Spectral-Spatial Transformer with Active Transfer Learning for Hyperspectral Image Classification

Muhammad Ahmad, Manuel Mazzara, Salvatore Distefano,

Abstract—The classification of hyperspectral images (HSI) is a challenging task due to the high spectral dimensionality and limited labeled data typically available for training. In this study, we propose a novel multi-stage active transfer learning (ATL) framework that integrates a Spatial-Spectral Transformer (SST) with an active learning process for efficient HSI classification. Our approach leverages a pre-trained (initially trained) SST model, fine-tuned iteratively on newly acquired labeled samples using an uncertainty-diversity (Spatial-Spectral Neighborhood Diversity) querying mechanism. This mechanism identifies the most informative and diverse samples, thereby optimizing the transfer learning process to reduce both labeling costs and model uncertainty. We further introduce a dynamic freezing strategy, selectively freezing layers of the SST model to minimize computational overhead while maintaining adaptability to spectral variations in new data. One of the key innovations in our work is the self-calibration of spectral and spatial attention weights, achieved through uncertainty-guided active learning. This not only enhances the model's robustness in handling dynamic and disjoint spectral profiles but also improves generalization across multiple HSI datasets. Additionally, we present a diversity-promoting sampling strategy that ensures the selected samples span distinct spectral regions, preventing overfitting to particular spectral classes. Experiments on benchmark HSI datasets demonstrate that the SST-ATL framework significantly outperforms existing CNN and SST-based methods, offering superior accuracy, efficiency, and computational performance. The source code can be accessed at https://github.com/mahmad000/ATL-SST.

Index Terms—Hyperspectral Image Classification; Active Learning; Hybrid Query Function; Transfer Learning; Spatial-Spectral Transformer.

I. INTRODUCTION

HYPERSPECTRAL imaging (HSI) is a powerful remote sensing technique that captures detailed spectral information across narrow wavelength bands, enabling precise identification and classification of diverse land-cover materials [1], [2]. Due to its high spectral resolution, HSI is extensively applied in various fields [3] including remote sensing [4], [5], environmental monitoring [6], food processing [7], red chili adulteration [8], [9], minced meat adulteration [10], [11], forensic [12], biomedical [13], [14], and many more. However, inherent challenges, including high dimensionality, limited labeled samples, and complex spectral-spatial variability, complicate the classification process [15]–[17].

Recently, transformer-based architectures have gained prominence in the realm of HSI classification [18], [19].

M. Ahmad and S. Distefano are with Dipartimento di Matematica e Informatica-MIFT, University of Messina, 98121 Messina, Italy. (e-mail: mahmad00@gmail.com; sdistefano@unime.it

M. Mazzara is with the Institute of Software Development and Engineering, Innopolis University, Innopolis, 420500, Russia. (e-mail: m.mazzara@innopolis.ru)

These architectures leverage self-attention mechanisms that effectively capture long-range dependencies across both spectral and spatial dimensions, significantly enhancing feature representation [20]–[22]. Spatial-spectral transformers (SSTs) have emerged as a powerful alternative to convolutional neural networks (CNNs), focusing on modeling interactions between spectral bands and spatial contexts across patches [23]–[25]. SSTs exhibit excellent scalability when applied to high-resolution HSIs, efficiently handling large HSI datasets without the necessity for complex pooling operations. This capability has facilitated their successful use in HSI classification, with their flexible architecture contributing to wide-ranging applications. Additionally, SSTs minimize the dependence on manually engineered features by learning hierarchical representations directly from raw pixel data [26], streamlining model development and often leading to enhanced performance. The attention maps produced by SSTs also improve interpretability by identifying the most influential image regions in the model's predictions, thereby offering valuable insights into the decision-making process [27].

Several studies have explored transformer-based architectures for HSI classification. For example, Shi et al. [28] developed a dual-branch transformer network that effectively captures multi-scale spatial-spectral dependencies by separately processing spectral and spatial tokens. Lan et al. [29] introduced a hybrid-spatial transformer combining convolutional layers for better feature extraction. While these approaches show the effectiveness of transformers, challenges remain [30]. Training large SSTs is computationally demanding due to the quadratic complexity of self-attention, affecting scalability with longer sequences [31]. SSTs may also lack translation invariance compared to CNNs, impacting consistent spatial relationship capture [32]. The tokenization of input images into fixed-size patches can miss fine-grained details, and SSTs often need substantial training data, risking overfitting with smaller datasets [33]. Acquiring labeled HSI data is costly and time-consuming, limiting the feasibility of fully supervised methods [34], [35]. Thus, semi-supervised or active learning (AL) strategies that utilize both labeled and unlabeled data are essential for enhancing classification performance while reducing dependence on annotated samples [36], [37].

