MAPLE: Multi-Path Adaptive Propagation with Level-Aware Embeddings for Hierarchical Multi-Label Image Classification

Boshko Koloski[†,1,2] **Marjan Stoimchev**[†,1,2]
Jurica Levatić[1] Dragi Kocev[1] Sašo Džeroski[1]
[1] Jožef Stefan Institute, Ljubljana, Slovenia
[2] Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
[†]These authors contributed equally to this work.
{boshko.koloski, marjan.stoimchev, jurica.levatic}@ijs.si
{dragi.kocev, saso.dzeroski}@ijs.si

Abstract

Hierarchical multi-label classification (HMLC) is essential for modeling structured label dependencies in remote sensing. Yet existing approaches struggle in *multi-path* settings, where images may activate multiple taxonomic branches, leading to underuse of hierarchical information. We propose MAPLE (*Multi-Path Adaptive Propagation with Level-Aware Embeddings*), a framework that integrates (i) *hierarchical semantic initialization* from graph-aware textual descriptions, (ii) *graph-based structure encoding* via graph convolutional networks (GCNs), and (iii) *adaptive multi-modal fusion* that dynamically balances semantic priors and visual evidence. An *adaptive level-aware objective* automatically selects appropriate losses per hierarchy level. Evaluations on CORINE-aligned remote sensing datasets (AID, DFC-15, and MLRSNet) show consistent improvements of up to +42% in few-shot regimes while adding only 2.6% parameter overhead, demonstrating that MAPLE effectively and efficiently models hierarchical semantics for Earth observation (EO).

1 Introduction

Remote sensing image (RSI) classification demands methods that respect the hierarchical organization of land cover types [1]. While modern deep learning excels at flat multi-label classification (MLC) [2, 3], it overlooks taxonomic structure in standards like CORINE [4], limiting utility in environmental monitoring, urban planning, and climate assessment.

Hierarchical multi-label classification (HMLC) models label dependencies across levels [5–7], but faces key limitations: (i) many assume *single-path* hierarchies and fail when images belong to multiple branches; (ii) *network-based* designs [8, 9] are computationally heavy while *loss-based* formulations [10, 11] miss long-range dependencies; and (iii) most remain purely supervised despite prevalent low-label regimes in satellite imagery [12–15].

We present *Multi-Path Adaptive Propagation with Level-Aware Embeddings* (MAPLE), a hierarchical multi-label image classification framework for remote sensing that explicitly encodes *multi-path* structure. MAPLE initializes label nodes with *hierarchical semantic embeddings* from contextual

[*] Correspondence to boshko.koloski@ijs.si and marjan.stoimchev@ijs.si

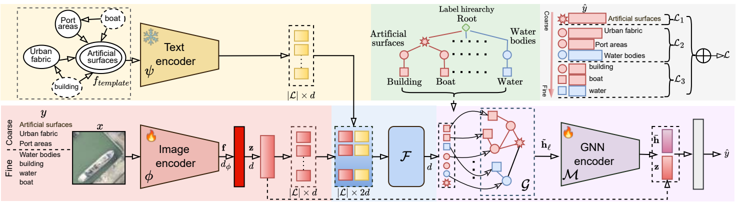


Figure 1: MAPLE architecture overview. The framework processes an input image with a ViT encoder that uses hierarchy-specific class tokens. A GCN refines these token embeddings by propagating information along the label taxonomy. Finally, visual features and refined semantic embeddings are fused via adaptive gating to produce level-aware classifications.

templates; fuses these with Vision Transformer (ViT) [16] features via *adaptive multimodal gating*; and refines representations through *graph-based propagation* on the taxonomy. A unified prediction head with an *adaptive level-aware objective* yields consistent predictions across hierarchy levels without manual loss tuning.

Contributions. (i) We introduce a multi-token transformer with graph-based hierarchical reasoning for multi-path HMLC in RSI classification; (ii) we construct CORINE-aligned *multi-path* hierarchies for AID[17], DFC-15[18], and MLRSNet[19], and evaluate MAPLE across nine datasets spanning remote sensing, medical imaging, and fine-grained visual categorization to assess broader generalizability beyond EO; and (iii) we demonstrate strong performance under limited annotation budgets, achieving up to 42% gains over flat baselines with only 2.6% parameter overhead, highlighting MAPLE's accuracy-efficiency balance for EO applications [15].

2 Methodology

2.1 Problem Formulation

Given an input image $\mathbf{x} \in \mathbb{R}^{C\times H\times W}$ and a hierarchical label graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where nodes $\mathcal{V}$ represent labels across L levels and edges $\mathcal{E}$ encode parent-child relationships, our goal is to predict a multi-label vector $\hat{\mathbf{y}} \in \{0, 1\}^{|\mathcal{V}|}$ that respects the hierarchical constraints.

2.2 Hierarchical Semantic Initialization

Each node $\ell \in \mathcal{L}$ is initialized with a semantic embedding $\mathbf{e}_\ell^{(0)} \in \mathbb{R}^d$ derived from both its label and its position in the hierarchy. Instead of random initialization, we generate a contextual description $\tau(\ell)$ that combines the node name with its parent and child relations, encoded through a pre-trained sentence transformer ψ and linearly projected to the model dimension:

$$\mathbf{e}_\ell^{(0)} = \mathrm{norm}(\mathbf{W}_\psi\, \psi(\tau(\ell)))$$

where $\mathbf{W}_\psi$ is a learnable projection and $\mathrm{norm}(\cdot)$ denotes L2 normalization. This initialization provides semantically meaningful embeddings that reflect the hierarchical organization of the taxonomy. A detailed example is shown in Appendix A.2.

2.3 Visual Representation and Multi-Token Transformer

We extend the standard ViT by introducing $M = |\mathcal{V}|$ learnable class tokens, one for each node in the hierarchy. These tokens $T_{\text{CLS}} \in \mathbb{R}^{M\times d}$ are prepended to image patch tokens T_p, forming the input sequence $T = [T_{\text{CLS}} \| T_p]$. This multi-token design allows each class token to learn specialized, label-specific patterns by attending to relevant image regions. The ViT processes the input through self-attention mechanisms, extracting the CLS token representation as the global image descriptor. Visual features are projected into a common d-dimensional latent space through a learnable linear transformation to enable effective multimodal fusion with semantic embeddings.

2.4 Graph-Based Hierarchical Refinement

To explicitly leverage the label hierarchy, we refine the class token embeddings using a graph neural network (GNN). The output tokens from the ViT encoder serve as initial node features for a GraphSAGE-style [20] message passing process on the graph $\mathcal{G}$. This refinement is performed iteratively for L_g layers:

$$\mathbf{H}^{(k+1)} = \text{GELU}\left(\text{LayerNorm}\left(\mathcal{M}^{(k)}(\mathbf{H}^{(k)}, \mathcal{E}) + \mathbf{H}^{(k)}\right)\right), \tag{1}$$

where $\mathcal{M}^{(k)}$ is the message passing function that aggregates information from neighboring nodes. The residual connection preserves original node information while learning relational context. This process allows parent nodes to aggregate features from their children and vice versa, creating robust, hierarchy-aware embeddings.

2.5 Adaptive Multimodal Fusion

To produce the final representations, we combine visual information with semantic, hierarchy-aware information through an adaptive gating mechanism. The visual representation is expanded to match all hierarchy nodes by replication. For each node v, we compute fusion weights $\boldsymbol{\gamma}_v \in [0,1]^d$ that dynamically balance the contribution of the visual representation $\mathbf{z} \in \mathbb{R}^d$ and the refined token embedding $\mathbf{e}_v \in \mathbb{R}^d$:

$$\tilde{\mathbf{h}}_v = \boldsymbol{\gamma}_v \odot \mathbf{e}_v + (1 - \boldsymbol{\gamma}_v) \odot \mathbf{z}, \tag{2}$$

where $\odot$ denotes element-wise multiplication. The fused representation $\tilde{\mathbf{h}}_v \in \mathbb{R}^d$ allows the model to decide, on a per-label basis, whether to rely more on visual cues or learned taxonomic context. The gating weights are computed through a learned network with LayerNorm and sigmoid activation, enabling node-specific fusion strategies across hierarchical levels.

