F1	Cohen's κ
2	ProtoNet	0.951 ± 0.004	0.951 ± 0.003	0.941 ± 0.005
	SimpleShot	0.955 ± 0.007	0.954 ± 0.006	0.945 ± 0.008
	LaplacianShot	0.952 ± 0.008	0.952 ± 0.007	0.942 ± 0.010
	IT-TIM	$\mathbf{0.958 \pm 0.006}$	$\mathbf{0.958 \pm 0.006}$	$\mathbf{0.949 \pm 0.008}$
3	ProtoNet	0.911 ± 0.007	0.909 ± 0.007	0.893 ± 0.009
	SimpleShot	0.915 ± 0.007	0.914 ± 0.008	0.898 ± 0.009
	LaplacianShot	0.912 ± 0.009	0.910 ± 0.010	0.895 ± 0.010
	IT-TIM	$\mathbf{0.921 \pm 0.006}$	$\mathbf{0.920 \pm 0.007}$	$\mathbf{0.905 \pm 0.007}$
5	ProtoNet	0.848 ± 0.009	0.845 ± 0.009	0.818 ± 0.011
	SimpleShot	0.850 ± 0.014	0.847 ± 0.014	0.820 ± 0.017
	LaplacianShot	0.843 ± 0.016	0.836 ± 0.018	0.812 ± 0.019
	IT-TIM	$\mathbf{0.857 \pm 0.013}$	$\mathbf{0.854 \pm 0.013}$	$\mathbf{0.828 \pm 0.015}$
6	ProtoNet	0.820 ± 0.008	0.817 ± 0.008	0.784 ± 0.009
	SimpleShot	0.820 ± 0.010	0.816 ± 0.011	0.784 ± 0.012
	LaplacianShot	0.810 ± 0.013	0.799 ± 0.019	0.772 ± 0.016
	IT-TIM	$\mathbf{0.827 \pm 0.010}$	$\mathbf{0.824 \pm 0.010}$	$\mathbf{0.792 \pm 0.012}$

In this context, IT-TIM achieves the highest accuracy, macro-F1, and Cohen's κ across all values of N. The ranking of methods remains stable

as the label space expands; no baseline method outperforms IT-TIM at any evaluated point. The accuracy margin over the strongest baseline at each N remains consistent, ranging from +0.3 to +0.7 percentage points, and is statistically significant in 11 out of 12 IT-TIM-versus-baseline comparisons (paired t-test, $p < 0.05$; the only exception is IT-TIM versus ProtoNet at $N = 6$, $p = 0.056$). This single non-significant comparison occurs at the most challenging setting ($N = 6$), where the margin between all methods is minimal. All methods demonstrate reduced absolute performance as N increases (IT-TIM accuracy 0.958 to 0.827), indicating the increased difficulty associated with larger label spaces; however, IT-TIM maintains its relative ranking. Furthermore, LaplacianShot's performance declines relative to other baselines as N increases: it is comparable to ProtoNet at $N \in \{2, 3\}$, but falls below ProtoNet at $N = 6$ (-1.0 percentage points, $p = 0.044$). This trend is analyzed in greater detail in Subsection 7.3, which identifies a shared underlying cause with LaplacianShot's behavior across shot counts and individual classes.

As a control, the analysis was repeated using the meta-train (seen) class pool. In this context, no statistically significant difference was detected between IT-TIM and ProtoNet at all values of N ($p > 0.05$), indicating that IT-TIM's advantage is specific to novel classes. When evaluation classes are encountered during meta-training, the backbone representation is already well-separated, and transductive refinement provides no additional benefit. The intended application of the method is the classification of genuinely unseen ecosystem types.

7.2. Spatial autocorrelation and leakage

The sample sites exhibit high spatial density, with each observation incorporating a 3×3 pixel neighborhood. As a result, sites in close proximity may share pixels, local environmental conditions, or originate from the same source polygon. Assigning these sites to opposite sides of the support/query boundary may artificially simplify the few-shot task relative to prediction at entirely new locations. To address this, the extent of clustering is quantified, and the novel-class protocol is re-evaluated using two spatial controls.

Of the 24,009 sites, 17.0% have a same-class neighbor within 90 m, the distance at which neighborhood windows of two sites physically overlap, and 42.2% within 500 m. The degree of clustering varies by more than two orders of magnitude across classes, with median nearest-neighbor distances ranging from 30 m for intensive pasture to $7,564$ m for temperate deciduous forest.

This variation reflects the mapped extent of each class: classes covering large contiguous areas are sparsely sampled, while those with limited mapped extent are sampled densely within a small number of polygons. Among the six meta-test classes in the realm-diverse partition, two are severely clustered (99.9% and 86.4% of sites within 90 m of a same-class neighbor), and two are effectively independent (0.2% and 2.6%). Per-class distributions are provided in Appendix C. At the extreme, three pairs of sites share identical coordinates, all within a single class.

The novel-class evaluation is repeated under two spatial controls (Table 11). The first control enforces a minimum distance between support and query sites of the same class, set at 90, 200, 500, and $1,000$ m. Episode sampling is successful in all conditions, enabling the use of all 600 episodes without selection. The second control is more stringent: sites are grouped into spatially disjoint blocks using single-linkage clustering, and no query site may belong to a block occupied by a support site. This method prevents sites from the same contiguous polygon from appearing on opposite sides of the boundary. Accuracy declines under both controls, but only by a small margin. At $K = 5$, the decrease is 0.72 percentage points for ProtoNet and 0.66 percentage points for IT-TIM under a 1 km buffer, and 1.24 and 1.18 percentage points under blocking at the same threshold. Halving the linkage threshold to 500 m changes accuracy by less than 0.2 pp, within the confidence interval, consistent with the block structure of these classes being robust to the threshold. An additional control, in which support and query sets are drawn from different geographic quadrants of the study area, produces similar results for the meta-training classes, with accuracy ranging from 0.897 to 0.910 for all four methods (Appendix D).

Per-class analysis demonstrates that the aggregate figure conceals a highly uneven distribution, and that clustering does not predict susceptibility as might be expected. Of the 0.72 percentage points lost by ProtoNet at 1 km, nearly all is attributable to a single class: marshes lose 3.9 percentage points, while the four spatially dispersed classes lose only 0.07 percentage points in total. Intensive pasture, the most densely clustered class in the dataset, with every site within 95 m of another, loses only 0.2 percentage points. This result arises because its 1,000 sites are distributed across 31 spatially disjoint polygons, so excluding a radius around each support site removes only 2.6% of the candidate pool, whereas a single block accounts for 52% of the sites in marshes. Therefore, nearest-neighbor distance is a poor indicator of susceptibility to spatial leakage; the critical factors are the number and

Table 11: Novel-class evaluation on the realm-diverse partition under spatial controls. The buffer enforces a minimum support–query distance within each class; the block constraint additionally prevents support and query sites from sharing a spatially contiguous group. Mean accuracy over five seeds, 600 episodes each.

		$K = 1$		$K = 5$	
Control	Threshold	ProtoNet	IT-TIM	ProtoNet	IT-TIM
None	—	0.759	0.790	0.848	0.857
Buffer	90 m	0.759	0.790	0.848	0.856
	200 m	0.759	0.790	0.847	0.856
	500 m	0.758	0.789	0.845	0.853
	1,000 m	0.756	0.787	0.841	0.850
Block	500 m	0.752	0.785	0.835	0.845
	1,000 m	0.754	0.786	0.836	0.845

relative size of the spatially disjoint groups within a class.