AL aims to address the challenges of limited labeled data by selectively querying the most informative samples from a pool of data [38], thereby reducing annotation efforts while maximizing model performance [39], [40]. AL has been effectively utilized in HSI classification to minimize annotation costs. For example, Ahmad et al. [41] implemented fuzziness-based sampling to iteratively update training sets, demonstrating improved performance with fewer labeled samples.

Cho et al. [42] proposed an active deep learning framework for HSI classification, integrating CNN with AL to reduce labeling costs and enhance accuracy, with a Markov random field (MRF) refining classification by enforcing label smoothness across classes. Liu et al. [43] introduced MDL4OW, a multitask HSI classification that addresses unknown classes by performing simultaneous classification and reconstruction, identifying unknown classes through reconstruction errors.

Liao et al. [44] presented CGE-AL, a class-wise graph-embedding-based AL framework that utilizes a class-wise graph convolutional network (CGCN) to optimize sample selection for labeling in HSI classification. By measuring uncertainty through graph parameters, CGE-AL iteratively queries and labels the most informative samples, thereby enhancing classification performance with minimal manual annotation. Zhao et al. [45] introduced MAT-ASSAL, an HSI classification framework that combines a multi-attention transformer (MAT) with adaptive superpixel segmentation-based AL to enhance classification performance. MAT captures long-range dependencies and fine-grained local features, while adaptive superpixel segmentation optimizes sample selection for training, preserving essential spatial details. Wang et al. [46] proposed an AL approach that enhances deep classification performance by integrating a co-auxiliary learning strategy and multi-level diversity selection, where an auxiliary network aids in pseudo-labeling and reduces redundancy while ensuring the selection of both representative and uncertain samples. Liu et al. [47] developed an adversarial domain alignment framework with contrastive learning for HSI classification, which enhances cross-domain feature consistency for effective knowledge transfer. By combining spectral and semantic alignment with multiscale feature selection, this method improves robustness and adaptability in few-shot scenarios using jointly optimized adversarial and contrastive losses.

Despite the advancements, deep AL often struggles with the challenges posed by limited labeled samples, which constrain the diversity and quality of selected data, ultimately leading to suboptimal model generalization [48]. Additionally, HSIs present unique challenges due to their intricate spatial-spectral dependencies, which traditional AL methods may not fully capture [49]. These limitations underscore the necessity for an active transfer learning (ATL) framework, wherein an SST can leverage knowledge from labeled source domains to enhance sample selection and feature learning in low-labeled target domains. The integration of AL with transfer learning provides an innovative approach to these challenges, allowing models trained on a source domain to adapt efficiently to a target domain with minimal retraining. This methodology is particularly advantageous for dynamic remote sensing tasks, where the spectral characteristics of new images may differ significantly from those of the training set. By combining SST with ATL, the model can selectively query the most uncertain and diverse samples, iteratively updating the training set and thereby improving the model's generalization capabilities in the target domain.

In this study, we propose an ATL framework based on an SST (SST-ATL) specifically designed for HSI classification. This framework directly addresses the core challenges of HSI classification, including limited labeled samples, high spectral-spatial variance, and dynamic environmental conditions. It achieves this by utilizing a pre-trained SST model with frozen layers and selectively querying the most valuable samples for annotation. Through this transfer learning paradigm, the pre-trained SST model is fine-tuned on new datasets using AL, allowing the model to adapt to new data distributions while maintaining computational efficiency by querying only the most informative samples. Furthermore, the SST's self-attention mechanism adeptly captures the complex spatial-spectral dependencies inherent in HSIs, leading to enhanced classification accuracy with minimal annotation effort. The main contributions of this work are as follows:

1) **Multi-stage ATL Framework with SST:** We introduce a multi-stage ATL framework that integrates SST with an AL process optimized for HSI classification. This framework adopts a pre-trained SST (initially trained and freezer) model in a dynamic setting, where the model is iteratively fine-tuned on newly acquired labeled samples. The novelty lies in the multi-stage adaptation, which not only improves efficiency by minimizing retraining cost but also leverages spectral-spatial context during each stage to effectively handle spectral variability.