2.6 Unified Prediction Head and Training Objective

After graph refinement, node embeddings are mean-pooled and concatenated with the original visual features. A single linear transformation maps this combined representation to classification logits for all hierarchy nodes simultaneously. The output logits are partitioned by level to enable level-specific training.

To accommodate varying label structures at different hierarchy depths, we use an adaptive loss function. At each level t, we apply Cross-Entropy (CE) loss if the ground-truth vector $\mathbf{y}_t$ is single-label ($\|\mathbf{y}_t\|_1 = 1$) and Binary Cross-Entropy (BCE) loss otherwise. The total loss averages across all L levels: $\mathcal{L} = \frac{1}{L}\sum_{t=1}^{L} \mathcal{L}_{\text{adaptive}}$. This formulation enables simultaneous supervision across all semantic resolutions while automatically adapting to label distribution characteristics at each hierarchical level.

3 Experiments

We evaluate MAPLE on three remote sensing datasets (AID, DFC-15, and MLRSNet) with CORINE-aligned hierarchies [4, 17–19]. The model employs a ViT-B/16 backbone [16] pre-trained on ImageNet [21] and a 2-layer GraphSAGE network [20]. Performance is reported as mean Area Under the Precision-Recall Curve (AUPRC) over three runs (computed with `scikit-learn` [22]). Dataset statistics, hierarchy construction, and implementation details are provided in Appendices A.1–A.2.

Hierarchical vs. Flat Classification. Table 1 shows that MAPLE consistently outperforms the flat MLC baseline (leaf labels only) across all datasets, with gains from 0.56% (MLRSNet) to 3.61% (AID). Improvements are strongest on AID, where the moderate dataset size and rich hierarchical structure allow effective exploitation of CORINE relations [4]. On the larger MLRSNet, MAPLE still yields consistent improvements despite the strong baseline.

Few-Shot Learning. Table 2 highlights MAPLE's robustness under limited supervision. The largest relative gains occur at 4-shot (AID +25.0%, DFC-15 +6.6%, MLRSNet +18.5%), confirming that hierarchical propagation acts as a strong inductive bias and improves label efficiency across datasets.

Table 1: Hierarchical vs. flat classification (AU$\overline{\text{PRC}}$ %). MAPLE outperforms the flat baseline at all hierarchy levels (l_1–l_3) and leaf nodes.

	MAPLE					
Dataset	l_1	l_2	l_3	Leaf	**MLC**	Δ (%)
AID	95.31	94.32	84.83	**87.25**	84.21	+3.6
DFC-15	99.62	98.33	98.37	**98.71**	98.65	+0.1
MLRSNet	98.23	97.88	96.77	**96.71**	96.17	+0.6

Table 2: Few-shot performance (AU$\overline{\text{PRC}}$) with K-shot training per class. $\mu \pm \sigma$ over 3 runs. Δ shows MAPLE's improvement over the MLC baseline.

Method	4-shot	8-shot	12-shot	16-shot
		AID		
MLC	0.286±0.018	0.334±0.012	0.341±0.025	0.310±0.009
MAPLE	**0.357±0.021**	**0.371±0.017**	**0.396±0.039**	**0.440±0.046**
Δ (%)	+25.0	+11.1	+16.1	+41.9
		DFC-15		
MLC	0.541±0.044	0.541±0.012	0.790±0.008	0.807±0.021
MAPLE	**0.577±0.058**	**0.747±0.036**	**0.751±0.021**	**0.854±0.012**
Δ (%)	+6.6	+38.1	-4.9	+5.8
		MLRSNet		
MLC	0.528±0.096	0.545±0.000	0.675±0.031	0.743±0.028
MAPLE	**0.626±0.011**	**0.752±0.012**	**0.762±0.008**	**0.801±0.017**
Δ (%)	+18.5	+38.0	+12.9	+7.8

Table 3: Comparison with state-of-the-art HMLC methods on *leaf-level* AU$\overline{\text{PRC}}$ (↑). Best in **bold**, second-best <u>underlined</u>.

Method	**AID**	**DFC-15**	**MLRSNet**
C-HMCNN [11]	0.764	0.962	0.792
HiMulConE [7]	<u>0.770</u>	<u>0.970</u>	<u>0.865</u>
HMI [23]	0.647	0.923	0.437
MAPLE (Ours)	**0.872**	**0.987**	**0.967**

Comparison with State of the Art. Table 3 compares MAPLE with representative state-of-the-art HMLC methods, including C-HMCNN [11], HiMulConE [7], and HMI [23]. Evaluated at the *leaf* level, MAPLE attains the best AU$\overline{\text{PRC}}$ on all three datasets by a clear margin, confirming the benefits of multi-path propagation and level-aware embeddings.

Beyond EO, MAPLE generalizes effectively to medical imaging and fine-grained categorization, achieving strong gains on complex hierarchies (PadChest +21.9%, ETHEC +10.4%) [24, 25]. Full results and visualizations are included in Appendices A.5–A.8.

4 Discussion

MAPLE effectively captures hierarchical dependencies in remote sensing imagery, outperforming flat baselines with only 2.6% additional parameters and negligible latency (Appendix A.6). Key insights include: (i) consistent gains across datasets (0.56–3.61%), (ii) up to 42% improvement in few-shot regimes, confirming taxonomy-driven inductive bias, and (iii) strong generalization to non-EO hierarchies such as PadChest (+21.9%).

The results show that explicitly encoding class hierarchies not only improves predictive accuracy but also enhances stability and robustness to label noise. By constraining predictions along valid hierarchical paths, MAPLE mitigates inconsistent label assignments and promotes semantically coherent outputs. This structural regularization effect becomes especially beneficial in low-data settings, where flat classifiers often overfit or ignore fine-grained relations between categories.

Beyond accuracy, MAPLE offers interpretable predictions by revealing how decisions propagate through the hierarchy, a property valuable for trust and transparency in applications such as environmental monitoring and medical analysis.

Its efficiency and label efficiency make MAPLE practical for large-scale or operational EO scenarios where annotations are limited. While gains narrow on massive datasets, consistent improvements across all regimes highlight the importance of hierarchical modeling for structured output prediction. Future work will extend MAPLE to semi-supervised and contrastive learning and explore automatic hierarchy discovery to reduce reliance on expert-defined taxonomies.

Acknowledgments

We acknowledge the financial support of the Slovenian Research and Innovation Agency (ARIS) through the core research programme P2-0103 (Knowledge Technologies), projects J1-3033, J2-2505, J2-4452, J2-4660, J3-3070, J4-3095, J5-4575, J7-4636, J7-4637, and N2-0236. The work of the BK was supported by the Young Researcher grant PR-12394.

References

[1] G. Sumbul, M. Charfuelan, B. Demir, and V. Markl. BigEarthNet: A Large-Scale Benchmark Archive for Remote Sensing Image Understanding. *IEEE International Geoscience and Remote Sensing Symposium*, 12(2):5901–5904, 2019.

[2] Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. Multi-label image recognition with graph convolutional networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5177–5186, 2019.

[3] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017.

[4] Referans Copernicus. Corine land cover. *Copernicus Land Monitoring Service. L. Monit. Serv*, 2018.

[5] Jurica Levatić, Dragi Kocev, and Sašo Džeroski. The importance of the label hierarchy in hierarchical multi-label classification. *Journal of Intelligent Information Systems*, 45(2):247–271, October 2015. ISSN 1573-7675.

[6] Jonatas Wehrmann, Ricardo Cerri, and Rodrigo Barros. Hierarchical multi-label classification networks. In *International conference on machine learning*, pages 5075–5084. PMLR, 2018.

[7] Shu Zhang, Ran Xu, Caiming Xiong, and Chetan Ramaiah. Use all the labels: A hierarchical multi-label contrastive learning framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16660–16669, June 2022.

[8] Xinqi Zhu and Michael Bain. B-cnn: branch convolutional neural network for hierarchical classification. *arXiv preprint arXiv:1709.09890*, 2017.

[9] Brendan Kolisnik, Isaac Hogan, and Farhana Zulkernine. Condition-CNN: A hierarchical multi-label fashion image classification model. *Expert Systems with Applications*, 182:115195, 2021.