A central question is whether the advantage of IT-TIM over ProtoNet persists under these spatial controls. The advantage remains consistent and does not diminish. At $K = 5$, the difference is +0.84 percentage points under random sampling, +0.84 percentage points with a 90 m buffer, +0.90 percentage points with a 1 km buffer, and +0.90 percentage points under blocking, remaining significant in all cases. At $K = 1$, the difference increases from +3.08 to +3.26 percentage points under blocking at 1 km, and from +3.16 to +3.41 percentage points for the two clustered classes. These results indicate that spatial proximity between support and query sites does not account for the observed transductive gain.

7.3. Per-class behavior

To identify the source of IT-TIM's aggregate advantage, per-class recall was investigated for the six meta-test classes of Split C at $K = 5$. Table 12 shows that the relative performance of the methods varies substantially depending on the specific class.

IT-TIM's transductive refinement yields the most substantial improvements for the most challenging classes. The greatest gain over ProtoNet is observed for the Marine open ocean (class 11, +4.9 pp), which represents the class with the second-lowest recall. Conversely, for classes that are already straightforward to classify (subalpine forest, pasture, marshes, all

Table 12: Per-class recall on the six Split C meta-test classes at $K = 5$. Mean ± 95% CI over five model seeds (t distribution).

Class	Description	ProtoNet	SimpleShot	LaplacianShot	IT-TIM
0	Subalpine coniferous	0.978 ± 0.027	0.988 ± 0.009	0.992 ± 0.005	0.989 ± 0.008
7	Intensive pasture	0.969 ± 0.016	0.960 ± 0.015	0.965 ± 0.013	0.970 ± 0.010
11	Marine open ocean	0.686 ± 0.014	0.739 ± 0.022	0.766 ± 0.018	0.734 ± 0.021
12	Permanent upland streams	0.908 ± 0.020	0.900 ± 0.012	0.924 ± 0.006	0.899 ± 0.014
19	Coastal shorelines (aggregated)	0.623 ± 0.023	0.583 ± 0.055	0.479 ± 0.094	0.612 ± 0.047
22	Marshes	0.921 ± 0.028	0.929 ± 0.022	0.931 ± 0.018	0.932 ± 0.014

above 0.92), the improvement remains minimal (+0.1 to +1.1 pp). These results suggest that the refinement primarily adjusts prototypes where the support-mean estimate is least reliable, rather than applying changes uniformly across all classes. One class deviates from this pattern: for coastal shorelines (class 19), which exhibits the lowest recall, IT-TIM performs 1.1 percentage points below ProtoNet. This class is analyzed separately below, where its aggregated composition, rather than the inference rule, is identified as the primary source of difficulty.

The behavior of LaplacianShot across multiple analyses indicates a single underlying mechanism. LaplacianShot achieves competitive performance at $K = 1$, but its effectiveness diminishes as the shot count increases. Its performance also declines relative to ProtoNet as N increases (Subsection 7.1), and it performs worst on the spectrally heterogeneous coastal-shoreline class (recall 0.479, substantially lower than all other methods). These outcomes appear to result from a common factor: the Laplacian regularizer propagates labels along a k-nearest-neighbor graph over query embeddings, which is effective only when each class forms a compact and well-populated region in the embedding space. Low support counts, a large number of classes competing for the same query neighborhood, and dispersed class embeddings each undermine this condition, leading to consistent performance degradation. This pattern persists across shot, class-count, and per-class analyses, rather than arising from any single experimental setting.

Coastal shorelines (class 19) represent the lowest-recall class for all evaluated methods (recall 0.612 even for IT-TIM). This class encompasses three IUCN GET functional groups (MT1.1, MT1.3, MT1.4) that the national reference products do not delineate separately. Figure 3 presents the row-normalized confusion matrix for IT-TIM on the six meta-test classes

Approximately 28% of class-19 queries are predicted as Marine open ocean

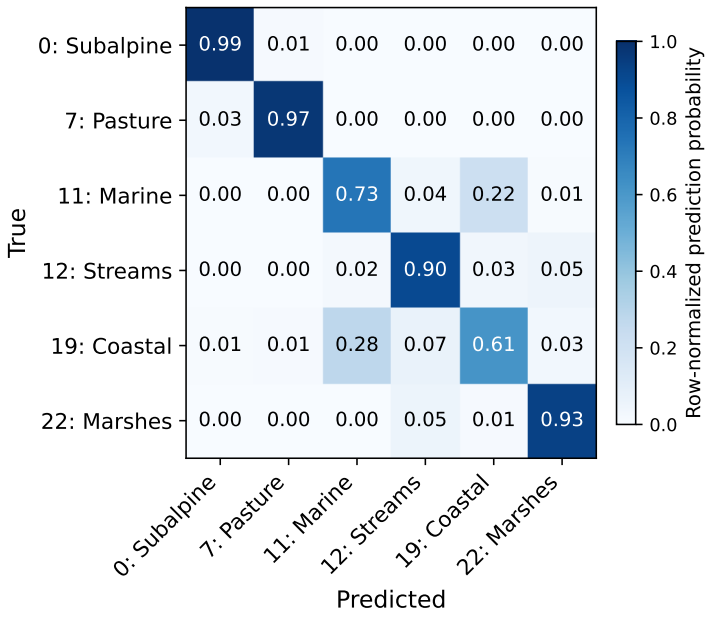


Figure 3: Row-normalized confusion matrix for IT-TIM on the realm-diverse split (Split C) at $K = 5$, averaged over 5 model seeds and 600 episodes per seed.

(class 11), with a reciprocal confusion in which approximately 22% of class-11 queries are assigned to class 19. In contrast, the remaining four classes (Subalpine, Pasture, Streams, Marshes) each achieve recall rates of 0.90 or higher. This confusion pattern is concentrated almost entirely along a single off-diagonal pair, indicating a specific failure mode rather than a widespread degradation across classes.

This class spans spectrally diverse subtypes, such as open-coast, estuarine, and intertidal shorelines, which produce embeddings that cover a broad, water-dominated region and overlap with the open-ocean class. In contrast, the one marine–terrestrial class the source layers do resolve individually, tidal flats (class 20), achieves a recall above 0.9 in the splits where it is present. This comparison supports a testable hypothesis: the low recall of class 19 is attributable to the aggregation of subtypes in the label set, rather than to an inherent limitation of the method. Treating the constituent groups separately would likely yield accuracy comparable to other marine–terrestrial classes. This result highlights an important consideration for practitioners assembling label sets under the IUCN GET, as functional groups that cannot

be resolved individually impose a measurable classification cost.

7.4. Decomposing the transductive objective

The IT-TIM objective (Equation 6) integrates a support cross-entropy term with two query-side entropy terms. Accurate attribution of observed performance gains to specific terms is essential, since IT-TIM differs from ProtoNet not only in its objective but also in its use of $M = 100$ prototype optimization iterations and operation on ℓ_2-normalized embeddings. Removing one entropy term and comparing the result to ProtoNet would confound the effect of that term with the influence of prototype refitting on the labeled support set.

Table 13 isolates these factors. Five experimental conditions utilize an identical optimization procedure and differ only in the active objective terms: support cross-entropy alone (B), with the marginal-entropy term (C), with the conditional-entropy term (D), and the full objective (E). Two non-optimizing references, ProtoNet (A) and SimpleShot (A′), serve as baselines for these conditions. SimpleShot corresponds to initializing IT-TIM without any optimization steps. The same episode stream is maintained throughout, ensuring that each comparison is paired at the split and seed level.