2) **Self-Calibration of Spectral and Spatial Attention Layers Through Uncertainty-Guided Querying:** A key contribution is the uncertainty-guided querying mechanism used to adaptively calibrate the SST's attention layers. By identifying uncertain samples and incorporating them into the training process, we propose a mechanism that self-calibrates the spectral and spatial attention weights. This self-calibration improves model robustness in environments with dynamic spectral profiles and addresses key limitations in existing SST-based models that struggle to generalize.

3) **Diversity-Promoting Sampling Strategy:** We propose a novel diversity-promoting (Spatial-Spectral Neighborhood Diversity) sampling strategy that complements uncertainty-based querying in the ATL loop. Unlike existing AL methods that predominantly focus on uncertainty, our approach ensures that selected samples span diverse spectral-spatial regions, thus preventing overfitting to any particular spectral class. This diversity-driven selection is critical for HSI datasets, where different land-cover types may exhibit highly correlated spectral signatures, leading to ambiguous classification boundaries.

4) **Dynamic Freezing of SST Layers for Efficient Transfer Learning:** We introduce a dynamic layer freezing technique for efficient transfer learning, wherein layers of the SST are selectively frozen and unfrozen based on the incoming data. This method contrasts with static freezing techniques used in transfer learning and achieves a balance between preserving learned spectral-spatial representations and adapting to new spectral variations. This contribution significantly reduces computational overhead, enabling real-time adaptation in
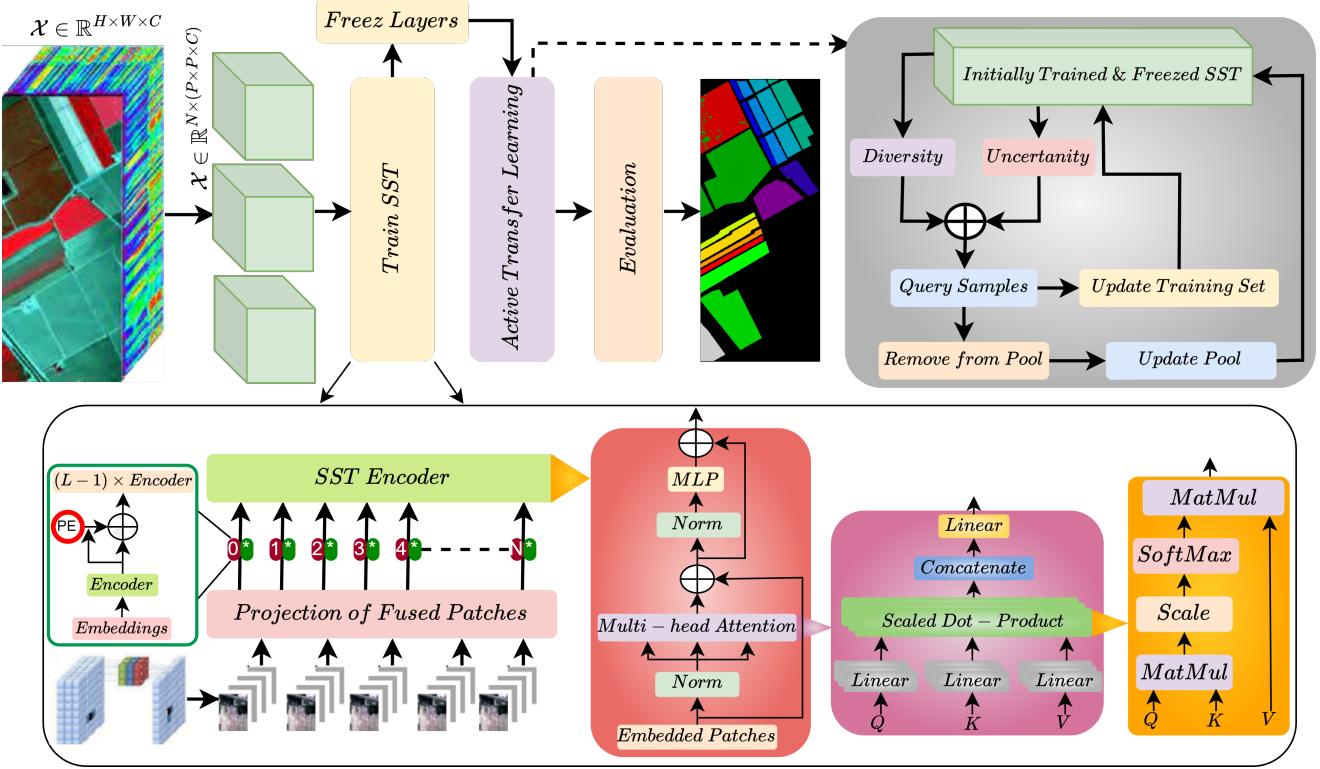
Fig. 1: The workflow of SST with Active Transfer Learning framework for HSI classification. In the top-left, an HSI is processed, followed by patch generation and initial SST training with frozen layers. The active transfer learning loop (top-right) leverages both diversity and uncertainty-based query sampling to iteratively update the training set and pool while reusing the pre-trained SST to minimize computational overhead. The bottom section details the SST encoder, where fused patches are projected through multiple encoding layers. Each patch undergoes multi-head attention, scaled dot-product, and normalization processes. The learned embeddings are enhanced through concatenation and feedforward layers (MLP). This integrated approach aims to refine classification performance with minimal annotation effort.