[10] Jingzhou Chen, Peng Wang, Jian Liu, and Yuntao Qian. Label relation graphs enhanced hierarchical residual network for hierarchical multi-granularity classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4858–4867, 2022.

[11] Eleonora Giunchiglia and Thomas Lukasiewicz. Coherent hierarchical multi-label classification networks. *Advances in neural information processing systems*, 33:9662–9673, 2020.

[12] Yi Wang, Conrad M. Albrecht, Nassim Ait Ali Braham, Lichao Mou, and Xiao Xiang Zhu. Self-supervised learning in remote sensing: A review. *IEEE Geoscience and Remote Sensing Magazine*, 10(4):213–247, 2022.

[13] Jesper E Van Engelen and Holger H Hoos. A survey on semi-supervised learning. *Machine learning*, 109(2):373–440, 2020.

[14] Xiangli Yang, Zixing Song, Irwin King, and Zenglin Xu. A survey on deep semi-supervised learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(9):8934–8954, 2022.

[15] Anastasiia Safonova, Gohar Ghazaryan, Stefan Stiller, Magdalena Main-Knorn, Claas Nendel, and Masahiro Ryo. Ten deep learning techniques to address small data problems with remote

sensing. *International Journal of Applied Earth Observation and Geoinformation*, 125:103569, 2023.

[16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=YicbFdNTTy`.

[17] Y. Hua, L. Mou, and X.X. Zhu. Relation Network for Multi-label Aerial Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 2019.

[18] Y. Hua, L. Mou, and X.X. Zhu. Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional LSTM network for multi-label aerial image classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 149:188–199, 2019.

[19] Q. Xiaoman Qi, Z. Panpan, W. Yuebin, Z. Liqiang, P. Junhuan, W. Mengfan, C. Jialong, Z. Xudong, Z. Ning, and P.M.Takis. MLRSNet: A multi-label high spatial resolution remote sensing dataset for semantic scene understanding. *ISPRS Journal of Photogrammetry and Remote Sensing*, 169:337–350, 2020.

[20] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

[21] J. Deng, W. Dong, R. Socher, L.J. Li, L. Kai, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[23] Bo Xiong, Michael Cochez, Mojtaba Nayyeri, and Steffen Staab. Hyperbolic embedding inference for structured multi-label prediction. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 33016–33028. Curran Associates, Inc., 2022.

[24] Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria de la Iglesia-Vayá. Padchest: A large chest x-ray image dataset with multi-label annotated reports. *Medical Image Analysis*, 66:101797, 2020. ISSN 1361-8415.

[25] Ankit Dhall, Anastasia Makarova, Octavian Ganea, Dario Pavllo, Michael Greeff, and Andreas Krause. Hierarchical image classification using entailment cone embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 836–837, 2020.

[26] Manuel Alejandro Rodríguez, Hasan AlMarzouqi, and Panos Liatsis. Multi-label retinal disease classification using transformers. *IEEE Journal of Biomedical and Health Informatics*, 27(6): 2739–2750, 2022.

[27] Casper Winsnes, Devin Sullivan, Elizabeth Park, Emma Lundberg, Martin Hjelmare, and Phil Culliton. Human protein atlas image classification, 2018. URL `https://kaggle.com/competitions/human-protein-atlas-image-classification`. Kaggle Competition.

[28] Wei Ouyang, Casper F Winsnes, Martin Hjelmare, Anthony J Cesnik, Lovisa Åkesson, Hao Xu, Devin P Sullivan, Shubin Dai, Jun Lan, Park Jinmo, et al. Analysis of the human protein atlas image classification competition. *Nature methods*, 16(12):1254–1261, 2019.

[29] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012.

[30] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *2013 IEEE International Conference on Computer Vision Workshops*, pages 554–561, 2013.

[31] World Health Organization et al. The icd-10 classification of mental and behavioural disorders: clinical descriptions and diagnostic guidelines. *World Health Organization*, 362, 1992.

[32] Peng Xia, Xingtong Yu, Ming Hu, Lie Ju, Zhiyong Wang, Peibo Duan, and Zongyuan Ge. Hgclip: Exploring vision-language models with graph representations for hierarchical understanding. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 269–280, 2025.

[33] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

[34] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mpnet: Masked and permuted pre-training for language understanding. *Advances in neural information processing systems*, 33:16857–16867, 2020.

[35] Thomas Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 09 2017.

[36] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[37] Boshko Koloski, Andrei Margeloiu, Xiangjian Jiang, Blaž Škrlj, Nikola Simidjievski, and Mateja Jamnik. Llm embeddings for deep learning on tabular data, 2025. URL `https://arxiv.org/abs/2502.11596`.

[38] Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*, 2024.

[39] Kenneth Ward Church. Word2vec. *Natural Language Engineering*, 23(1):155–162, 2017.

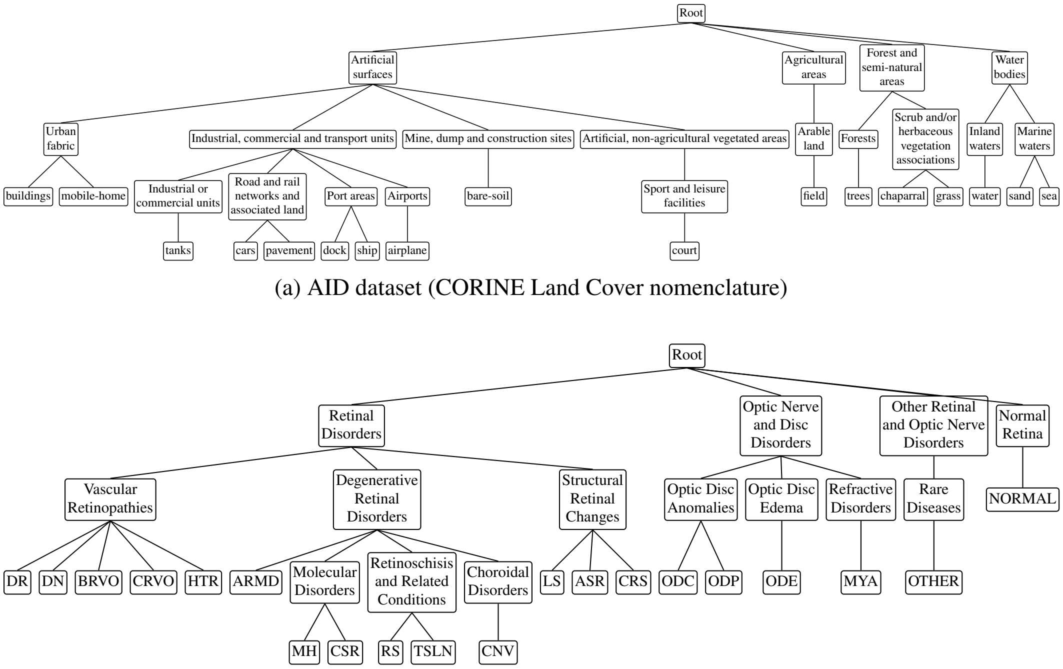
[40] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

A Appendix

A.1 Datasets and Hierarchy Construction

To assess the broader applicability of our hierarchical multi-label learning approach beyond EO, we evaluated MAPLE across nine datasets spanning three distinct domains. Our evaluation includes three publicly available RSI datasets: AID [17], DFC-15 [18], and MLRSNet [19]; three medical imaging datasets: MuRed [26], HPA [27, 28], and PadChest [24]; and three fine-grained visual categorization (FGVC) benchmark datasets: Oxford Pets-37 [29], Stanford Cars [30], and ETHEC [25]. These datasets exhibit varying characteristics in terms of their original label structures and hierarchical organization. The remote sensing and medical imaging datasets are inherently multi-label at the leaf level, as images can contain multiple land cover types or medical conditions simultaneously. The FGVC datasets, in contrast, already possess established hierarchical taxonomies that reflect natural categorical relationships within their respective domains.

