Supplementary: Locality-Aware Hyperspectral Classification

Fangqin Zhou
f.zhou@tue.nl
Mert Kilickaya
kilickayamert@gmail.com
Joaquin Vanschoren
j.vanschoren@tue.nl

Automated Machine Learning
Eindhoven University of Technology
Eindhoven, Netherlands

1 Experimental Setup

Implementation Details. The proposed **Hy**perspectral **L**ocality-aware **I**mage **T**ransform**E**r is implemented with PyTorch. We adopt the backbone code from Hong et al. [7] and implement additional local attention modules and the regularization objective function. The experiments were run on NVIDIA RTX A6000 GPU. Our work strictly follows the experimental setup of Hong *et al.* [7], with identical train-test splits, hyper-parameters, datasets and metrics. The hyper-parameters are listed in Table 4. We provide the details below. **CAF Layer**: To preserve the information across all layers, following [7], we apply Cross-Adaptive Fusion (CAF) between the output of the $(b-2)$-th block and the output of the (b)-th block. The representations of these two blocks are fused with a 1×2 convolution layer. We refer to Hong et al. [7] for more details.

Datasets. We evaluate our model on three standard, public benchmarks. *i)* ***Indian Pines*** *[3]*: The dataset is collected by AVIRIS sensor over the Indian Pines test site in North-Western Indiana, USA. The dataset consists of 224 spectral bands, sampled between $400-2500$ nm, with a spatial resolution of 145×145 pixels. The dataset includes 16 distinct categories, 695 training and 9671 testing images. Typical categories include $\{corn, woods, wheat\}$. *ii)* ***Houston2013*** *[7]*: The dataset is collected by ITRES CASI-1500 sensor over the University of Houston campus. It contains 144 spectral bands sampled between $380-1050$nm, with a spatial resolution of 349×1905 pixels. The dataset includes 15 distinct categories, 2832 training and 12197 testing images. Typical categories include $\{water, soil, tree\}$. *iii)* ***Pavia University*** *[3]*: The dataset is collected by ROSIS sensor over Pavia University, Italy. It contains 103 spectral bands sampled between $430-860$nm, with a spatial resolution of 610×340 pixels. The dataset includes 9 distinct categories, 3921 training and 40002 testing images. Typical categories include $\{asphalt, meadows, bricks\}$.

Baselines. We compare our model to several state-of-the-art networks. The *K-nearest neighbor* model makes predictions by calculating the pairwise Euclidean distance of spectral bands between training pixels [10]. The *random forest* classifier incorporates bagging of training

Class No.	Class Name	Training	Testing
1	Healthy Grass	198	1053
2	Stressed Grass	190	1064
3	Synthetic Grass	192	505
4	Tree	188	1056
5	Soil	186	1056
6	Water	182	143
7	Residential	196	1072
8	Commercial	191	1053
9	Road	193	1059
10	Highway	191	1036
11	Railway	181	1054
12	Parking Lot1	192	1041
13	Parking Lot2	184	285
14	Tennis Court	181	247
15	Running Track	187	473
	Total	2832	12197

Table 1: Land-cover classes of Houston2013 dataset, together with the training and testing samples for each class.

Class No.	Class Name	Training	Testing
1	Asphalt	548	6304
2	Meadows	540	18146
3	Gravel	392	1815
4	Trees	524	2912
5	Metal Sheets	265	1113
6	Bare Soil	532	4572
7	Bitumen	375	981
8	Bricks	514	3364
9	Shadows	231	795
	Total	3921	40002

Table 2: Land-cover classes of Pavia University dataset, together with the training and testing samples for each class.