resource-constrained environments.

5) **Theoretical Formulation and Comprehensive Empirical Validation:** We present a rigorous theoretical formulation of the uncertainty-diversity-driven query mechanism and derive bounds for its effectiveness in ATL scenarios. Additionally, we validate our framework on multiple real-world HSI datasets with different spectral characteristics and spatial resolutions, providing a thorough comparison against state-of-the-art methods, including baseline CNNs, and SSTs. Our results demonstrate significant improvements in classification accuracy, robustness, and computational efficiency.

II. PROPOSED METHODOLOGY

Given an HSI $\mathbf{X} \in \mathbb{R}^{M \times N \times k}$, where k denotes the number of spectral bands, M is the image height, and N is the width, the model initiates by reshaping $\mathbf{X}$ into a sequence of patches. This is done by dividing $\mathbf{X}$ into overlapping patches of size $W \times W$.

A. Spectral-Spatial Transformer (SST)

The SST leverages the self-attention mechanism within a transformer architecture to capture spectral and spatial dependencies in HSI. The model has three main components: patch embedding, positional encoding, and the transformer encoder block. The architecture can be formulated as follows.

1) Patch Embedding: Given an input HSI $\mathbf{X} \in \mathbb{R}^{W \times W \times k}$, where W is the spatial window size (patch) and k is the number of spectral bands, we divide the image into patches. The patch embedding operation extracts meaningful features from local regions, producing a set of patches for subsequent processing. Specifically, we use a 3D convolutional layer to extract $p \times p$ patches:

$$\mathbf{P}_i = \text{Conv3D}(\mathbf{X}, \mathbf{W}_e), \quad i = 1, \dots, N_p \quad (1)$$

where $\mathbf{W}_e$ are learnable weights of the embedding layer, and N_p is the number of patches. Each patch $\mathbf{P}_i$ is of dimension $\mathbb{R}^{p \times p \times k}$, and the total number of patches is given by $N_p = \left(\frac{W}{p}\right)^2$. The embedding dimension is denoted by d.

2) Positional Encoding: To incorporate information about the relative positions of the patches, we add a positional encoding $\mathbf{E}_{\text{pos}}$ to the patch embeddings. The positional encoding is computed as follows:

$$\mathbf{E}_{\text{pos}}(i, 2j) = \sin\left(\frac{i}{10000^{\frac{2j}{d}}}\right), \quad (2)$$

$$\mathbf{E}_{\text{pos}}(i, 2j+1) = \cos\left(\frac{i}{10000^{\frac{2j}{d}}}\right) \quad (3)$$

where i is the position, j is the index of the dimension, and d is the embedding dimension. The resulting positional encoding is added to the patch embeddings:

$$\mathbf{Z} = \mathbf{P} + \mathbf{E}_{\text{pos}} \quad (4)$$

3) Transformer Encoder Block: Each transformer encoder block consists of a multi-head self-attention mechanism followed by a feedforward network (FFN). For each input patch embedding $\mathbf{Z}_i$, the self-attention mechanism computes attention scores and aggregates information from all patches. The multi-head attention is defined as:

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)\mathbf{W}_o \quad (5)$$

where the queries, keys, and values are computed as:

$$\mathbf{Q}_i = \mathbf{Z}_i\mathbf{W}_Q, \quad \mathbf{K}_i = \mathbf{Z}_i\mathbf{W}_K, \quad \mathbf{V}_i = \mathbf{Z}_i\mathbf{W}_V \quad (6)$$