Since comprehensive HMLC image datasets are limited, we adapted these datasets to the hierarchical multi-label setting through domain-appropriate approaches. For the RSI datasets, we constructed hierarchical label structures by systematically mapping the inherently multi-label leaf categories to the CORINE Land Cover (CLC) nomenclature [4]. The CLC provides a comprehensive and standardized taxonomy of land cover classes across multiple hierarchical levels, enabling us to create meaningful hierarchical relationships while preserving the multi-label nature at the leaf level. For the medical imaging datasets, we organized the inherently multi-label medical conditions into hierarchical structures based on the International Classification of Diseases, 10th Revision (ICD-10) codes [31], a clinically validated classification system. This approach ensures that the resulting hierarchies reflect genuine medical taxonomic relationships while maintaining the multi-label characteristics essential for realistic diagnostic scenarios. For the FGVC datasets, we used the existing hierarchical structures inherent to each domain, such as the natural breed taxonomies for Oxford Pets-37 and manufacturer-model relationships for Stanford Cars, which already provide meaningful hierarchical organizations suitable for hierarchical classification tasks.



(a) AID dataset (CORINE Land Cover nomenclature)

(b) MuRed dataset (ICD-10 classification)

Figure 2: Examples of constructed label hierarchies for (a) the AID dataset, derived from the CORINE Land Cover (CLC) nomenclature, and (b) the MuRed dataset, with abbreviations in leaf labels corresponding to ICD-10 codes of disease names.

When direct mapping to established nomenclatures was not feasible due to dataset-specific terminologies or emerging categories, we supplement our approach by querying ChatGPT, followed by a manual inspection, to generate appropriate hierarchical placements [32]. This hybrid strategy ensures comprehensive coverage while maintaining the benefits of standardized taxonomic structures, providing a solid foundation for hierarchical learning across diverse visual recognition tasks. Figure 2 illustrates representative examples of the constructed hierarchies, demonstrating how the CORINE Land Cover nomenclature and ICD-10 codes translate into structured taxonomic relationships for the remote sensing and medical domain datasets, respectively.

Figure 3 provides representative examples from all nine datasets, illustrating the diversity of visual content and the varying complexity of hierarchical structures across domains. The hierarchical organizations range from relatively simple two-level structures (e.g., Oxford Pets-37 and Stanford Cars) to complex multi-level taxonomies with up to six hierarchical levels (e.g., PadChest). Table 4 presents detailed characteristics of all datasets, including the number of labels at each hierarchical level and the dataset splits used for evaluation.

A.1.1 Remote Sensing Image Datasets

The AID dataset consists of 3,000 aerial images with a resolution of 600×600 pixels, originally categorized into 30 scene classes for single-label classification [17]. For our hierarchical multi-label extension, we mapped these scene categories to the CORINE Land Cover nomenclature, resulting in a four-level hierarchy with 35 labels organized across hierarchical levels, providing a structured classification framework for aerial scene understanding. The DFC-15 dataset, originating from the 2015 IEEE GRSS Data Fusion Contest, comprises 3,341 image patches with a resolution of 600×600 pixels [18]. Originally designed for semantic segmentation, we adapted it for hierarchical classification by organizing the labels into a three-level hierarchy with 17 distinct labels based on land cover taxonomies. The MLRSNet dataset includes 109,151 images with a resolution of 256×256 pixels, originally designed for multi-label scene classification with 60 categories [19]. The hierarchical version expands this structure to include 104 labels organized across four hierarchical

Table 4: Summary of the datasets used in this study. N denotes the total number of images in each dataset, while N_{train} and N_{test} represent the number of images in the training and test sets, respectively. $|\mathcal{L}|$ indicates the number of unique labels at each hierarchical level, where ℓ corresponds to the deepest (leaf) level.

	N	N_{train}	N_{test}	$\lvert\mathcal{L}\rvert$						
Dataset				1	2	3	4	5	6	ℓ
AID	3,000	2,400	600	4	9	15	7	-	-	17
DFC-15	3,341	2,672	669	3	7	7	-	-	-	8
MLRSNet	109,151	87,336	21,815	7	15	22	60	-	-	60
MuRed	2,208	1,764	444	4	8	17	5	-	-	20
HPA	31,072	24,837	6,235	6	28	-	-	-	-	28
PadChest	121,230	97,203	24,027	2	5	9	9	7	2	19
OxfordPets-37	7,349	3,680	3,669	2	37	-	-	-	-	37
Stanford Cars	16,185	8,144	8,041	9	196	-	-	-	-	196
ETHEC	47,978	42,929	5,049	6	21	135	561	-	-	561

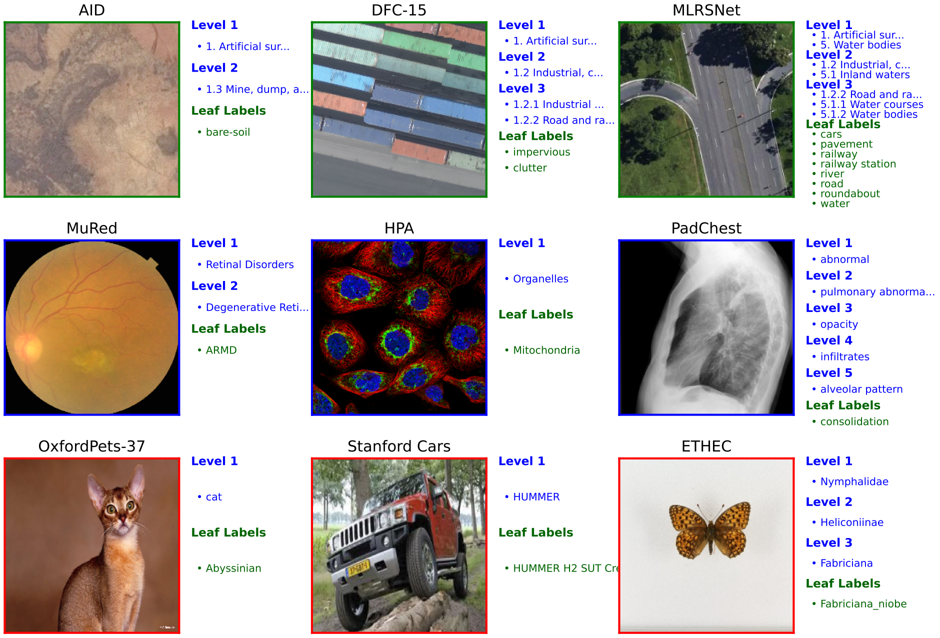


Figure 3: Representative examples from the nine datasets used in our evaluation, showing sample images alongside their corresponding hierarchical label structures. The datasets span three domains: remote sensing (AID, DFC-15, MLRSNet), medical imaging (MuRed, HPA, PadChest), and fine-grained visual categorization (OxfordPets-37, Stanford Cars, ETHEC). Each hierarchy displays the multi-level taxonomic organization from coarse-grained categories at Level 1 to fine-grained leaf labels.

levels using CORINE Land Cover mapping, making it the most complex dataset among the RSI datasets considered in this study.

A.1.2 Medical Image Datasets

The Multi-label Retinal Disease (MuRed) dataset [26] is a publicly available collection of 2,208 retinal fundus images originally designed for MLC of retinal diseases. For our hierarchical extension, we organized the 20 original disease labels using ICD-10 codes[31], resulting in a four-level hierarchy with 34 labels. The Human Protein Atlas (HPA) Image Classification dataset [27, 28] contains 28 distinct protein labels and consists of 31,072 samples. The hierarchical extension organizes the 28 protein labels into a two-level hierarchy with 34 total labels based on cellular localization and functional relationships. The PadChest dataset is a large-scale chest X-ray dataset comprising 160,868 images from 67,000 patients, acquired at Hospital San Juan (Spain) between 2009 and 2017

[24]. The original dataset contains 193 labels, including 174 radiographic findings, 19 differential diagnoses, and 104 anatomical locations, already organized hierarchically and mapped to the Unified Medical Language System (UMLS). Approximately 27% of the labels were manually annotated by board-certified radiologists. We utilized a subset of 121,230 samples and organized them into a refined hierarchy of 32 labels across six hierarchical levels based on ICD-10 codes, maintaining the clinical relevance of the original taxonomic organization.

