RC2: This paper proposed an ultra-lightweight 30-m Earth embedding from 2000 to 2024 for global land monitoring. Multi-sensor observations from the Landsat series (5, 7, 8, and 9), MODIS Terra, and various land cover products were utilized as input data. ESDNet was designed for embedding generation. Experimental results show the performance of the proposed embedding and ESDNet. Overall,this topic is interesting and the presentation could be further improved.

Response: We sincerely thank the reviewer for the careful summary of the manuscript and for confirming that the topic is of interest. We are also grateful for the comprehensive list of suggestions on the methodological description, the figure and equation captions, and the experimental discussion. We have addressed each of the ten points individually below, and the revisions have substantially improved the clarity, reproducibility, and analytical depth of the manuscript.

2.1 What are differences between the proposed Earth embedding and Alpha Earth embedding?

Response: We thank the reviewer for this important question. The two products share the goal of compressing multi-sensor Earth observation into a globally consistent latent space, but they differ along several orthogonal design axes that lead to complementary, rather than competing, use cases. (1) **Sensor composition and historical depth.** AlphaEarth Foundations assimilates a broad multi-modal corpus that includes Sentinel-2 and Landsat-8/9 optical imagery, Sentinel-1 and ALOS PALSAR-2 radar, GEDI LiDAR, ERA5-Land climate reanalysis, GRACE gravity fields, GLO-30 elevation, and geotagged Wikipedia text, with annual embeddings released for the 2017 to 2024 period (Brown et al., 2025). ESD is, by design, a narrower but temporally deeper product. It is built exclusively on Landsat-5/7/8/9 and MODIS Terra surface reflectance time series, and it spans the full 2000 to 2024 window. Therefore, ESD enables longitudinal analyses across the 25-year record, including the pre-2017 era of Landsat-5 and Landsat-7, that lie outside the temporal coverage of AlphaEarth. (2) **Intra-annual temporal granularity.** The released AlphaEarth product distributes one 64-dimensional embedding per pixel for each year, summarizing the annual signal into a single vector. ESD instead preserves twelve latent steps per year with shape [12, H, W], explicitly representing intra-annual phenology and seasonal hydrology. This makes ESD better suited to phenology-driven downstream applications such as crop type discrimination, surface-water seasonality, and the timing of fire or flood disturbances, for which an annual composite would average out the signal of interest. (3) **Latent-space construction and storage format.** The two products adopt fundamentally different bottleneck designs. AlphaEarth maps each pixel to the mean direction of a von Mises-Fisher distribution on the 63-sphere S^{63}, and the embeddings are regularized with a batch-uniformity objective that promotes a uniform distribution of the embeddings on this hypersphere (Brown et al., 2025). The released product is then quantized to 8 bits per channel for distribution. ESD instead uses the Finite Scalar Quantization (FSQ) approach of Mentzer et al. (2023), which projects each latent dimension onto a small fixed set of scalars. The two schemes are therefore not directly comparable, but they reflect a deliberate difference in priorities. AlphaEarth optimizes for a smooth, continuous embedding field suitable for similarity search and sparse-label transfer, whereas ESD optimizes for a discrete, hardware-efficient representation that compresses one year of the global 30-m land surface to approximately 2.4 TB. (4) **Spatial resolution.** AlphaEarth is delivered at 10 m, whereas ESD is at 30 m. We acknowledge this resolution gap as a current limitation in Section 5.4 and

outline a path toward a 10-m extension via Sentinel-2 integration. (5) **Training objective.** AlphaEarth is trained primarily with self-supervised objectives that include multi-source reconstruction, teacher-student consistency, batch uniformity, and a CLIP-style alignment with geotagged text, on a large and mostly unlabeled corpus (Brown et al., 2025). ESD is instead trained with a hybrid objective combining reconstruction with auxiliary supervision from global land-cover and water products. This explicit semantic supervision biases the latent space toward separability for the target thematic classes and is consistent with the higher downstream classification accuracy reported in Section 4.3, at the cost of slightly larger reconstruction error than a purely self-supervised configuration (Section 5.2, Table 10). In summary, ESD and AlphaEarth occupy distinct points on the joint Pareto front of spatial resolution, temporal depth, and temporal granularity. AlphaEarth is a higher-spatial-resolution, multi-modal, annual product over the recent 8-year window, whereas ESD is a longer-record, intra-annual-resolved optical product at moderate spatial resolution.