The attention scores are calculated using the scaled dot-product attention:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \quad (7)$$

where d_k is the dimension of the keys and $\mathbf{W}_Q$, $\mathbf{W}_K$, $\mathbf{W}_V$ are learnable weight matrices for the queries, keys, and values, respectively. The output of the multi-head attention layer is passed through a feedforward network consisting of two dense layers with a ReLU activation in between:

$$\mathbf{F}(\mathbf{Z}_i) = \max(0, \mathbf{Z}_i\mathbf{W}_1 + b_1)\mathbf{W}_2 + b_2 \quad (8)$$

The final output of the transformer encoder layer is given by:

$$\mathbf{Z}' = \text{LayerNorm}(\mathbf{Z} + \text{Dropout}(\text{Attention}(\mathbf{Z}))) \quad (9)$$

$$\mathbf{Z}'' = \text{LayerNorm}(\mathbf{Z}' + \text{Dropout}(\mathbf{F}(\mathbf{Z}'))) \quad (10)$$

where LayerNorm represents layer normalization and Dropout is applied for regularization.

4) Cross-Attention Mechanism: The SST model integrates a cross-attention mechanism to enhance feature representation across patches. Let $\mathbf{Q}_c \in \mathbb{R}^d$ be a learnable query token and $\mathbf{K}_c, \mathbf{V}_c$ be the keys and values derived from the patch embeddings. The cross-attention is computed as:

$$\mathbf{A}_{\text{cross}} = \text{Softmax}\left(\frac{\mathbf{Q}_c\mathbf{K}_c^T}{\sqrt{d}}\right)\mathbf{V}_c \quad (11)$$

This allows the model to incorporate global context from all patches while maintaining spatial-spectral coherence.

5) Classification Head: The output of the transformer encoder and cross-attention mechanisms are concatenated, followed by two fully connected layers with ReLU activations:

$$\mathbf{O} = \text{ReLU}(\mathbf{Z}''\mathbf{W}_3 + b_3) \quad (12)$$

$$\mathbf{O}_{\text{final}} = \text{Softmax}(\mathbf{O}\mathbf{W}_4 + b_4) \quad (13)$$

where $\mathbf{O}_{\text{final}} \in \mathbb{R}^C$ is the final class prediction, and C is the number of target classes.

B. Active Learning

We employ an active learning technique to iteratively query the most informative samples from a pool of data. The following query strategies are applied:

1) Hybrid Query Strategy: The hybrid query strategy combines uncertainty and diversity sampling. Let $\mathcal{U}$ and $\mathcal{D}$ denote the set of samples selected by uncertainty and diversity sampling, respectively. The final query set $\mathcal{Q}$ is formed as:

$$\mathcal{Q} = \mathcal{U}(\mathbf{x}_i) \cup \mathcal{D}(\mathbf{x}_i) \quad (14)$$

where $\mathcal{Q}$ ensures that both uncertain and diverse samples are included in the labeled set. The hybrid strategy combines uncertainty and diversity sampling to balance exploration and exploitation in the active learning setting. Uncertainty sampling selects samples for which the model is least confident. For a sample $\mathbf{x}_i$, the uncertainty score is calculated as:

$$\mathcal{U}(\mathbf{x}_i) = -\max(p(y|\mathbf{x}_i)) \quad (15)$$

where $p(y|\mathbf{x}_i)$ is the predicted probability of the class for $\mathbf{x}_i$. Samples with the highest uncertainty scores are prioritized for labeling. Spatial-spectral neighborhood diversity sampling ensures that the selected pixels from the HSI are representative of diverse spatial-spectral characteristics within their local neighborhoods. This approach computes diversity based on the average pairwise spectral distance among neighboring pixels. Mathematically, this can be framed as:

$$\mathcal{D}(\mathbf{x}_i) = \underset{S \subseteq X_{\text{pool}}, |S| = \text{query\_size}}{\arg\max} \text{Diversity}(S) \quad (16)$$

where the diversity of a set S of pixels is defined as:

$$\text{Diversity}(S) = \frac{1}{m(m-1)} \sum_{j=1}^{m} \sum_{k=1, k \neq j}^{m} d_{jk} \quad (17)$$

where d_{jk} is the pairwise Euclidean distance between the spectral features of pixels j and k within their respective neighborhoods, and m is the number of pixels in the local neighborhood defined as:

$$m = (n_{\text{neighborhood}})^2 \quad (18)$$

The overall diversity for each pixel is computed as:

$$\text{diversity\_metric}(h, w) = \frac{1}{m(m-1)} \sum_{j=1}^{m} \sum_{k=1, k \neq j}^{m} d_{jk} \quad (19)$$

where (h, w) represents the coordinates of the pixel in the HSI. The query indices of the most diverse pixels are selected based on:

$$\text{query\_indices} = \text{argsort}(\text{diversity\_values})[-\text{query\_size} :] \tag{20}$$

This ensures that the final selection of pixels captures the spatial-spectral diversity, enhancing the effectiveness of AL strategies.