A.1.3 Fine-Grained Visual Categorization Datasets

The Oxford Pets-37 dataset comprises 7,349 images of 37 different pet breeds[29]. We organized the breeds into a two-level hierarchy based on pet species (cats vs. dogs). The Stanford Cars dataset contains 16,185 images of 196 car models[30]. We structured this into a two-level hierarchy based on manufacturer relationships, creating 9 top-level manufacturer categories and 196 specific model categories. The ETHEC dataset, with 47,978 images spanning 561 categories across four hierarchical levels, represents one of the most comprehensive fine-grained datasets with inherent hierarchical structure. Originally designed for hierarchical classification, this dataset provides a natural four-level taxonomy that progresses from broad categorical distinctions to specific sub-categories, offering an extensive evaluation of hierarchical classification capabilities in complex taxonomic structures.

A.2 Implementation Details

MAPLE is implemented in `PyTorch Lightning` with a ViT-B/16 backbone (ImageNet initialization). We introduce one learnable class token per label node in the hierarchy, departing from the standard single-token design. This multi-token approach allows each token to specialize in detecting its corresponding semantic category by attending to relevant image regions.

Graph-Based Refinement: A two-layer GraphSAGE network implemented in `PyTorch Geometric (PyG)` performs iterative message passing on the label graph. Each layer aggregates information from neighboring nodes (parents and children) in the hierarchy, with residual connections, LayerNorm, and GELU activation ensuring stable training. Dropout (rate=0.1) is applied between layers for regularization.

Image Processing: RGB images resized to 224×224 pixels; latent dimension d=768 matches the ViT-B/16 output dimension.

Augmentations: Random horizontal flips, color jitter (brightness=0.2, contrast=0.2, saturation=0.2), and random resized crops (scale=0.8 to 1.0).

Optimization: AdamW optimizer with base learning rate 1×10^{-4}, weight decay 1×10^{-2}, cosine learning rate scheduler with linear warmup (10 epochs), batch size 16, 150 epochs total. Mixed precision training (FP16) is employed to reduce memory consumption and accelerate training.

Hierarchical Semantic Initialization (Sentence Transformer). For transformer-based initialization, we employ the `all-mpnet-base-v2` model from Hugging Face `Sentence Transformers` [33, 34]. For each node in the hierarchy, we construct a short natural-language prompt encoding its taxonomic context:

```
The category '[label]' which is a subcategory of [parent] and
includes subcategories like [child_1, child_2, child_3].
```

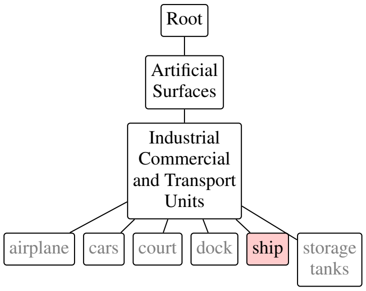
If a node has no children, the final clause is omitted. These prompts are encoded into 768-dimensional sentence embeddings, projected to the model dimension d via a learnable linear layer, and L2-normalized to initialize hierarchy-specific class tokens.

Figure 4 illustrates this process for a CORINE-derived branch where only the *ship* class is active. The left panel shows the corresponding path; the right panel presents the instantiated prompts for both the parent and leaf nodes.

Adaptive Multimodal Fusion: Visual features from the ViT encoder are replicated for all nodes and concatenated with semantic embeddings. A learned gating network (linear layer + LayerNorm +

`https://github.com/Lightning-AI/pytorch-lightning`

`https://pytorch-geometric.readthedocs.io/en/latest/`



Parent node prompt (*Industrial, Commercial and Transport Units*):
`The category 'Industrial, Commercial and Transport Units' which is a subcategory of Artificial Surfaces and includes subcategories like airplane, cars, court, dock, ship, and storage tanks`

Leaf node prompt (*ship*):
`The category 'ship' which is a subcategory of Industrial, Commercial and Transport Units.`

Embedding: $\mathbf{e}_\ell^{(0)} = \mathrm{norm}\big(\mathbf{W}_\psi\, \psi(\tau(\ell))\big)$

Figure 4: Hierarchical semantic initialization using the Sentence Transformer on a CORINE-derived path. Left: subgraph with the active node (*ship*) highlighted. Right: instantiated prompts for parent and leaf nodes used for semantic embedding generation.

Sigmoid) computes per-node, per-dimension fusion weights, allowing dynamic balancing between semantic priors and visual evidence.

Unified Prediction Head: After GNN refinement, node embeddings are mean-pooled and concatenated with the original visual features. A single linear layer maps this combined representation to logits for all hierarchy nodes simultaneously. The output is partitioned by level for loss computation.

A.3 Evaluation Strategy

We train on fixed splits (Table 4) and report metrics on the test set. For FGVC datasets (Oxford Pets-37, Stanford Cars, ETHEC), we use official train/test splits. For remote sensing and medical datasets, we use 80/20 train/test splits with iterative stratification to preserve label frequency distributions. A validation set (10% of training data) is used for early stopping and hyperparameter selection.

Unless stated otherwise, performance is reported on leaf labels only to enable fair comparison with flat baselines that predict only at the most specific level. However, MAPLE predicts at all hierarchy levels, and we report per-level performance in detailed analyses.

Few-Shot Settings: To study label scarcity effects, we conduct few-shot experiments with $K \in \{4, 8, 12, 16\}$ labeled examples per leaf category. For each shot configuration, we randomly sample K examples per leaf class from the training set, ensuring balanced representation. We perform three independent runs with different random seeds and report mean and standard deviation of AU$\overline{\text{PRC}}$ on the test set.

A.4 Computational Resources

All experiments were conducted on four NVIDIA A100 GPUs equipped with 40 GB memory each.

A.4.1 Evaluation Metrics

We employ the micro-averaged Area Under the Precision-Recall Curve (AU$\overline{\text{PRC}}$), a widely recognized evaluation metric to comprehensively assess method performance. AU$\overline{\text{PRC}}$ is specifically suited to multi-label classification, providing a global measure of performance across all classes. The metric is computed using micro-averaged precision ($\overline{\text{Prec}}$) and recall ($\overline{\text{Rec}}$), which are calculated as follows:

$$\overline{\text{Prec}} = \frac{\sum_i \text{TP}_i}{\sum_i \text{TP}_i + \sum_i \text{FP}_i}, \quad \overline{\text{Rec}} = \frac{\sum_i \text{TP}_i}{\sum_i \text{TP}_i + \sum_i \text{FN}_i}, \tag{3}$$

where TP_i, FP_i, and FN_i denote the true positives, false positives, and false negatives for class i, respectively. By varying the decision threshold, a precision-recall curve is generated, and the area beneath this curve provides a single scalar performance measure summarizing precision and recall trade-offs across classes. AU$\overline{\text{PRC}}$ is preferred over accuracy or F1-score for multi-label tasks because: (i) it is threshold-independent, avoiding arbitrary cutoff selection; (ii) it handles class imbalance well by focusing on positive predictions; and (iii) it provides a comprehensive view of precision-recall trade-offs across the operating range.

Table 5: Complete performance comparison across hierarchical levels using AU$\overline{\text{PRC}}$ (%). This table extends Table 1 from the main paper with results across all nine datasets and all hierarchical levels. l_i denotes the i-th hierarchical level. MAPLE* uses semantic initialization while MAPLE uses random initialization. Δ^* and Δ show relative improvements over flat baseline.