2.2 For Fig. 2, please provide a detailed caption to help readers better follow.

Response: We thank the reviewer for this suggestion. The original caption was indeed too brief to allow readers to follow the figure independently of the main text. We have rewritten the caption so that each of the three panels of Figure 2 is introduced separately and its visual elements are explained without requiring the reader to consult Section 3.2. The revised caption now reads as follows.

Revision in manuscript (Figure 2 caption): "*Figure 2: Detailed network architecture and computational workflow of ESDNet. (a) Overall workflow. For each pixel* (x,y) *in an SDC30 tile, the annual input feature sequence is passed to the Encoder, which compresses the signal into* M=12 *monthly tokens. These tokens are discretized by the Finite Scalar Quantization (FSQ) module, which maps the encoder output onto a fixed scalar lattice within a bounded latent space. The quantized monthly tokens are then routed to two branches. The first branch is the Decoder, which reconstructs the input temporal sequence and preserves the spectral-temporal information required for reconstructive fidelity. The second branch consists of the MLP task heads, which generate supervised predictions aligned with the auxiliary thematic products used in training. (b) Encoder architecture. The Encoder begins with* N_1 *strided Conv1D layers that progressively shorten the temporal dimension and enlarge the effective temporal receptive field, followed by* N_2 *residual Conv1D layers for deeper feature extraction, and a final pointwise projection that maps the hidden representation into the latent embedding space before quantization. (c) Decoder architecture. The Decoder mirrors the Encoder by first expanding the latent representation through residual Conv1D layers and then progressively restoring the original temporal length through transposed Conv1D upsampling layers. Tensor shapes are annotated at each stage as [T, C], where T denotes temporal length and C denotes channel width. Here,* T_0 *denotes the annual input sequence length,* M=12 *denotes the monthly tokens at the bottleneck, and* C_0, C_1, *and* C_2 *denote the channel widths at the input, intermediate convolutional, and residual stages, respectively."*

2.3 For ESDNet, how to avoid the high-frequency noise during multimodal feature fusion?

Response: We thank the reviewer for raising this important point. High-frequency noise is not handled by a single explicit denoising module, but is suppressed jointly by the upstream data harmonization and by the architecture of ESDNet itself. We have clarified this mechanism in the revised manuscript. First, the dominant sources of high-frequency artifacts are reduced before the data enter ESDNet. The SDC30 production pipeline harmonizes Landsat and MODIS observations through radiometric normalization, temporal gap filling, and cloud- and shadow-affected observation reconstruction. As a result, a substantial fraction of sensor-dependent noise and short-lived contamination is removed at the data-cube level rather than being passed directly into the embedding network. Second, within ESDNet, the front-end strided Conv1D layers act on the annual input sequence by progressively shortening the temporal dimension and enlarging the effective temporal receptive field. This means that the downstream latent representation is formed from temporally aggregated patterns rather than from isolated short-term fluctuations. In practice, this design reduces the influence of day-to-day observation noise and encourages the network to focus on the more persistent spectral-temporal structure of vegetation phenology, surface-water seasonality, and land-surface change. Third, the Finite Scalar Quantization (FSQ) bottleneck further stabilizes the latent space. Because each latent dimension is discretized onto a fixed scalar lattice, small perturbations that do not exceed the quantization interval are absorbed during rounding and therefore do not propagate to the decoder or the supervised branches. This makes the bottleneck naturally resistant to weak high-frequency fluctuations and is one of the main reasons why the reconstructed outputs are visually smoother and more temporally coherent than the raw input sequence. Finally, the auxiliary supervision also plays a regularizing role. By jointly constraining the latent space with thematic targets, the model is encouraged to retain temporally persistent land-surface structure rather than overfitting to transient local noise.