pixels and random subspace feature selection of spectral bands [4]. The *support vector machine* firstly projects spectral bands into high-dimensional feature space and maximizes the geometrical margin between categories [11]. The *1-D CNN* model performs 1-D convolution along the spectral dimension [8], while the *2-D CNN* model firstly patchifies the input image and then performs 2-D convolution along the spatial dimension [2]. The *RNN* model processes spectral bands as time sequences with a stack of recurrent layers and gated recurrent units [5]. The *miniGCN* network firstly generates an adjacency metric with a KNN-based graph and then processes the generated graph via graph convolution layers [6]. The *ViT*-based network uses ViT encoder blocks to process spectral bands as a sequence of signals [7]. Different from pure ViT, the *SpectralFormer* patchifies the input image and groups neighboring spectral bands to a sequence of input vectors, and then processes the input via a stack of ViT encoder blocks [7]. Finally, *MAEST* pre-trains a feature extractor via a masked encoder-decoder reconstructing network and then fine-tunes the pre-trained encoder with labeled data [9]. Since the proposed system model is built upon SpectralFormer, we did not re-run all the baseline models except for the SpectralFormer and MAEST. Hence, we refer to Hong et al. [7] for more experimental details of those comparison baselines.

2 Hyper-Parameter Tuning of λ

According to our objective equation of the proposed HyLITE, to find a suitable λ that balances well between cross-entropy loss and local-spectral regularization loss, we evaluate HyLITE with 6 different λ values. The results are listed in Table 5, which shows that simply selecting $\lambda = 1$ works the best across most accuracy metrics.

3 Addition Category-level comparisons

The results of category-level comparison with SpectralFormer on Houston2013 and Pavia University datasets are presented in Figure 1 and Figure 2, respectively.

Class No.	Class Name	Training	Testing
1	Corn Notill	50	1384
2	Corn Mintill	50	784
3	Corn	50	184
4	Grass Pasture	50	447
5	Grass Trees	50	697
6	Hay Windrowed	50	439
7	Soybean Notill	50	918
8	Soybean Mintill	50	2418
9	Soybean Clean	50	564
10	Wheat	50	162
11	Woods	50	1244
12	Buildings Grass Trees Drives	50	330
13	Stone Steel Towers	50	45
14	Alfalfa	15	39
15	Grass Pasture Mowed	15	11
16	Oats	15	5
	Total	695	9671

Table 3: Land-cover classes of Indian Pines dataset, together with the training and testing samples for each class.

Hyperparameter	Value
Number of transformer blocks	5
Number of attention heads	4
Embedding dimension	64
Dimension head	16
MLP dimension	8
Optimizer	Adam (weight decay=5e-3)
Initial learning rate	5e-4
Learning rate scheduler	StepLR (gamma=0.9)
Batch size	32
Total training epochs	300

Table 4: Hyperparameters of the architecture and training. Notably, the dimension head denotes the scaling factor of the Query-Key dot product. It usually has the same value as the embedding dimension, but we follow the setting of SpectralFormer using a varied value.

	Indian Pines			Houston2013			Pavia University		
λ	OA	AA	Kappa	OA	AA	Kappa	OA	AA	Kappa
0.1	86.96	91.68	0.85	84.27	86.06	0.83	83.25	89.63	0.79
0.5	88.10	92.40	0.86	85.48	87.29	0.84	90.19	92.37	0.87
1	89.80	94.69	0.88	88.49	89.74	0.86	91.28	92.25	0.88
2	87.24	91.74	0.85	88.09	88.94	0.87	91.79	89.55	0.89
5	87.01	93.28	0.85	85.18	86.67	0.84	81.70	90.65	0.77
10	84.33	89.82	0.82	87.82	88.73	0.87	87.54	92.25	0.84

Table 5: Hyperparameter tuning the magnitude of the regularization loss (λ). Simply selecting $\lambda = 1$ works best across most accuracy metrics.