C. Model Training and Evaluation

The SST model is trained using a categorical cross-entropy loss function:

$$\mathcal{L} = -\sum_{i=1}^{N}\sum_{c=1}^{C} y_{ic} \log(\hat{y}_{ic}) \tag{21}$$

where y_{ic} is the ground truth label and $\hat{y}_{ic}$ is the predicted probability for class c for the i-th sample. The model is optimized using Adam with an initial learning rate of 1×10^{-4}. We evaluate the model on various metrics including overall accuracy (OA), average accuracy (AA), per-class accuracy (PA), and the kappa coefficient (κ). The kappa coefficient is calculated as:

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{22}$$

where p_o is the observed agreement and p_e is the expected agreement by chance.

III. EXPERIMENTAL SETTINGS

The experimental setup for the SST-ATL incorporates several key parameters and methodologies to ensure clarity and reproducibility. Following data (HSI cube and ground truths) reading, image cubes are created from the processed data, and the dataset is split into training, validation, and test sets with ratios defined as training ratio set to 0.01%, pool ratio 0.49%, and test ration 0.50%. This results in an initial training sample count of 1% of the dataset, while the pool set comprises 49% of the remaining samples and the test set is 50% of the original data.

The model training is executed with several defined hyperparameters. The learning rate is set at 0.001 with a decay of 1×10^{-6}, utilizing the Adam optimizer. The training process spans 50 epochs with a batch size of 56. The architecture of SST incorporates multiple transformer layers and attention mechanisms. Specifically, the model includes 4 transformer layers, with each layer employing 8 attention heads. The dimensionality of the model (d_{model}) is set to 54, and the feedforward network's dimensionality is $4 \times 64 = 256$. The dropout rate is fixed at 0.1 to prevent overfitting, and the layer normalization epsilon value is set to 1×10^{-6}.

The model's input shape is defined as $(W \times W \times k)$, where W represents the patch size, which is derived from the dimensions (k) of the input data. The model employs a patch embedding mechanism to transform input data into manageable segments for processing within the transformer architecture. The embeddings are enhanced with positional encoding to preserve spatial information, which is essential in processing sequential data in transformer networks.

The experimental framework incorporates AL techniques. Initially, the model is trained on a small dataset, after which AL iterations take place. The hybrid query strategy is utilized to query new samples from the pool, combining uncertainty sampling and diversity sampling methods. The queried samples are integrated into the training set, and the model undergoes fine-tuning through retraining with a subset of the last layers frozen. This iterative process continues for several cycles, ensuring the model adapts and improves its performance with each iteration.

Finally, the evaluation of the model's performance is conducted using the test set, calculating metrics such as OA, AA, PA, and κ to assess the quality of predictions. This comprehensive setup emphasizes the importance of hyperparameters, dataset partitioning, model architecture, and evaluation strategies in the context of HSI classification, providing a clear framework for reproducibility.

IV. EXPERIMENTAL DATASETS

This section introduces the datasets used for evaluating the proposed model, highlighting their spectral and spatial characteristics and source details. Table I overviews the class names, initial training sample sizes, validation pool, and test set samples.

The **Salinas (SA)** dataset is a hyperspectral remote sensing image captured by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor, depicting the Salinas Valley region in southern California, USA. This dataset consists of images measuring 512 by 217 pixels, with a spatial resolution of 3.7 meters and a spectral resolution ranging from 9.7 to 12 nm, covering a spectral range of 400 to 2500 nm. It encompasses 16 ground object categories. The original dataset includes 224 spectral bands, but 20 bands affected by atmospheric water absorption and low signal-to-noise ratio (SNR) are typically excluded, leaving 204 bands for analysis.

The **Pavia University (PU)** was gathered using the Reflective Optics System Imaging Spectrometer (ROSIS) sensor during an aerial campaign over Pavia in northern Italy. This dataset comprises 103 spectral bands and has a spatial resolution of 1.3 meters, with an image size of 610×340 pixels. Prior to analysis, spectral samples without relevant information must be discarded. The ground truth data for this HSI is divided into 9 distinct classes.