Revision in manuscript (Section 3.2): "*The ESDNet architecture is engineered to transform time-series spatiotemporal reflectance data into discrete, information-dense embeddings. High-frequency noise is mitigated jointly by the upstream SDC30 harmonization process and by the architecture of ESDNet itself. Before entering the network, Landsat and MODIS observations are harmonized through radiometric normalization, temporal gap filling, and cloud/shadow reconstruction, which reduce short-lived contamination and sensor-dependent artifacts. Within ESDNet, the strided Conv1D layers aggregate information over longer temporal windows, and the FSQ bottleneck further suppresses weak perturbations by discretizing each latent dimension onto a fixed scalar lattice.*"

2.4 Network details should be provided for reproductivity, such as the number of encoder-decoder layers and kernel size.

Response: We thank the reviewer for this helpful suggestion. We agree that the original manuscript did not provide sufficient architectural detail for independent reimplementation. In the revised manuscript, Section 3.2 has been expanded and a new Table 2 has been added to summarize the released ESDNet configuration, including the input and output dimensionalities, the number of downsampling stages and residual blocks, the kernel sizes and strides of the encoder-decoder layers, the temporal compression factor, the embedding dimensionality, and the number of tokens produced per pixel-year.

In brief, the released ESDNet uses three strided Conv1D layers in the Encoder to reduce the temporal dimension, followed by a stack of 10 residual blocks and a pointwise projection into a 6-dimensional latent space. The FSQ bottleneck produces 12 monthly tokens per pixel-year. The Decoder mirrors this structure with residual blocks followed by three transposed Conv1D layers that restore the temporal dimension and reconstruct the six surface-reflectance bands. These details are now explicitly summarized in Table 2 for reproducibility.

Revised Table 2. Architectural hyperparameters of the released ESDNet configuration.

Hyperparameter	**Value**
Input feature dimensionality	13
Reconstruction target dimensionality	6
Hidden channel width C	32
Residual block hidden width	256
Number of stride-2 sampling stages N_1	3
Number of residual blocks N_2	10
Temporal compression factor	8
Embedding dimensionality d	6
Tokens per pixel-year M	12

Revision in manuscript (Section 3.2): We added a short architectural description of the released ESDNet configuration to Section 3.2 and introduced a new Table 2 that explicitly lists the corresponding hyperparameters, including the encoder-decoder depth, kernel sizes, strides, temporal compression factor, embedding dimensionality, and number of output tokens.

"The released ESDNet configuration uses three strided Conv1D layers in the Encoder to progressively reduce the temporal dimension, followed by a stack of 10 residual blocks and a pointwise projection into a 6-dimensional latent space. The FSQ bottleneck produces 12 monthly tokens per pixel-year. The Decoder mirrors this structure with residual blocks followed by three transposed Conv1D layers that restore the temporal dimension and reconstruct the six surface-reflectance bands. The main architectural hyperparameters of the released configuration are summarized in Table 2. Specifically, for the released compression factor of 8, the Encoder applies three Conv1D layers with kernel size 4, stride 2, and padding 1, followed by a Conv1D layer with kernel size 3 and stride 1 before the residual stack. Each residual block contains a Conv1D layer with kernel size 3 and a pointwise Conv1D layer with kernel size 1. The Decoder mirrors this design, beginning with a Conv1D layer with kernel size 3 and stride 1 and then using three transposed Conv1D layers with kernel size 4, stride 2, and padding 1 to restore the temporal dimension."

2.5 For Section 3.3, details are missing for total loss functon. How to choose the reasonable weight parameters for weighted loss? Are there experiments to show the result of different

weights?