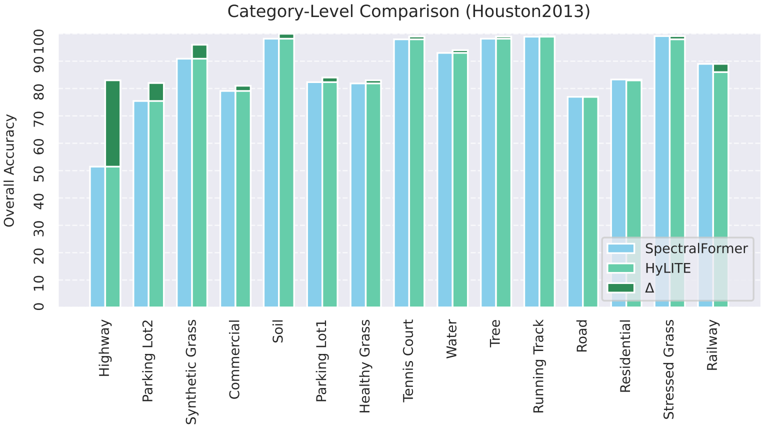

Figure 1: Category-level comparison to the SpectralFormer on Houston2013. The contribution of HyLITE is generic, with fine-grained categories of 'Highway', 'Parking Lot2', and 'Synthetic Grass' receiving the highest benefits.

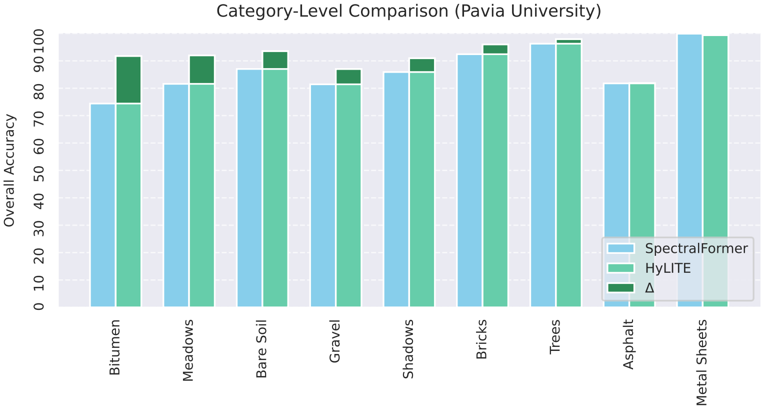

Figure 2: Category-level comparison to the SpectralFormer on Pavia University. The contribution of HyLITE is generic, with fine-grained categories of 'Bitumen', 'Meadows', and 'Bare Soil' receiving the highest benefits.

References

[1] 2013 IEEE GRSS Data Fusion Contest – Fusion of Hyperspectral and LiDAR Data. URL https://hyperspectral.ee.uh.edu/?page_id=459.

[2] Yushi Chen, Hanlu Jiang, Chunyang Li, Xiuping Jia, and Pedram Ghamisi. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 54(10):6232–6251, 2016.

[3] M Graña, MA Veganzons, and B Ayerdi. Hyperspectral remote sensing scenes. *Hyperspectral Remote Sensing Scenes - Grupo de Inteligencia Computacional (GIC)*. URL https://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes.

[4] Jisoo Ham, Yangchi Chen, Melba M Crawford, and Joydeep Ghosh. Investigation of the random forest framework for classification of hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing*, 43(3):492–501, 2005.

[5] Renlong Hang, Qingshan Liu, Danfeng Hong, and Pedram Ghamisi. Cascaded recurrent neural networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(8):5384–5394, 2019.

[6] Danfeng Hong, Lianru Gao, Jing Yao, Bing Zhang, Antonio Plaza, and Jocelyn Chanussot. Graph convolutional networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(7):5966–5978, 2020.

[7] Danfeng Hong, Zhu Han, Jing Yao, Lianru Gao, Bing Zhang, Antonio Plaza, and Jocelyn Chanussot. Spectralformer: Rethinking hyperspectral image classification with transformers. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2021.

[8] Wei Hu, Yangyu Huang, Li Wei, Fan Zhang, and Hengchao Li. Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors*, 2015:1–12, 2015.

[9] Damian Ibanez, Ruben Fernandez-Beltran, Filiberto Pla, and Naoto Yokoya. Masked auto-encoding spectral–spatial transformer for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2022.

[10] Li Ma, Melba M. Crawford, and Jinwen Tian. Local manifold learning-based k -nearest-neighbor for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 48(11):4099–4109, 2010. doi: 10.1109/TGRS.2010.2055876.

[11] Farid Melgani and Lorenzo Bruzzone. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on geoscience and remote sensing*, 42(8):1778–1790, 2004.