Both potentially confounding factors are negligible. Feature normalization has no measurable effect in either shot regime, and its direction varies across partitions. Prototype refitting on the support set exerts minimal influence and changes sign with K, resulting in a loss of 0.18 percentage points at $K = 1$ due to overfitting with a single labeled example, and a gain of 0.43 percentage points at $K = 5$. Combined, these factors account for -0.14 and $+0.24$ percentage points, compared to the total IT-TIM advantage over ProtoNet of $+2.41$ and $+0.83$ percentage points. Therefore, the query-side terms explain nearly the entire observed advantage. Among these, the marginal-entropy term is responsible for almost all of the gain, demonstrating positive and significant effects in all five partitions at both shot counts. The conditional-entropy term contributes little independently and nothing at $K = 5$ when the marginal term is present. These findings confirm, under controlled comparison, that class-balance enforcement, rather than per-query confidence sharpening, is the primary source of the transductive gain in this task.

Table 14 provides the effect of the conditional-entropy weight λ_2.

The results indicate that IT-TIM is substantially robust to the choice of λ_2. Across the entire $20\times$ range $\lambda_2 \in [0.05, 1.0]$, accuracy varies by no more

Table 13: Controlled decomposition of the IT-TIM inference objective. All conditions except ProtoNet and SimpleShot apply the identical prototype optimization procedure and differ only in which objective terms are active. Upper block: mean accuracy over the 25 split–seed runs (five class splits × five model seeds, 600 episodes each). Lower block: differences paired over the five class partitions after averaging seeds within each partition ($n = 5$, two-sided paired t-test).

		Active terms		Accuracy	
Condition	Optimization	$\mathcal{H}_{\text{marg}}$	$\mathcal{H}_{\text{cond}}$	$K = 1$	$K = 5$
ProtoNet (A)	none	—	—	0.710	0.819
SimpleShot (A′)	none	—	—	0.711	0.817
Support CE only (B)	✓	—	—	0.709	0.821
+ marginal (C)	✓	✓	—	0.733	**0.827**
+ conditional (D)	✓	—	✓	0.712	0.821
IT-TIM, full (E)	✓	✓	✓	**0.734**	**0.827**

		$K = 1$		$K = 5$	
Contrast		Δ (pp)	p	Δ (pp)	p
Inductive components					
A′ − A	feature normalization	+0.04	0.950	−0.19	0.503
B − A′	prototype refitting	−0.18	0.020	+0.43	0.007
Transductive components					
C − B	marginal entropy	+2.37	< 0.001	+0.61	< 0.001
D − B	conditional entropy	+0.28	0.032	+0.03	0.255
E − C	conditional given marginal	+0.17	0.016	−0.01	0.643
E − B	all transductive terms	+2.55	< 0.001	+0.60	< 0.001
E − A	IT-TIM vs. ProtoNet	+2.41	0.007	+0.83	0.012

than 0.27 percentage points at $K = 1$ and 0.04 percentage points at $K = 5$. Several of these differences reach statistical significance at $K = 1$ because they are consistent across seeds, but all are negligible in magnitude. Consequently, the default $\lambda_2 = 0.1$ does not represent a tuned hyperparameter.

The decomposition indicates that the marginal-entropy term accounts for the observed performance gain, yet it does not elucidate the underlying mechanism. While the stated purpose of this term is to prevent degenerate solutions in which all queries are assigned to a single class (Boudiaf

Table 14: Effect of the conditional-entropy weight λ_2 on IT-TIM accuracy (Split C, mean $\pm$ 95% CI over five model seeds (t distribution)).

	$K = 1$		$K = 5$	
λ_2	Accuracy	p vs λ_2=0	Accuracy	p vs λ_2=0
0 (no cond.)	0.788 ± 0.019	—	0.857 ± 0.013	—
0.05	0.789 ± 0.019	< 0.001	0.857 ± 0.013	0.542
0.1 (default)	0.790 ± 0.020	0.001	0.857 ± 0.013	0.901
0.2	0.791 ± 0.020	0.003	0.857 ± 0.013	0.933
0.5	0.791 ± 0.020	0.009	0.857 ± 0.013	0.889
1.0	0.788 ± 0.021	0.842	0.856 ± 0.013	0.246
ProtoNet (ref.)	0.759 ± 0.024	< 0.001	0.848 ± 0.009	0.024

et al., 2020), such an effect is not observed in the current context. In all 15,000 episodes evaluated without the marginal-entropy term, predictions were never restricted to fewer than four of the five episode classes. Additionally, the marginal-entropy term does not simply increase prototype displacement. Because predictions depend exclusively on prototype direction, displacement is quantified as the angle between $\boldsymbol{\mu}^{(M)}$ and $\boldsymbol{\mu}^{(0)}$. All experimental conditions yield prototype rotations between 5.3° and 5.8°. Notably, the condition producing the largest rotation, the conditional-entropy term, results in lower accuracy than the marginal-entropy term.

The primary distinction of the marginal-entropy term lies in the direction of the resulting prototype displacement. Letting the displacement under condition X be Δ_X, the cosine similarity between Δ_C and Δ_B is 0.344 at $K = 1$ and 0.865 at $K = 5$. Consequently, at $K = 1$, the two displacements differ by more than the magnitude of either displacement. This redirection addresses a quantifiable objective: balancing the predicted class distribution, measured by $\mathrm{KL}(\hat{p}\|u) = \log N - H(\hat{p})$. Because the query set is balanced by design, this metric isolates prediction imbalance. The marginal-entropy term reduces this divergence by 51% at $K = 1$ and 31% at $K = 5$, whereas prototype refitting alone has no effect. The adjustment is minor in absolute terms, as the final prototypes under conditions B and C exhibit a cosine similarity of 0.994. It nonetheless modifies 5.5% of query predictions at $K = 1$, with 62% of these changes correcting previous errors. Thus, the marginal-entropy term operates on the predicted marginal as intended, functioning as a correction for a mildly skewed distribution rather than as a safeguard

against collapse.

7.5. Operating condition of the transductive gain

The previously established mechanism provides a direct and testable implication. The marginal-entropy term improves accuracy by aligning query predictions with a uniform class prior; consequently, any benefit should be restricted to query sets that are approximately class-balanced. Although the standard episodic protocol assumes this condition, real-world deployment does not ensure it. The Split C, $K = 5$ evaluation is repeated, substituting the balanced query set with a Dirichlet(α)-sampled per-class allocation at a fixed total of $Q = 75$, for $\alpha \in \{\infty, 5, 1, 0.5\}$. The weight λ_1 on the marginal-entropy term is varied in conjunction with three inference rules that do not utilize the query set.

The results differentiate the inference rules exactly as predicted by the objective decomposition (Table 15). The three query-independent conditions remain unaffected by the query distribution, exhibiting changes of no more than 0.09 percentage points between the balanced and most imbalanced regimes, none of which are statistically significant; the same holds for IT-TIM with the marginal term disabled ($\lambda_1 = 0$, +0.32 pp, not significant). Sensitivity appears only when the marginal-entropy term is active, increasing monotonically with its weight: -0.92 percentage points at $\lambda_1 = 0.25$, -1.63 at 0.5, and -2.45 at the default $\lambda_1 = 1$, all significant. For $K = 1$, the effect is approximately twice as large, reaching -4.93 percentage points ($d_z = -10.26$; Appendix E). In the most imbalanced regime, the default configuration performs worse than ProtoNet at both shot counts, by 1.62 percentage points at $K = 5$ and 1.71 at $K = 1$, across all seeds. Thus, the term responsible for the transductive gain is also the primary source of its fragility.