Response: We thank the reviewer for this important comment. We have revised Section 3.3 to make the total loss formulation clearer and to explain the rationale for the weighting parameters. In the manuscript, Eq. (1) presents the objective in compact form as the weighted sum of reconstruction and auxiliary supervision terms. For the released ESD configuration, the corresponding weights are fixed at $\alpha = 1.0$ and $\beta = \gamma = 0.1$. The rationale is that reconstruction is the dominant objective of the released data product, while the auxiliary supervision provides a weaker constraint to encourage thematic organization of the latent space.

To address the reviewer's question more directly, we also performed a lightweight sensitivity analysis of the loss weights. Without altering the compact notation of Eq. (1), we fixed $\alpha = 1.0$ and jointly varied the auxiliary supervision strength, reported as $\beta = \gamma \in \{0, 0.03, 0.1, 0.3\}$ (Table R1). This experiment was conducted on a fixed, globally distributed stratified subset, using seven EqualEarth 4W sample blocks over 2020–2022 for training and the FROMGLC_train sample set over 2020–2022 for evaluation, rather than by rerunning the full global production workflow.

The results show a consistent reconstruction–semantics trade-off. When $\beta = \gamma = 0$, reconstruction performance is best (Recon. MAE = 0.0161, CC = 0.9908), but thematic performance is weak (ESA WorldCover OA = 0.206, GLAD-CE F1 = 0.220, GLAD-SW F1 = 0.522). As the auxiliary weight increases, thematic performance improves, but reconstruction fidelity declines. At $\beta = \gamma = 0.30$, semantic performance is strongest (ESA WorldCover OA = 0.683, GLAD-CE F1 = 0.696), but reconstruction degrades noticeably (Recon. MAE = 0.0383, CC = 0.9529). The released setting $\beta = \gamma = 0.10$ provides a more balanced compromise, maintaining strong reconstruction (Recon. MAE = 0.0284, CC = 0.9761) while substantially improving thematic organization (ESA WorldCover OA = 0.671, GLAD-CE F1 = 0.432, GLAD-SW F1 = 0.953). We therefore consider $\alpha = 1.0$ and $\beta = \gamma = 0.1$ to be a reasonable and reproducible setting for the released ESD product.

Revised paragraph (Section 3.3, after Eq. (1)): "*For the released configuration, α was fixed at 1.0 and $\beta = \gamma$ were jointly set to 0.1 as an empirically balanced weighting, so that reconstruction remained the dominant objective while the auxiliary supervision provided a weaker constraint to improve thematic separability.*"

2.6 Please double check all the Eqs and figures to make sure that each term is explained clearly.

Response: We thank the reviewer for this suggestion. We have systematically re-checked every equation and figure in the manuscript, including the symbol definitions, the consistency between the equations and the corresponding figure annotations, and the captions. In particular, the definitions of K_i in Eq. (3) and b_i in Eq. (4) of Section 3.3, which were previously omitted, have now been added immediately under each equation, and all other symbols have been confirmed to be defined at their first appearance.

Revised paragraph (Section 3.3):

Eq. (3): "*where y_k represents the ground-truth label, $\hat{y}_k$ is the predicted probability for class k, K_i denotes the number of classes in supervisory task i, and a_i is a task-specific weight.*"

Eq. (4): "*where v and $\hat{v}$ are the target and predicted biophysical indices, respectively, and b_i is a per-variable weight.*"

2.7 “(b) Supervised Regression Loss” should be “(c) Supervised Regression Loss”.

Response: We sincerely thank the reviewer for catching this typographical error. The subsection has been re-labeled “(c) Supervised Regression Loss” in Section 3.3.