The **WHU-Hi-HanChuan (HC)** dataset was acquired on June 17, 2016, between 17:57 and 18:46 in Hanchuan, Hubei Province, China. The data collection utilized a Headwall Nano-HSI sensor with a 17-mm focal length, mounted on a Leica Aibot X6 UAV V1 platform. The acquisition occurred under clear skies, with a temperature of around $30°C$ and relative humidity of approximately 70%. This dataset encompasses a rural-urban transitional zone that includes various features such as buildings, water bodies, and agricultural fields. Specifically, it captures seven crop types: strawberry, cowpea, soybean, sorghum, water spinach, watermelon, and leafy greens. The

TABLE I: Training, Pool, and Test Sample Counts for Each Class Across Five Hyperspectral Datasets (WHU-Hi-HongHu (HH), WHU-Hi-HanChuan (HC), Salinas (SA), University of Houston (UH), and Pavia University (PU)). Each column group represents the class names along with the corresponding number of samples allocated to training, pooling, and testing for each dataset.

WHU-Hi-HongHu (HH)				WHU-Hi-HanChuan (HC)				Salinas (SA)				University of Houston (UH)				Pavia University (PU)				Qingyun (QUH)			
Class	Train	Pool	Test	Class	Train	Pool	Test	Class	Train	Pool	Test	Class	Train	Pool	Test	Class	Train	Pool	Test	Class	Train	Pool	Test
Red roof	70	6951	7020	Strawberry	224	22143	22368	Broccoli green weeds 1	10	995	1004	Healthy grass	6	619	626	Asphalt	33	3282	3316	Trees	1391	137684	139075
Road	18	1738	1756	Cowpea	114	11262	11377	Broccoli green weeds 2	19	1844	1863	Stressed grass	6	621	627	Meadows	93	9232	9324	Concrete building	897	88859	89756
Bare soil	109	10801	10911	Soybean	51	5093	5143	Fallow	10	978	988	Synthetic grass	4	345	348	Gravel	10	1040	1049	Car	69	6822	6892
Cotton	816	80826	81643	Sorghum	27	2649	2677	Fallow rough plow	7	690	697	Trees	6	616	622	Trees	15	1517	1532	Ironhide building	49	4835	4883
Cotton firewood	31	3078	3109	Water spinach	6	594	600	Fallow smooth	13	1326	1339	Soil	6	615	621	Painted metal sheets	7	665	673	Plastic playground	1088	107779	108868
Rape	223	22055	22279	Watermelon	23	2244	2266	Stubble	20	1959	1980	Water	6	161	162	Bare Soil	25	2490	2514	Asphalt road	1280	126693	127973
Chinese cabbage	120	11931	12052	Greens	29	2922	2952	Celery	18	1771	1790	Residential	7	627	634	Bitumen	7	658	665	Total	4774	472672	477447
Pakchoi	20	2007	2027	Trees	90	8899	8989	Grapes untrained	56	5579	5636	Commercial	6	616	622	Self-Blocking Bricks	18	1823	1841				
Cabbage	54	5356	5409	Grass	47	4688	4734	Soil vineyard develop	31	3070	3102	Road	6	620	626	Shadows	5	468	474				
Tuber mustard	62	6135	6197	Red roof	52	5206	5258	Corn senesced green weeds	16	1623	1639	Highway	6	607	614	Total	213	21175	21388				
Brassica parachinensis	55	5453	5507	Gray roof	84	8372	8455	Lettuce romaine 4wk	5	529	534	Railway	6	611	618								
Brassica chinensis	45	4432	4477	Plastic	18	1821	1840	Lettuce romaine 5wk	10	954	963	Parking Lot 1	6	610	617								
Small Brassica chinensis	113	11140	11254	Bare soil	46	4512	4558	Lettuce romaine 6wk	5	453	458	Parking Lot 2	3	232	234								
Lactuca sativa	37	3641	3678	Road	93	9187	9280	Lettuce romaine 7wk	5	530	535	Tennis Court	2	212	214								
Celtuce	5	496	501	Bright object	6	562	568	Vineyard untrained	36	3598	3634	Running Track	3	327	330								
Film covered lettuce	36	3595	3631	Water	377	37324	37700	Vineyard vertical trellis	9	895	903	Total	75	7439	7515								
Romaine lettuce	15	1490	1505	Total	1287	127478	128765	Total	270	26794	27065												
Carrot	16	1593	1608																				
White radish	44	4312	4356																				
Garlic sprout	17	1726	1743																				
Broad bean	7	657	664																				
Tree	20	2000	2020																				
Total	1933	191413	193347																				

UAV operated at an altitude of 250 meters, producing images with dimensions of 1217 × 303 pixels across 274 spectral bands, covering a range from 400 to 1000 nm. The spatial resolution of the imagery is approximately 0.109 meters. It is important to note that the dataset contains substantial shadowed areas due to the low solar elevation angle during the late afternoon acquisition.