Dataset		**MAPLE***	**MAPLE**	**Flat Baseline**	**Δ^* (%)**	**Δ (%)**
		Remote Sensing Datasets				
AID	l_1	**95.31**	94.11	-	-	-
	l_2	**94.32**	93.10	-	-	-
	l_3	**84.83**	82.50	-	-	-
	Leaf	**87.25**	86.66	84.21	+3.61	+2.91
DFC-15	l_1	**99.62**	99.51	-	-	-
	l_2	98.33	**99.41**	-	-	-
	l_3	**98.37**	98.35	-	-	-
	Leaf	**98.71**	98.36	98.65	+0.06	-0.29
MLRSNet	l_1	**98.23**	98.21	-	-	-
	l_2	**97.88**	97.11	-	-	-
	l_3	**96.77**	96.50	-	-	-
	Leaf	**96.71**	96.31	96.17	+0.56	+0.15
		Medical Imaging Datasets				
MuRed	l_1	**78.69**	78.60	-	-	-
	l_2	73.24	**73.25**	-	-	-
	l_3	**49.00**	48.11	-	-	-
	Leaf	**55.04**	54.23	53.52	+2.84	+1.33
HPA	l_1	**79.08**	78.01	-	-	-
	Leaf	**51.18**	48.50	44.99	+13.76	+7.80
PadChest	l_1	**72.28**	72.21	-	-	-
	l_2	**38.16**	38.10	-	-	-
	l_3	15.29	**15.30**	-	-	-
	l_4	**9.57**	9.56	-	-	-
	l_5	**8.35**	8.10	-	-	-
	Leaf	**14.24**	13.11	11.68	+21.92	+12.24
		Fine-Grained Visual Categorization				
Pets	l_1	**99.85**	99.80	-	-	-
	Leaf	**93.62**	92.30	91.99	+1.77	+0.34
Cars	l_1	**97.89**	96.31	-	-	-
	Leaf	**93.14**	92.50	84.54	+10.17	+9.41
ETHEC	l_1	**99.74**	99.31	-	-	-
	l_2	**99.22**	99.10	-	-	-
	l_3	**96.55**	93.31	-	-	-
	Leaf	**86.89**	81.31	78.72	+10.38	+3.29

AU$\overline{\text{PRC}}$ is computed using the established implementation available in the `scikit-learn` library [22].

A.5 Broader Generalizability Assessment

Table 5 shows the complete performance comparison across all hierarchical levels for all nine datasets, extending the results presented in Table 1. MAPLE* (with semantic initialization) consistently outperforms both MAPLE (with random initialization) and flat baselines across remote sensing, medical imaging, and fine-grained visual categorization domains.

Figure 5 demonstrates the consistent performance advantages of MAPLE* over the flat MLC baseline across majority of datasets and shot configurations. The hierarchical approach shows particularly pronounced benefits in low-data regimes, with the performance gap being most substantial at $K = 4$ and $K = 8$ shots per category. This pattern indicates that taxonomic relationships among classes can compensate for limited direct supervision.

The magnitude of improvement varies significantly across domains and correlates with hierarchical complexity. Medical datasets exhibit the most substantial gains, with HPA and MuRed showing considerable performance differences that persist across all shot settings. Remote sensing datasets demonstrate more modest but consistent improvements, while FGVC benchmarks show intermediate gains with notable variation. Stanford Cars and ETHEC display strong hierarchical benefits that become more pronounced as shot count increases, suggesting that these fine-grained domains particularly benefit from structured learning even with moderate data availability.

Interestingly, some datasets like PadChest show complex performance patterns where the hierarchical advantage varies with shot count, potentially reflecting the interaction between taxonomic structure

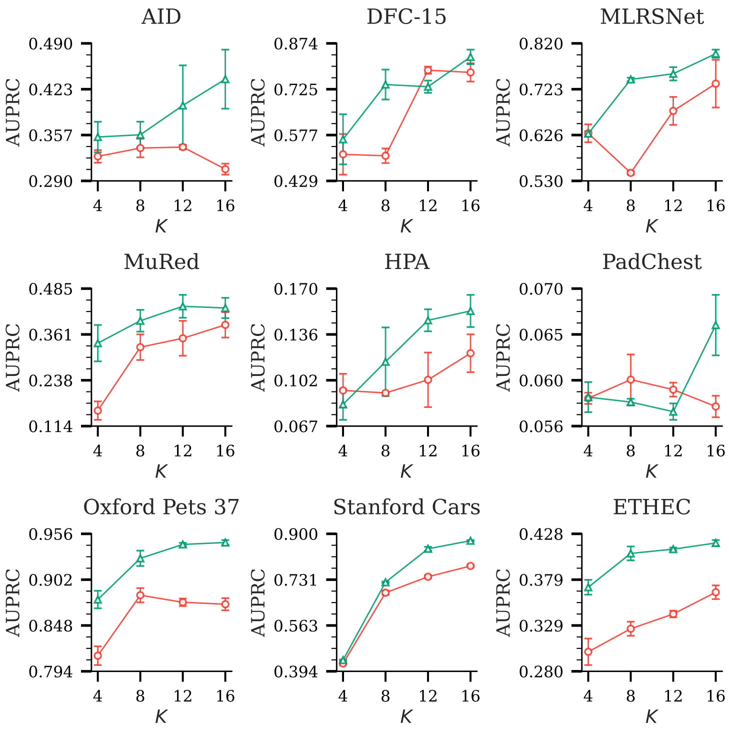


Figure 5: Few-shot learning performance comparison between MAPLE (in **green**) and flat MLC baseline (in **red**) across representative datasets. Results show AU$\overline{\text{PRC}}$ performance ($\mu \pm \sigma$ across three experimental repeats) for varying numbers of shots per category ($K \in \{4, 8, 12, 16\}$). MAPLE consistently outperforms the baseline, with particularly pronounced benefits in low-data regimes.

complexity and available training signal. The consistent upward trend for MAPLE across increasing shot counts demonstrates that hierarchical supervision scales effectively with data availability while maintaining its advantage over the flat MLC approach.

A.6 Computational Efficiency Analysis

Table 6 presents detailed computational overhead analysis. MAPLE adds only 0.7% additional GFLOPs and 2.6% more parameters compared to the flat baseline, with inference time overhead ranging from 2.4% to 5.3% across different batch sizes.

Table 6: Computational efficiency comparison on AID dataset.

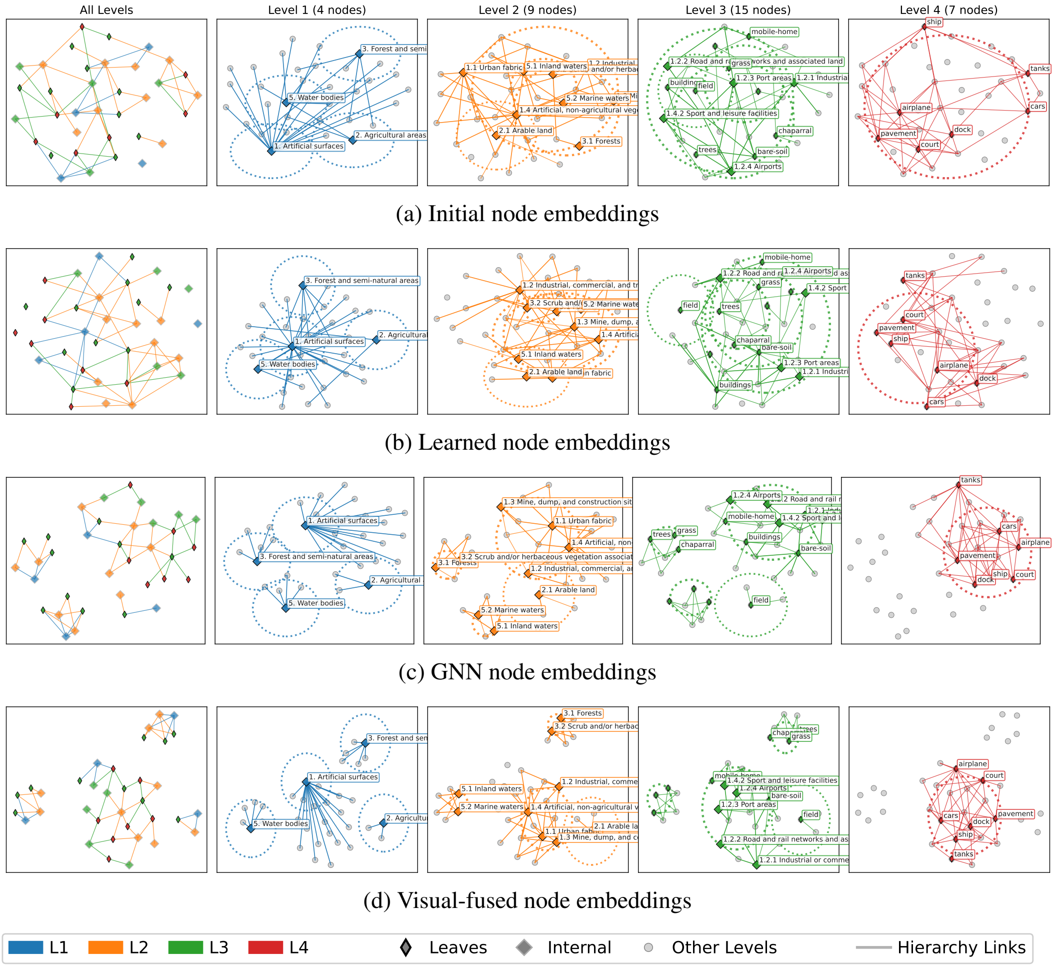
	Per Image (ms)					
Model	8	16	32	64	GFLOPs	Params (M)
Flat Baseline	3.02	2.96	2.62	2.55	33.54	86.57
MAPLE	3.18	3.01	2.65	2.61	33.77	88.84
Overhead (%)	+5.3	+1.7	+1.1	+2.4	+0.7	+2.6