This analysis indicates a direct solution. Maximizing $\mathcal{H}_{\text{marg}}$ is equivalent to minimizing $\text{KL}(\bar{\mathbf{p}}(\boldsymbol{\mu}) \| \mathbf{u}) = \log N - \mathcal{H}_{\text{marg}}(\boldsymbol{\mu})$, where $\mathbf{u}$ is the uniform distribution over the N episode classes, so Equation 6 implicitly assumes a uniform class prior. The estimated-prior variant replaces $\mathbf{u}$ with an estimate $\hat{\boldsymbol{\pi}}$:

$$\mathcal{L}_{\hat{\pi}}(\boldsymbol{\mu}) = \alpha \cdot \mathcal{L}_{\text{CE}}(\boldsymbol{\mu}) + \lambda_1 \cdot \sum_{c=1}^{N} \bar{p}_c(\boldsymbol{\mu}) \log \frac{\bar{p}_c(\boldsymbol{\mu})}{\hat{\pi}_c} + \lambda_2 \cdot \mathcal{H}_{\text{cond}}(\boldsymbol{\mu}), \tag{12}$$

which reduces to Equation 6, up to an additive constant, when $\hat{\boldsymbol{\pi}} = \mathbf{u}$. The support set cannot be used to estimate the prior, as it contains exactly K ex-

Table 15: Accuracy under Dirichlet(α)-sampled imbalanced query distributions (Split C, $K = 5$, mean over five seeds). $\alpha = \infty$ recovers the standard balanced protocol and reproduces the values of Table 6. The upper block comprises inference rules that do not use the query set; the lower block varies the weight λ_1 on the marginal-entropy term. The final row replaces the uniform prior with one estimated from an initial inductive prediction. Δ is the change from $\alpha = \infty$ to $\alpha = 0.5$. Paired t-test (matched on seed): $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$.

Method	$\alpha = \infty$	$\alpha = 5$	$\alpha = 1$	$\alpha = 0.5$	Δ (pp)
Query-independent					
ProtoNet	0.848	0.848	0.849	0.848	+0.00
SimpleShot	0.850	0.851	0.850	0.851	+0.07
Support CE only	0.852	0.853	0.851	0.853	+0.09
IT-TIM, varying λ_1					
$\lambda_1 = 0$	0.852	0.854	0.853	0.856	+0.32
$\lambda_1 = 0.25$	0.854	0.853	0.848	0.845	-0.92^{**}
$\lambda_1 = 0.5$	0.855	0.853	0.844	0.839	-1.63^{***}
$\lambda_1 = 1$ (default)	**0.857**	0.852	0.839	0.832	-2.45^{***}
$\lambda_1 = 1$, estimated prior	0.852	0.853	0.854	**0.857**	$+0.51^{*}$

amples per class by design. Instead, $\hat{\boldsymbol{\pi}}$ is computed once, before optimization, from the inductive ProtoNet prediction on the query set:

$$\hat{\pi}_c = \frac{n_c + 1}{|\mathcal{Q}| + N}, \qquad n_c = \left|\{\mathbf{x}_j^q \in \mathcal{Q} : \arg\max_{c'} p(c' \mid \mathbf{x}_j^q; \boldsymbol{\mu}^{(0)}) = c\}\right|, \tag{13}$$

where the add-one term prevents zero probabilities. $\hat{\boldsymbol{\pi}}$ is held fixed during the M iterations, and all other settings are unchanged. This substitution eliminates the observed sensitivity.

At $\alpha = 0.5$, accuracy increases from 0.832 to 0.857 (+2.50 percentage points, $p < 0.0001$), which also exceeds ProtoNet by 0.88 percentage points. Consequently, the transductive advantage is maintained under imbalance rather than simply neutralized. The estimated prior incurs a cost of 0.46 percentage points when the query set is balanced and the uniform assumption holds, and 1.99 percentage points at $K = 1$. The choice between the two priors depends on whether the deployment distribution is known to be approximately uniform. Furthermore, reducing λ_1 offers additional flexibility for practitioners who anticipate imbalance but cannot estimate its extent.

7.6. Taxonomy-aware evaluation

The previously described random class partitions may underestimate the difficulty presented by truly novel functional groups, since ecologically similar classes are often present in both the meta-training and meta-test pools. To more rigorously assess generalization, a partition was created that withholds most of one ecological realm: six of the seven freshwater classes (F1.1 to F1.5 and the lake group) are assigned exclusively to the meta-test pool, and the remaining eighteen classes, including the seventh freshwater class (rice paddies, F3.3), are retained for meta-training. The backbone models were meta-trained following the standard protocol using five random seeds. Table 16 presents the results.

Table 16: Accuracy under a taxonomy-aware partition in which six of the seven freshwater classes are held out (mean over five seeds).

Method	$K = 1$	$K = 5$
ProtoNet	0.315	0.390
SimpleShot	0.317	0.392
LaplacianShot	0.306	0.381
IT-TIM	**0.324**	**0.397**

A marked reduction in accuracy is observed. IT-TIM achieves 0.397 at $K = 5$, compared to 0.827 under random partitioning, and 0.324 at $K = 1$. The held-out classes differ primarily in stream order and seasonality. Without comparable freshwater classes in the meta-training pool, the learned representations demonstrate limited capacity to distinguish among these classes. Meta-training performance remains stable, with episode-level validation accuracy within the expected range. These findings suggest that the decline in accuracy results from increased transfer difficulty, rather than deficiencies in the training process.

Although overall accuracy declines, the ordering among methods is preserved. IT-TIM remains ahead of ProtoNet by 0.89 percentage points at $K = 1$ ($p = 0.014$, $d_z = 1.85$) and by 0.71 at $K = 5$, though the latter does not reach significance ($p = 0.068$, $d_z = 1.11$); SimpleShot and LaplacianShot retain their respective positions. These results indicate a limitation in the generalizability of the primary findings: few-shot transfer within a typology depends on the inclusion of functionally related groups in the meta-training pool.

8. Discussion

8.1. Principal findings

This study presents the systematic evaluation of transductive few-shot learning for ecosystem functional classification using satellite time series, leveraging a newly developed benchmark of around 24,000 sites spanning 24 IUCN GET functional groups and six realms, labeled under a common classification framework applied to national thematic products. With five labeled sites per class, IT-TIM reaches approximately 92% of the accuracy of a supervised reference that uses forty times more labeled examples per target class, under a matched protocol on the realm-diverse partition. With a learned representation and balanced query sets, IT-TIM ranks first across all evaluated shot counts, class counts, and class partitions, and this ranking is preserved under spatial controls and when six of the seven freshwater classes are withheld.

The margin over the strongest baseline is, however, small, below 1.3 percentage points at $K = 5$ in every partition, and it comes at about thirteen times the per-episode inference cost of ProtoNet. Three of the IT-TIM versus ProtoNet comparisons are not statistically significant: the agricultural partition ($p = 0.63$), the full 6-way setting ($p = 0.056$), and the taxonomy-aware partition at $K = 5$ ($p = 0.068$). The ranking does not hold without a learned representation or under strongly imbalanced queries (Subsection 6.4 and Subsection 7.5).

The transductive gain observed for IT-TIM is primarily due to the marginal-entropy term, rather than the conditional-entropy term. A controlled decomposition, which holds the prototype optimization procedure fixed (Subsection 7.4), shows that the two components differing between IT-TIM and ProtoNet outside the objective, namely feature normalization and prototype refitting on the support set, account for -0.14 percentage points at $K = 1$ and $+0.24$ percentage points at $K = 5$. This contrasts with the total advantages of $+2.41$ and $+0.83$ percentage points, respectively. Nearly all of the gain comes from the query-side terms, with the marginal-entropy term responsible for almost all of this improvement. Compared to an identically optimized support-only baseline, adding the marginal term yields $+2.37$ percentage points at $K = 1$ and $+0.61$ percentage points at $K = 5$. The conditional-entropy term adds $+0.28$ and $+0.03$ percentage points, but provides no additional benefit at $K = 5$ once the marginal term is present. Thus, the practical advantage of transductive inference in this task is more

accurately attributed to class-balance enforcement rather than confidence maximization.