2.8 How to jointly optimize three loss functions? The detailed training strategy could be provided.

Response: We thank the reviewer for this helpful suggestion. We agree that the original manuscript did not describe the optimization strategy with sufficient clarity. In the revised manuscript, we now state explicitly that the three loss terms in Eq. (1) are optimized jointly through their weighted sum in a single end-to-end training process, rather than by staged or alternating training. At each training step, the input mini-batch is forwarded through the Encoder and the FSQ bottleneck to produce the quantized latent representation. The Decoder computes the reconstruction term, while the parallel supervised branches compute the auxiliary supervision terms. These components are combined into a single scalar objective according to Eq. (1), and a single backward pass updates all trainable parameters simultaneously. Gradients through the FSQ quantization step are handled using the straight-through estimator of Mentzer et al. (2023).

Revision in manuscript (Section 3.3, after Eq. (1)):

We added a short paragraph clarifying that the three loss terms in Eq. (1) are optimized jointly as a weighted sum in a single end-to-end training process, with one backward pass per mini-batch and straight-through gradient propagation through the FSQ bottleneck.

Added paragraph (Section 3.3): “*The three loss terms in Eq. (1) are optimized jointly rather than in separate stages. At each training step, the input mini-batch is forwarded through the Encoder and the FSQ bottleneck to produce the quantized latent representation. The Decoder computes the reconstruction term, while the parallel supervised branches compute the auxiliary supervision terms. These components are combined into a single scalar objective according to Eq. (1), and a single backward pass updates all trainable parameters simultaneously.*”

2.9 For Table 4, a discussion for the performance with different band is encouraged to be added.

Response: We thank the reviewer for this suggestion. The original manuscript noted the SWIR2 minimum and the NIR maximum in Section 4.2 but did not explain the systematic differences across the six bands. We have now added a follow-up paragraph immediately after the existing Table 5 discussion in Section 4.2, focusing on how the physical and statistical properties of each spectral region shape the observed MAE and CC values, including the apparent paradoxes of NIR (high MAE with high CC) and SWIR2 (low MAE with relatively low CC).

Added paragraph (Section 4.2): *“A closer inspection of Table 5 reveals systematic differences in reconstructive performance across the six spectral bands that reflect their underlying physical and statistical properties. The visible bands (Blue, Green, and Red) show consistently low MAE values and moderate-to-high CC values, which is consistent with their relatively narrow dynamic range over typical land surfaces. The NIR band exhibits the largest MAE but also the highest CC, indicating that although the absolute reconstruction error increases with its much larger*

reflectance range, the seasonal trajectory is still reproduced faithfully. SWIR1 shows intermediate behavior, while SWIR2 combines the lowest MAE with a comparatively lower CC, which we attribute to its generally low and spatially stable reflectance levels: these reduce absolute error but also provide less variance for the embedding to match."

2.10 What is the potential for this embedding to be deployed in the foundation model?

Response: We thank the reviewer for this forward-looking suggestion. We agree that the potential relationship between ESD and geospatial foundation models should be stated more explicitly. In the revised manuscript, we added a short outlook paragraph in Section 5.4 to clarify that ESD may serve as a compact temporal token representation for sequence-based models, as a candidate target in distillation settings, and as a frozen feature space for sparse-label transfer over long historical periods. We also emphasize that these applications remain prospective and are outside the scope of the present data paper.

Revision in manuscript (Section 5.4):

We added a short outlook paragraph at the end of Section 5.4 discussing the potential of ESD as a compact temporal token representation, a candidate target for distillation, and a frozen feature space for future geospatial foundation-model studies.

Added paragraph (Section 5.4): "***Outlook on Integration with Geospatial Foundation Models:*** *Beyond its standalone use, ESD may also be useful within the emerging geospatial foundation-model ecosystem. Because each pixel-year is represented by 12 quantized temporal tokens stored in uint16 format, ESD provides a compact and temporally structured representation that is substantially lighter than the original reflectance time series while still preserving seasonal information. This makes it a natural candidate for use as a compact temporal input to sequence-based or token-based models, as a target representation in distillation settings, or as a frozen feature space for sparse-label transfer and few-shot downstream applications over long historical periods. The discrete token structure induced by the FSQ bottleneck may also be compatible with masked-token pretraining objectives.*"