A.7 Qualitative Results

This section presents a qualitative evaluation of the proposed method on three representative datasets: AID (remote sensing), Oxford Pets (fine-grained visual categorization), and MuRed (medical imaging). Unless stated otherwise, we use the semantically initialized variant (SentenceTransformer-based initialization) and refer to it simply as MAPLE throughout.

A.7.1 Embedding Evolution Analysis

For simplicity and clarity of presentation, we visualize the learned embeddings for the AID dataset as a representative example. Figure 6 demonstrates the progressive emergence of semantic structure across four key training stages. The initial embeddings (a) exhibit random spatial distribution with no semantic coherence. Following hierarchical semantic initialization (b), embeddings start to develop semantic organization where taxonomically related nodes tend to cluster together. Graph neural network refinement (c) further enhances clustering through iterative message passing, achieving improved separation with enhanced intra-cluster cohesion. The final multimodal fusion (d) produces the most sophisticated organization, where the adaptive gating mechanism integrates visual and semantic information.



(a) Initial node embeddings

(b) Learned node embeddings

(c) GNN node embeddings

(d) Visual-fused node embeddings

Figure 6: Evolution of node embeddings throughout MAPLE training stages visualized using UMAP dimensionality reduction on the AID dataset. (a) Initial embeddings show random spatial distribution. (b) Learned embeddings after hierarchical semantic initialization demonstrate clear semantic clustering. (c) GNN embeddings after graph neural network refinement exhibit enhanced separation between semantic groups. (d) Visual-fused embeddings after multimodal fusion achieve sophisticated organization, integrating both visual and semantic information while respecting hierarchical structure.

A.7.2 Error Analysis

To understand the practical benefits of hierarchical classification, we conduct a detailed analysis of leaf-level classification errors comparing MAPLE with the baseline MLC method. Table 7 quantifies these improvements across three representative datasets. MAPLE achieves consistent error reduction, with the most substantial improvement on Oxford Pets (70.7% reduction). This

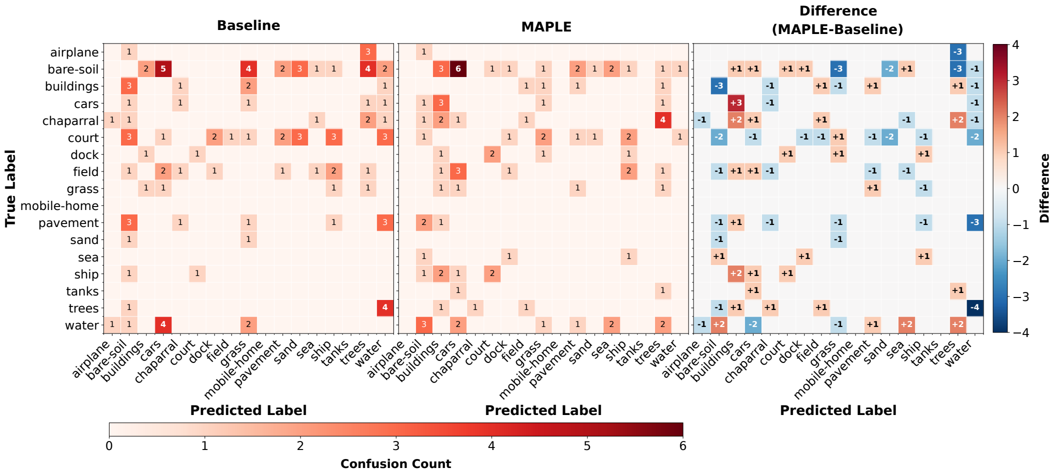


Figure 7: Leaf-level confusion matrix comparison for AID dataset. Left: baseline model confusions, Center: MAPLE confusions, Right: difference matrix (MAPLE − Baseline). Blue cells indicate reduced confusions (improvements), red cells indicate increased confusions. MAPLE achieves a 12.1% reduction in total leaf-level confusions, with notable improvements in semantically challenging pairs such as trees→water and buildings→bare-soil.

significant improvement on a fine-grained recognition task demonstrates MAPLE's particular strength in scenarios where hierarchical relationships between classes are semantically meaningful. The AID dataset shows a moderate but consistent improvement (12.1%), while MuRed exhibits minimal improvement (1.3%), likely due to its already low baseline confusion rate.

Table 7: Leaf-level confusion reduction across datasets. MAPLE consistently reduces confusion instances compared to baseline models, with particularly strong improvements on Oxford Pets (70.7% reduction) and AID (12.1% reduction). Lower values indicate better performance.

Dataset	Baseline	MAPLE	Improvement (%)	Absolute Reduction
AID	107	94	12.1	13
MuRed	77	76	1.3	1
Oxford Pets	41	12	70.7	29

For a detailed understanding of these improvements, we present a comprehensive confusion matrix analysis for the AID dataset as a representative example. Figure 7 shows confusion matrices of the baseline MLC and MAPLE, and their difference matrix. The difference matrix clearly illustrates where MAPLE reduces errors (blue cells) versus where new errors emerge (red cells).

The most significant improvements occur in semantically challenging confusion pairs. MAPLE substantially reduces problematic confusions such as trees-to-water and pavement-to-water misclassifications, suggesting that hierarchical representations help the model better distinguish between natural and artificial surface types. Similarly, reductions in structural versus natural category confusions indicate improved understanding of taxonomic relationships.

The error analysis reveals that MAPLE's hierarchical structure provides the most benefit when classes have clear taxonomic relationships, visual similarities can be resolved through higher-level semantic understanding, and sufficient hierarchical structure exists to guide the learning process. These findings validate our hypothesis that incorporating hierarchical knowledge leads to more robust and semantically coherent predictions at the leaf level.

A.8 Ablation Studies

A.8.1 Graph Encoder Analysis

We analyze the impact of different graph neural network architectures on MAPLE's performance by comparing three widely-used GNN variants: Graph Convolutional Networks (GCN)[35],

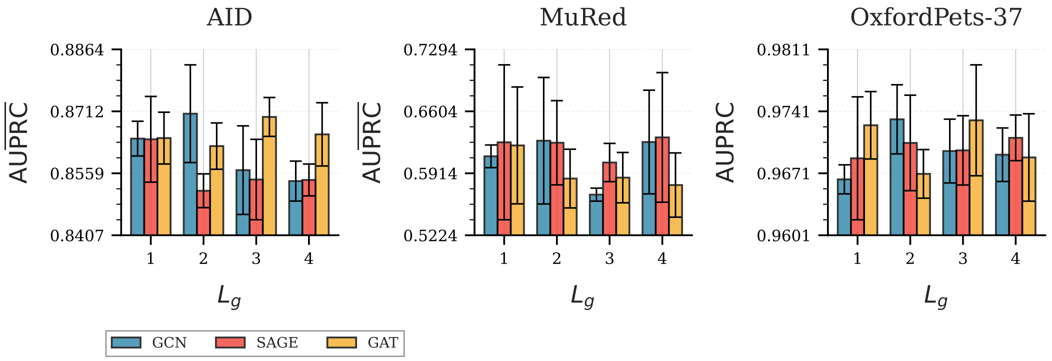


Figure 8: Performance comparison of different GNN architectures across datasets.