The mechanism underlying the observed effect is more specific than initially proposed (Subsection 7.4). Single-class collapse, the degeneracy the term is designed to prevent, does not occur on this benchmark even when the term is omitted. Additionally, the term does not increase prototype displacement: all conditions result in rotations between 5.3 and 5.8 degrees, and the condition with the greatest rotation achieves lower accuracy. Instead, the term redirects a displacement of similar magnitude in a direction that reduces the divergence between the predicted class distribution and uniformity by half. The resulting decision changes are limited in number (5.5% of queries at $K = 1$), but are favorable in outcome (62% convert an error into a correct prediction).

This mechanism also clarifies the conditions under which the observed gain persists. Because the term aligns query predictions with a uniform prior, the benefit depends on the query set being approximately class-balanced. When Dirichlet-sampled imbalance is introduced, the advantage diminishes and reverses, and the sensitivity increases monotonically with the weight assigned to the term, while query-independent inference rules remain unaffected (Subsection 7.5). Replacing the uniform prior with one estimated from an initial inductive prediction removes this sensitivity, but incurs a cost of 0.46 percentage points at $K = 5$ under balanced queries. Therefore, the transductive gain should be interpreted not as an inherent property of the method, but as the result of a class-balance mechanism that operates within a specific range and can be adapted when the deployment prior is unknown.

The primary source of classification error in this evaluation arises from constraints on the label set, rather than from methodological limitations. The class with the lowest accuracy comprises three IUCN GET coastal-shoreline groups that the national reference products do not distinguish individually, and its errors are mainly due to confusion with the spectrally similar open-ocean class (Subsection 7.3). In contrast, the only marine–terrestrial class that the source layers resolve individually, tidal flats, is classified accurately. Aggregating functional groups, whether by necessity or due to limitations in the available reference data, incurs a measurable cost.

The quality of the learned representation is the primary determinant of accuracy, rather than the choice of inference rule. In a controlled ablation that holds the inference rule, classifier, input dimensionality, and input channels constant while varying only the encoder weights (Subsection 6.4), replacing

a randomly initialized encoder with an episodically meta-trained encoder improves accuracy by 8.56 percentage points at $K = 5$ ($p = 0.007$, $d_z = 2.32$). In contrast, replacing inductive with transductive inference under the same encoder improves accuracy by 0.83 percentage points ($p = 0.012$, $d_z = 1.97$). This pattern is consistent across a second encoder family, indicating that the ordering of these effects is not architecture-specific. Within representation learning, most of the benefit derives from having a learned representation: a conventionally supervised encoder recovers 81.8% of the random-to-episodic gap at $K = 5$, but only 62.2% at $K = 1$. These results indicate that the episodic objective contributes most when the support set is smallest, although this contrast reaches significance only at $K = 1$. The primary contribution of the episodic objective is transferability rather than separability; the supervised encoder separates the seen classes slightly better but transfers less effectively.

The crossed experimental design further demonstrates that the value of the inference rule depends on the quality of the learned representation, rather than the reverse. With a randomly initialized encoder, IT-TIM provides no gain over ProtoNet (the difference is slightly negative and not significant), whereas the gain is positive for all learned representations. This outcome aligns with the previously described mechanism: balancing the predicted marginal is beneficial only when the embedding already separates the classes. These findings suggest that practitioners should prioritize optimizing the learned representation over selecting among closely related transductive inference rules.

Among those rules, the behavior of LaplacianShot is determined by its query-graph regularizer: it is advantageous when support is limited and the embedding is compact, but its effectiveness declines as shot count, class count, or within-class spectral dispersion increases (Subsection 7.3). These are not three independent failure modes but one, since each degrades the neighborhood structure the regularizer relies on.

8.2. Practical implications

Since the transductive advantage arises from class-balance enforcement rather than from the objective as a whole, the same benefit may be achievable from a simpler formulation. A meta-trained prototypical classifier with an explicit class-balance constraint would eliminate the per-episode iterative optimization required by IT-TIM, though we did not evaluate this.

Two practical adjustments emerge directly from these results. First, the number of refinement iterations can be reduced at a less than proportional loss in accuracy, since returns diminish with M: $M = 50$ retains 71% of the gain over ProtoNet at 98 ms per episode against 170 ms at the default $M = 100$, and $M = 10$ retains 36% at 30 ms (Table 8). The appropriate setting depends on whether the residual fraction of an already small gain justifies the additional time. Second, when the deployment class prior is non-uniform or unknown, replacing the uniform prior in the marginal-entropy term with one estimated from an initial inductive prediction restores the advantage under imbalance, incurring a cost of 0.46 percentage points at $K = 5$ when the query set is actually balanced.

When the available reference data do not resolve functional groups individually, the merged-class analysis provides specific guidance for label-set design. Subtypes with substantial spectral heterogeneity, such as coastal shorelines that include open-coast, estuarine, and intertidal forms, are likely to experience the greatest reduction in accuracy. Where the source data permit, these subtypes should be maintained as separate classes or supplemented with additional labeled examples, rather than combined.

A small number of labeled sites per class recovers most of the accuracy attainable with extensive supervision. At $K = 5$, IT-TIM achieves approximately 92% of the accuracy of a supervised reference that uses forty times more labeled examples per target class, under the matched protocol on the realm-diverse partition. The marginal benefit of additional labels declines sharply beyond this point (Subsection 6.1). Under the specific conditions of this benchmark, which include a single country and year, map-derived labels, and functionally related groups present during meta-training, these findings indicate an efficient operating point of approximately five labeled sites per functional group. Further investigation is necessary to assess whether this conclusion applies to other regions.

A fundamental condition informs the preceding discussion. Transductive inference necessitates the collective availability of query samples at prediction time. This condition is generally satisfied when mapping a region in batches, as demonstrated by the current benchmark. In contrast, for real-time, sample-by-sample prediction, the inductive baseline remains the most suitable approach.

8.3. Limitations and future work

There is a direct and quantifiable relationship between transductive gain and the balanced-query assumption. As established in Subsection 7.5, this relationship results from the intrinsic properties of the method rather than any external flaw. The advantage of IT-TIM over ProtoNet diminishes and reverses when the query class distribution deviates substantially from uniformity, and sensitivity increases with the weight assigned to the marginal-entropy term. Estimating the prior from an initial inductive prediction can partially mitigate this effect; however, estimation error remains a significant source of uncertainty. The development of principled prior-estimation strategies for few-shot remote sensing remains an open research question. Recent evidence that class-balanced evaluation may overstate few-shot performance (Mohammadi et al., 2024) indicates the broader relevance of this issue.

The generalizability of these results is constrained in three principal ways. First, the dataset includes only a single country and a single calendar year, leaving generalization across climatic gradients and inter-annual variation untested. Although the harmonized typology and phenology-based features are intended to support transferability, this capability was not assessed. Second, the taxonomy-aware partition described in Subsection 7.6 shows that performance declines sharply when six of the seven freshwater classes are excluded from the meta-training pool. Consequently, the headline results should be interpreted as relevant only to scenarios where functionally related groups are present during training. Third, the spatially controlled evaluation in Subsection 7.2 indicates that within-region performance under strict spatial separation is approximately 1 percentage point lower than with randomly sampled data.