GraphSAGE[20], and Graph Attention Networks (GAT)[36]. Figure 8 shows the performance across different numbers of GNN layers ($L_g = 1, 2, 3, 4$) on three datasets.

The results demonstrate that all three GNN architectures achieve comparable performance, with minor variations across datasets. GCN shows consistent performance across layer depths, while GAT exhibits slight improvements with deeper architectures on AID and Oxford Pets-37. GraphSAGE performs competitively but shows more sensitivity to layer depth on certain datasets. Notably, the performance differences between architectures are relatively small (typically within 0.01 AUPRC), suggesting that the hierarchical message passing mechanism is more important than the specific aggregation strategy.

The optimal number of layers varies by dataset, with most configurations benefiting from 2-3 layers. Deeper networks ($L_g = 4$) sometimes show diminishing returns, indicating that the hierarchical structure can be effectively captured with moderate network depth.

A.8.2 Embedding Initialization Strategies

The initialization of node embeddings in hierarchical classification systems plays a crucial role in model convergence and final performance. While random initialization has been the standard approach in many graph-based models, recent advances in sentence embedding models offer opportunities to leverage semantic priors that can better capture the inherent relationships between class labels. Recent work [37] has shown that different LLM embeddings can improve deep learning architectures for training on downstream tasks.

We evaluate five initialization strategies across representative datasets from different domains: (1) Random initialization with normalized Gaussian vectors; (2) NV-Embed-v2 initialization using the state-of-the-art generalist embedding model[38]; (3) MPNet-Base-v2 initialization using the all-mpnet-base-v2 sentence transformer[34]; (4) Word2Vec initialization using pre-trained word vectors[39]; and (5) GloVe initialization with pre-trained embeddings[40]. For semantic methods, we enhance class names with hierarchical context by incorporating parent-child relationships from the taxonomy structure (See Appendix A.2). All experiments are repeated three times with different random seeds to ensure statistical reliability.

Figure 9 presents the comparative results across three representative datasets. Surprisingly, random initialization consistently achieves competitive or superior performance compared to sophisticated pre-trained embedding strategies. On AID, random initialization (0.8665C AUPRC) slightly outperforms all semantic alternatives, while on MuRed, it achieves comparable results to the best semantic method. Only on Oxford Pets-37 do semantic methods show marginal improvements, with Word2Vec achieving the highest performance (0.9683 AUPRC).

These results reveal several important insights about hierarchical visual classification. Random initialization provides maximum plasticity, allowing the model to learn task-specific visual relationships without semantic bias. The strong visual features extracted by the ViT backbone appear sufficient for discovering meaningful hierarchical relationships through graph-based propagation. In contrast, pre-trained semantic embeddings may impose text-corpus relationships that do not align with visual similarities, potentially constraining the model's ability to learn optimal visual hierarchies.

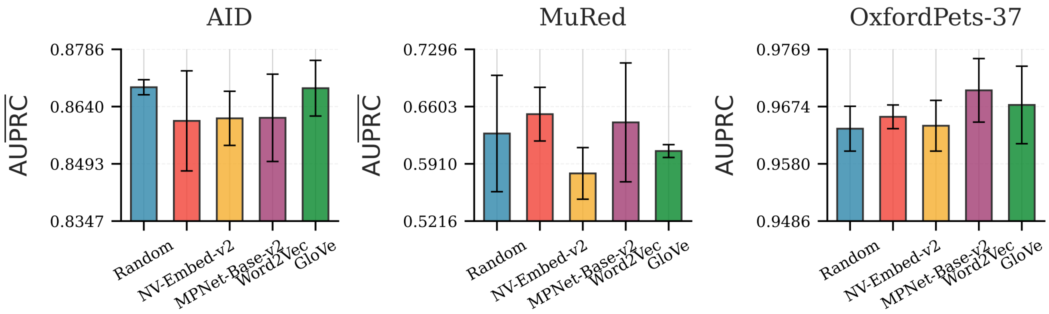


Figure 9: Comparison of embedding initialization strategies.

The modest performance differences between methods and relatively small standard deviations across repeats suggest that the GNN architecture effectively refines initial embeddings regardless of initialization strategy. This finding indicates that our hierarchical approach successfully learns visual taxonomies from data, with the graph-based refinement process being more influential than semantic priors for final performance.

A.9 Limitations

While MAPLE demonstrates strong performance across diverse HMLC tasks, several areas present opportunities for future improvement.

MAPLE's performance is influenced by the quality of the input hierarchical structure. Our hybrid construction approach combines expert-curated taxonomies (CORINE, ICD-10) with ChatGPT-generated hierarchies, which may occasionally introduce inconsistencies. Language model-generated taxonomies, while useful, may not always fully capture domain-specific relationships or could reflect biases present in training data, particularly in specialized domains such as medical imaging or remote sensing.

While MAPLE effectively models fine-grained distinctions between leaf categories, UMAP visualizations occasionally show that semantically related categories sharing common parent nodes may appear separated in the learned embedding space. This suggests there is room for improvement in fully exploiting higher-level taxonomic relationships across all hierarchy levels.

The benefits of hierarchical modeling vary across different domains. While medical imaging datasets consistently show substantial improvements, some remote sensing datasets (e.g., MLRSNet) exhibit more modest gains despite their large scale and complex structures. This variation indicates that the effectiveness of hierarchical modeling may depend on specific domain characteristics.

EurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Introduction describe MAPLE's components (hierarchical semantic initialization, graph-based structure encoding, adaptive fusion and level-aware objective) and claim consistent gains with small parameter overhead. These claims are supported by Tables 1–3, the few-shot results in Table 2, and analyses in Appendices A.5–A.8.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The Discussion and Appendix A.9 describe that gains narrow on very large datasets and that performance depends on the quality of the constructed hierarchy; we also note sensitivity to noisy mappings and oversmoothing with deeper GNNs. Computational overheads and latency are quantified in Appendix A.6 with details in Table 6.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: The paper is empirical and methodological; it does not introduce formal theorems or proofs.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Datasets, hierarchy construction, model variants, training setup, metrics, and evaluation protocols are detailed in Appendices A.1–A.2 and A.3. We disclose hardware and compute in Appendix A.4 and efficiency in Appendix A.6. The exact hierarchical label structures will be released as YAML configuration files with the code.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: All datasets are public and cited, but an anonymized code repository is not included in the submission. We will release code, YAML hierarchies, and run scripts with detailed instructions after the review period, preserving anonymity at submission time.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Implementation details, augmentations, optimizer, schedules, batch sizes, epochs, and split strategies are specified in Appendix A.2 and A.3; dataset statistics appear in Table 4.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We report averages over three runs for all settings and show mean±std in the learning curves (Fig. 5) and in the few-shot table (Table 2). For brevity, other tables report means only; variability is visible in the curves.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Hardware is specified in Appendix A.4 (4×A100 40 GB). Appendix A.6 provides inference-time, GFLOPs, and parameter overhead (Table 6), with consistent scaling across batch sizes.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We evaluate on public datasets with appropriate citations, do not process personally identifiable information, and adhere to standard research practices and anonymization.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: While applications and potential benefits for environmental monitoring, urban planning, and medical analysis are discussed in the Introduction and Discussion, a dedicated broader impacts section with explicit negative impact analysis is not included.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The work does not release high-risk models or scraped web datasets; we train task models on curated public benchmarks.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [No]

    Justification: We cite all datasets and prior methods, but specific license names are not listed in the current version. We will include license details for each asset in the camera-ready version.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: No new datasets are introduced at submission time. Configuration files (YAML hierarchies) and trained weights will be documented and released with the code after the review period.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The research does not involve crowdsourcing or human subjects.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The research does not involve human subjects.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: We used ChatGPT for limited assistance in hierarchy mapping when direct alignment to established taxonomies was ambiguous, followed by manual verification (Appendix A.1). We also used it for light paraphrasing to improve clarity. LLMs were not used to design methods, run experiments, or interpret results.