Several aspects of the study require further clarification. First, the labels are derived from national thematic products rather than being independently adjudicated at each site (Subsection 3.1), which means that systematic errors in these products are propagated to the benchmark. Validation against independent field evidence for a representative subset was not feasible. Second, the minimum-distance constraint in the spatial analysis applies only within each class, which allows sites of different classes to share pixels. This leads to cross-class feature mixing that the buffer does not address. Third, the representation ablation compared random initialization, conventional supervision, and episodic meta-training across two architectures. However, it did not include self-supervised pre-training, which has become a prominent method for label-efficient remote sensing. The observed threshold relation-

ship, in which the marginal-entropy term benefits all learned representations but is detrimental to untrained ones, suggests that self-supervised embeddings would likely exhibit similar behavior, though this remains to be verified. Fourth, the ablation study was limited to two convolutional families, excluding attention-based encoders for satellite time series. Consequently, the observed ordering of the encoder and inference axes applies specifically to convolutional architectures, rather than to all encoder families.

Two remedies could reduce the aggregation cost quantified in Subsection 7.3, although neither was evaluated here. Hierarchical classification predicts the parent group first and subsequently resolves constituent subtypes, which enables coarse decisions when the source data are reliable. Mixed-membership prototypes represent an aggregated class using several centroids rather than a single one, thereby directly accommodating spectral dispersion. Both approaches require label sets that at least partially distinguish the subtypes, a condition not satisfied by the present benchmark.

9. Conclusion

This study provides a systematic evaluation of transductive few-shot learning for ecosystem functional classification from satellite time series, employing a benchmark that encompasses 24 IUCN GET functional groups across six realms, with labels derived from a rule-based integration of national thematic maps rather than site-level expert annotation. With a unified backbone and consistent evaluation protocol, IT-TIM ranked first across all evaluated shot counts, class counts, and class partitions under balanced query sets, including under spatial controls and when six of the seven freshwater classes were withheld. Its margin over the strongest baseline was small, below 1.3 percentage points at $K = 5$, not statistically significant in three of the IT-TIM versus ProtoNet comparisons, and obtained at about thirteen times the per-episode inference cost. With five labeled sites per class, IT-TIM reached approximately 92% of the accuracy of a supervised reference that used forty times more labeled examples per target class, under a matched protocol on the realm-diverse partition. Analysis of the transductive objective indicates that class-balance enforcement, rather than per-query confidence sharpening, is the primary contributor to transductive improvement for this task. The classification cost of aggregating functional groups to meet sample-availability thresholds was also quantified. This gain, however, depends on the query set being approximately class-balanced; under strong

imbalance the advantage reverses unless the class prior is estimated rather than assumed uniform.

Within a single-country, single-year benchmark, the findings indicate that label-efficient classification aligned with a globally consistent typology is feasible with a small labeled set per functional group; whether this transfers across countries, climatic regions, or years remains untested. Nevertheless, the construction of the label set, especially decisions concerning the merging of spectrally heterogeneous subtypes, imposes direct limitations on attainable accuracy. Mechanistic analysis suggests that a simpler formulation, such as a meta-trained prototypical classifier with an explicit class-balance constraint, may capture much of the transductive benefit without per-episode optimization, though we did not evaluate this. Future research should extend these results to broader climatic gradients, alternative encoders, and deployment scenarios in which query class proportions are unknown or imbalanced. Furthermore, developing GET-aligned few-shot benchmarks in additional regions would enable more robust cross-regional comparisons.

Data and code availability

The code that supports the findings of this study is publicly available at `https://anonymous.4open.science/r/fewshot-biodiversity-167E`. The dataset is available from the corresponding author upon request.

Appendix A. Classification criteria

The criteria for all 24 classes are referenced in Subsection 3.1. Indices and GET codes are consistent with Table 1.

Source abbreviations are as follows: LCM refers to the Land Cover Map (Ministry of Environment, 2022), which comprises 41 classes derived from aerial orthophotography. FTM denotes the Forest Type Map (Korea Forest Service, 2022), derived from aerial orthophotography and field surveys. FM represents FarmMap (MAFRA, 2022), which utilizes aerial orthophotography and cadastral records. WI stands for the National Wetlands Inventory (National Institute of Ecology, 2022), based on field surveys. PRISM indicates MK-PRISM gridded mean air temperature (KMA, 2015 to 2019), collected from ground stations. GSW refers to Global Surface Water (JRC, 2021), based on the Landsat archive from 1984 to 2021. HR denotes HydroRIVERS (WWF, 2020), which is derived from DEM-based hydrography. DEM refers to the national digital elevation model (NGII, 2022),

produced via photogrammetry. CSM indicates the coastal sediment map (KHOA). With the exception of GSW, which provides only long-term water-occurrence frequency, none of these sources are derived from intra-annual medium-resolution optical time series.

Table A.1: Classification criteria for the 24 ecosystem functional groups.

Idx	GET code	Sources	Criteria
0	T2.1.1	FTM, DEM	Coniferous stands; elevation $\geq$ 1,000 m
1	T2.1.2	FTM, DEM	Mixed stands; elevation $\geq$ 1,000 m
2	T2.2.1	FTM, PRISM, DEM	Broadleaf stands; winter mean $<$ 1 °C, summer mean $\leq$ 22 °C, 4–6 non-consecutive months above 10 °C; elevation $<$ 1,000 m
3	T2.2.2	FTM, PRISM, DEM	Coniferous stands; climatic criteria as for T2.2.1; elevation $<$ 1,000 m
4	T2.2.3	FTM, PRISM, DEM	Mixed stands; climatic criteria as for T2.2.1; elevation $<$ 1,000 m
5	T2.4	FTM, PRISM, DEM	Evergreen broadleaf stands; 6–8 non-consecutive months above 10 °C; elevation $<$ 1,000 m
6	T7.1	FM	Fields
7	T7.2	LCM	Pasture and aquaculture class
8	T7.3	FM	Orchards
9	T7.4	LCM	Built-up areas and artificial bare ground
10	T7.5	LCM	Natural and artificial grassland
11	M2.1	LCM	Marine class, seaward of the shoreline zone
12	F1.1	LCM, HR, GSW	Rivers; stream order 1–3; permanent water
13	F1.2	LCM, HR, GSW	Rivers; stream order 4–9; permanent water
14	F1.3	LCM, PRISM	Rivers; winter mean $<$ 0 °C
15	F1.4	LCM, HR, GSW	Rivers; stream order 1–4; seasonal water
16	F1.5	LCM, HR, GSW	Rivers; stream order 5–9; seasonal water
17	F2.2/2.4/2.5	LCM	Lakes; constituent groups not resolved by the source layer
18	F3.3	FM	Rice paddies
19	MT1.1/1.3/1.4	LCM	Areas between marine waters and inland boundaries, excluding tidal flats; constituent groups not resolved by the source layer
20	MT1.2	LCM, CSM	Tidal flats intersected with mudflat substrate
21	TF1.2	LCM, WI	Inland wetlands; woody vegetation
22	TF1.3	WI	Lakes, rivers and wetlands smaller than 8 ha
23	TF1.7	LCM, WI	Inland wetlands; herbaceous vegetation

Appendix B. Static features and label circularity

Three static features are available for each site: elevation and two projected coordinates. However, all experiments in this study employ the no-

static variant. Their contribution is inconsistent rather than uniform: the difference is positive in all ten cells but ranges from +0.24 percentage points on one class partition, where it is not significant at any shot count, to +7.68 on another (Table B.2), for an average of 2.6 percentage points.

This pattern reflects the classification criteria rather than the inherent difficulty of the ecosystems. Several functional groups in this typology are defined in part by elevation or climate. For example, montane forests are distinguished from temperate forests at 1,000 m, freeze-thaw rivers are separated from permanent lowland rivers by mean winter temperature, and marine and marine–terrestrial classes are differentiated relative to the shoreline (Appendix Appendix A). As a result, elevation and coordinates encode aspects of the rules used for label assignment. The largest accuracy gain occurs in the partition where meta-test classes are separated by geography and winter temperature. No gain is observed in the partition where classes differ by forest stand type, and a negative effect is seen under quadrant transfer, where the static variant performs 2.55 percentage points worse than the no-static variant.

We therefore report the no-static variant throughout, at an average cost of 2.6 percentage points. A benchmark that admits projected coordinates as inputs cannot meaningfully test generalization to new locations.

Table B.2: Effect of including the three static features at $K = 5$, by class partition. Mean accuracy over five seeds; p from a paired t-test matched on seed. On partition A no difference is detected at any shot count. Paired t-test: $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$.

Partition	Method	Static	No-static	Δ (pp)
A	ProtoNet	0.855	0.853	+0.24
	IT-TIM	0.868	0.865	+0.35
B	ProtoNet	0.876	0.865	+1.11
	IT-TIM	0.882	0.867	+1.51
C	ProtoNet	0.877	0.848	$+2.85^{**}$
	IT-TIM	0.889	0.857	$+3.20^{**}$
D	ProtoNet	0.834	0.823	$+1.16^{*}$
	IT-TIM	0.851	0.835	+1.59
E	ProtoNet	0.781	0.704	$+7.68^{***}$
	IT-TIM	0.777	0.711	$+6.58^{***}$

Appendix C. Per-class spatial autocorrelation

The per-class distributions presented here support the analysis in Subsection 7.2. The 90 m threshold represents the distance at which the 3×3 neighborhood windows of two sites overlap. All columns report statistics for same-class neighbors. For any-class neighbors, the pooled figures are 18.3% within 90 m, 51.4% within 500 m, and a median distance of 477 m. The difference between same-class and any-class neighbors reaches 10.5 percentage points for class 23, demonstrating that sites of different classes can share pixels, a scenario not addressed by the minimum-distance constraint.

Classes 2, 4, and 9 have no same-class neighbors within 90 m, and three pairs of sites share identical coordinates. In contrast, every site in class 7 has a same-class neighbor within 500 m. However, class 7 is minimally affected by the buffered evaluation described in Subsection 7.2 because its 1,000 sites are distributed across 31 spatially disjoint blocks. A 500 m linkage threshold yields 32 blocks, indicating that the block structure is determined by the mapped polygons rather than by the linkage threshold.

Appendix D. Spatial transfer across quadrants

Episodes are constructed by selecting support and query sets from separate quadrants of the study area (Q0 to Q3; Figure 1), while maintaining a fixed class set from the 18-class meta-training pool. This approach is necessary because the realm-diverse partition does not provide per-class coverage across all quadrants. The configuration controls for geographic shift among previously observed classes; however, it does not assess novel-class transfer, for which the buffered and blocked protocols described in Subsection 7.2 are more appropriate. Notably, one quadrant consistently presents greater difficulty: when held out, it yields an average score of 0.875, compared to 0.913 to 0.917 for the other three quadrants, and it is the weakest quadrant for all four inference rules.

Appendix E. Imbalanced query distributions at $K = 1$

The values correspond to Subsection 7.5, which presents results for the $K = 5$ case and indicates that the effect is approximately twice as large when $K = 1$. Sensitivity is limited to the marginal-entropy term: all conditions with $\lambda_1 > 0$ exhibit significant changes across regimes, whereas the others do

Table C.3: Per-class spatial autocorrelation. Distances are to the nearest site of the same class; blocks are single-linkage clusters at 1 km.

			NN distance (m)			Fraction within	
Idx	GET code	Blocks	Median	q05	q95	90 m	500 m
0	T2.1.1	226	150	30	1,668	34.3	79.2
1	T2.1.2	437	560	108	2,254	3.4	45.9
2	T2.2.1	999	7,564	3,416	11,729	0.0	0.0
3	T2.2.2	997	7,257	3,717	12,119	0.2	0.4
4	T2.2.3	1,000	7,201	3,501	11,620	0.0	0.0
5	T2.4	395	382	67	12,794	8.8	56.7
6	T7.1	936	3,698	530	10,256	0.4	4.5
7	T7.2	31	30	30	42	99.9	100.0
8	T7.3	795	1,650	234	8,891	1.6	15.7
9	T7.4	966	4,120	940	11,854	0.0	2.0
10	T7.5	812	1,470	301	9,312	0.2	14.6
11	M2.1	893	2,004	497	5,981	0.2	5.3
12	F1.1	364	228	60	8,506	14.5	71.4
13	F1.2	281	258	67	2,934	9.4	75.8
14	F1.3	278	293	60	2,789	9.9	69.4
15	F1.4	585	608	42	9,547	13.7	44.8
16	F1.5	425	256	30	3,529	26.7	62.9
17	F2.2/2.4/2.5	698	1,076	212	11,053	1.4	25.5
18	F3.3	666	1,303	294	14,004	0.2	17.4
19	MT1.1/1.3/1.4	338	514	149	2,094	2.6	48.1
20	MT1.2	536	765	283	2,334	0.4	23.7
21	TF1.2	216	88	30	2,360	53.1	81.8
22	TF1.3	80	32	12	193	86.4	96.7
23	TF1.7	339	121	30	6,321	41.2	71.8
All classes pooled		—	741	—	—	17.0	42.2

Table D.4: Within-class spatial transfer over the 18-class meta-training pool of the realm-diverse partition, $K = 5$, mean over four held-out quadrants and five seeds.

Method	Accuracy	Macro-F1	Cohen's κ
ProtoNet	0.908	0.906	0.902
SimpleShot	0.905	0.903	0.899
LaplacianShot	0.897	0.890	0.890
IT-TIM	**0.910**	**0.909**	**0.904**

not. At $\alpha = 0.5$, the default configuration performs 1.71 percentage points lower than ProtoNet ($p = 0.0025$) for every seed, consistent with the result at $K = 5$ (1.62 percentage points, $p = 0.0010$). The primary difference between shot counts is the magnitude of the decline rather than the final ranking. The estimated prior incurs a cost of 1.99 percentage points under balanced queries at $K = 1$, compared to 0.46 at $K = 5$.

Table E.5: Accuracy under Dirichlet(α)-sampled imbalanced query distributions at $K = 1$, realm-diverse partition, mean over five seeds. Δ is the change from $\alpha = \infty$ to $\alpha = 0.5$; $^{***}p < 0.001$, all others not significant.

Method	$\alpha = \infty$	$\alpha = 5$	$\alpha = 1$	$\alpha = 0.5$	Δ (pp)
Query-independent					
ProtoNet	0.759	0.760	0.758	0.758	−0.14
SimpleShot	0.770	0.771	0.769	0.768	−0.20
Support CE only	0.769	0.770	0.768	0.767	−0.21
IT-TIM, varying λ_1					
$\lambda_1 = 0$	0.775	0.776	0.777	0.779	+0.40
$\lambda_1 = 0.25$	0.788	0.781	0.763	0.751	−3.68***
$\lambda_1 = 0.5$	0.790	0.779	0.757	0.745	−4.45***
$\lambda_1 = 1$ (default)	0.790	0.777	0.754	0.741	−4.93***
$\lambda_1 = 1$, estimated prior	0.770	0.771	0.770	0.770	+0.01

References

Belgiu, M., Drăguţ, L., 2016. Random forest in remote sensing: A review of applications and future directions. ISPRS journal of photogrammetry and remote sensing 114, 24–31.

Boudiaf, M., Ziko, I., Rony, J., Dolz, J., Piantanida, P., Ben Ayed, I., 2020. Information maximization for few-shot learning, pp. 2445–2457.

Chen, W., Zhang, Y., Xiao, Z., Chu, J., Wang, X., 2025. Spectral-spatial self-supervised learning for few-shot hyperspectral image classification. arXiv preprint arXiv:2505.12482 .

Chen, W.Y., Liu, Y.C., Kira, Z., Wang, Y.C.F., Huang, J.B., 2019. A closer look at few-shot classification. arXiv preprint arXiv:1904.04232 .

Claverie, M., Ju, J., Masek, J.G., Dungan, J.L., Vermote, E.F., Roger, J.C., Skakun, S.V., Justice, C., 2018. The harmonized landsat and sentinel-2 surface reflectance data set. Remote sensing of environment 219, 145–161.

Finn, C., Abbeel, P., Levine, S., 2017. Model-agnostic meta-learning for fast adaptation of deep networks, in: International conference on machine learning, PMLR. pp. 1126–1135.

Ismail Fawaz, H., Lucas, B., Forestier, G., Pelletier, C., Schmidt, D.F., Weber, J., Webb, G.I., Idoumghar, L., Muller, P.A., Petitjean, F., 2020. Inceptiontime: Finding alexnet for time series classification. Data mining and knowledge discovery 34, 1936–1962.

Keith, D.A., Ferrer-Paris, J.R., Nicholson, E., Bishop, M.J., Polidoro, B.A., Ramirez-Llodra, E., Tozer, M.G., Nel, J.L., Mac Nally, R., Gregr, E.J., et al., 2020a. Indicative distribution maps for ecosystem functional groups-level 3 of iucn global ecosystem typology .

Keith, D.A., Ferrer-Paris, J.R., Nicholson, E., Bishop, M.J., Polidoro, B.A., Ramirez-Llodra, E., Tozer, M.G., Nel, J.L., Mac Nally, R., Gregr, E.J., et al., 2022. A function-based typology for earth's ecosystems. Nature 610, 513–518.

Keith, D.A., Ferrer-Paris, J.R., Nicholson, E., Kingsford, R.T., 2020b. Iucn global ecosystem typology 2.0. Descriptive Profiles for Biomes and ecosystem functional groups .

Landis, J.R., Koch, G.G., 1977. The measurement of observer agreement for categorical data. biometrics , 159–174.

Lee, K., Baek, H., Choi, C.H., Han, S.H., Park, S., 2025. Mapping ecosystem functional groups in the republic of korea based on the iucn global ecosystem typology. Remote Sensing 17, 1659.

Li, Z., Lu, F., Zou, J., Hu, L., Zhang, H., 2024. Generalized few-shot meets remote sensing: Discovering novel classes in land cover mapping via hybrid semantic segmentation framework, in: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE. pp. 2744–2754.

Linke, S., Lehner, B., Ouellet Dallaire, C., Ariwi, J., Grill, G., Anand, M., Beames, P., Burchard-Levine, V., Maxwell, S., Moidu, H., et al., 2019. Global hydro-environmental sub-basin and river reach characteristics at high spatial resolution. Scientific data 6, 283.

Mohammadi, S., Belgiu, M., Stein, A., 2024. Few-shot learning for crop mapping from satellite image time series. Remote Sensing 16, 1026.

Mountrakis, G., Im, J., Ogole, C., 2011. Support vector machines in remote sensing: A review. ISPRS journal of photogrammetry and remote sensing 66, 247–259.

Murray, N.J., Phinn, S.R., Clemens, R.S., Roelfsema, C.M., Fuller, R.A., 2012. Continental scale mapping of tidal flats across east asia using the landsat archive. Remote Sensing 4, 3417–3426.

Pekel, J.F., Cottam, A., Gorelick, N., Belward, A.S., 2016. High-resolution mapping of global surface water and its long-term changes. Nature 540, 418–422.

Pelletier, C., Webb, G.I., Petitjean, F., 2019. Temporal convolutional neural network for the classification of satellite image time series. Remote Sensing 11, 523.

Rußwurm, M., Körner, M., 2020. Self-attention for raw optical satellite time series classification. ISPRS journal of photogrammetry and remote sensing 169, 421–435.

Rußwurm, M., Wang, S., Körner, M., Lobell, D., 2020. Meta-learning for few-shot land cover classification, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE. pp. 788–796.

Sakoe, H., Chiba, S., 1978. Dynamic programming algorithm optimization for spoken word recognition. IEEE transactions on acoustics, speech, and signal processing 26, 43–49.

Santiago Júnior, V.A.d., 2024. Empirical evidence regarding few-shot learning for scene classification in remote sensing images. Applied Sciences 14, 10776.

Snell, J., Swersky, K., Zemel, R., 2017. Prototypical networks for few-shot learning.

Tibshirani, R., Hastie, T., Narasimhan, B., Chu, G., 2002. Diagnosis of multiple cancer types by shrunken centroids of gene expression. Proceedings of the National Academy of Sciences 99, 6567–6572.

Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al., 2016. Matching networks for one shot learning.

Wang, Y., Albrecht, C.M., Braham, N.A.A., Mou, L., Zhu, X.X., 2022. Self-supervised learning in remote sensing: A review. IEEE Geoscience and Remote Sensing Magazine 10, 213–247.

Wang, Y., Chao, W.L., Weinberger, K.Q., Van Der Maaten, L., 2019. Simpleshot: Revisiting nearest-neighbor classification for few-shot learning. arXiv preprint arXiv:1911.04623 .

Wells, L., Dexter, K., Pennington, R., Coutinho, Í., Ramos, D., Phillips, O., Baker, T., Ryan, C., 2025. Satellite remote sensing can operationalise the iucn global ecosystem typology in the biome-diverse north-east of brazil. Frontiers of Biogeography 18.

Wu, J., Hovakimyan, N., Hobbs, J., 2023. Genco: An auxiliary generator from contrastive learning for enhanced few-shot learning in remote sensing, in: ECAI 2023: 26th European Conference on Artificial Intelligence, September 30–October 4, 2023, Kraków, Poland–Including 12th Conference on Prestigious Applications of Intelligent Systems (PAIS 2023), SAGE Publications 1 Oliver's Yard, 55 City Road, London, EC1Y 1SP. pp. 2663–2671.

Xi, B., Zhang, Y., Li, J., Huang, Y., Li, Y., Li, Z., Chanussot, J., 2025. Transductive few-shot learning with enhanced spectral-spatial embedding for hyperspectral image classification. IEEE Transactions on Image Processing 34, 854–868.

Yang, Z., Gong, F.Y., Yim, S.H.L., Yuen, C., 2026. CarbonCLIP: Enhance carbon prediction from satellite imagery via integrated street-view semantics and temporal context training. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing .

Zhao, S., Tu, K., Ye, S., Tang, H., Hu, Y., Xie, C., 2023. Land use and land cover classification meets deep learning: A review. Sensors 23, 8966.

Ziko, I., Dolz, J., Granger, E., Ayed, I.B., 2020. Laplacian regularized few-shot learning, in: International conference on machine learning, PMLR. pp. 11660–11670.