The **WHU-Hi-HongHu (HH)** dataset was collected on November 20, 2017, between 16:23 and 17:37 in Honghu City, Hubei Province, China, using a Headwall Nano-Hyperspec imaging sensor with a 17-mm focal length mounted on a DJI Matrice 600 Pro UAV. The data acquisition occurred under cloudy conditions, with an ambient temperature of approximately 8°C and a relative humidity of around 55%. This dataset covers a diverse agricultural landscape with multiple crop types, including different cultivars of the same species, such as Chinese cabbage and cabbage, as well as Brassica chinensis and small Brassica chinensis. The UAV operated at an altitude of 100 meters, capturing imagery with dimensions of 940 × 475 pixels across 270 spectral bands, spanning wavelengths from 400 to 1000 nm, and providing a spatial resolution of roughly 0.043 meters.

The University of Houston (UH) dataset was collected during the 2013 IEEE GRSS Data Fusion Contest and has been extensively utilized due to its diverse land cover types and challenging classification scenarios. It consists of hyperspectral imagery captured by the Compact Airborne Spectrographic Imager (CASI), which records reflectance data across 144 spectral bands, ranging from 380 nm to 1050 nm. The dataset provides a rich spectral profile for each pixel, enabling detailed analysis of the spatial-spectral characteristics of various land cover classes. The UH covers a spatial region of 349 × 1905 pixels with a spatial resolution of 2.5 meters per pixel, representing an area in the UH campus and its surrounding urban landscape. The region includes complex urban and natural features such as vegetation, buildings, roads, and parking lots, making it an ideal testbed for HSI classification techniques. The dataset is challenging due to the variability in spectral signatures caused by different lighting conditions, atmospheric effects, and material diversity. Ground truth labels are available for 15 land cover classes, representing diverse categories. These classes differ significantly in their spatial extent and spectral characteristics, providing a rigorous evaluation framework for classification models.

The **QUH-Qingyun** dataset was collected on 18 May 2021, near Qingyun Road Primary School and the surrounding residential area in Qingdao, China. Using a UAV at an altitude of 300 meters, images were captured with a spatial resolution of approximately 0.15 meters, an image size of 880 × 1360 pixels, and 270 spectral bands covering wavelengths from 400 to 1000 nm.

V. ACTIVE TRANSFER LEARNING EFFECTS ON THE SAME DATASET

The results presented in Table II highlight the consistent improvement in classification accuracy as the number of training samples increases. Most classes show steady progress, with notable gains in challenging categories such as Water, which improves from 49.4% to 98.1%, and Residential, which rises from 57.9% to 94.5%. These trends demonstrate the model's ability to effectively utilize additional training data to improve performance, particularly for underrepresented classes. Certain classes, such as Soil and Running Track, achieve near-perfect accuracy with minimal training samples, suggesting they are well-captured in the feature space. However, difficult classes like Parking Lot 2 begin with very low accuracy (1.7%) but see a significant improvement, reaching 77.8% as the sample size increases. This emphasizes the importance of ensuring sufficient representation of all classes in the training set.

Table II also provides computational metrics, showcasing the model's efficiency. Training times remain stable at around 44 seconds across iterations, despite the increase in sample size, indicating scalability. The testing time is consistently around 3.4 seconds, demonstrating the method's suitability for real-time applications. The reported FLOPs (110,336) and parameter count (836,559) reflect the model's balance between computational efficiency and capacity, making it practical for resource-constrained environments. Figure 2 further supports these findings by illustrating the trends in accuracy metrics—OA, AA, and κ—as the number of training samples increases. All metrics exhibit significant growth, with the most substantial improvements occurring at lower sample counts (75 to 223), followed by a diminishing rate of improvement beyond 510 samples. OA consistently outperforms AA and κ,

TABLE II: **UH Dataset**: Per-class accuracy for training samples, showing accuracy improvements as the sample count increases per iteration. Additionally, average training and testing times (in seconds) and the total FLOPs and parameter count provide a comprehensive overview of computational requirements.

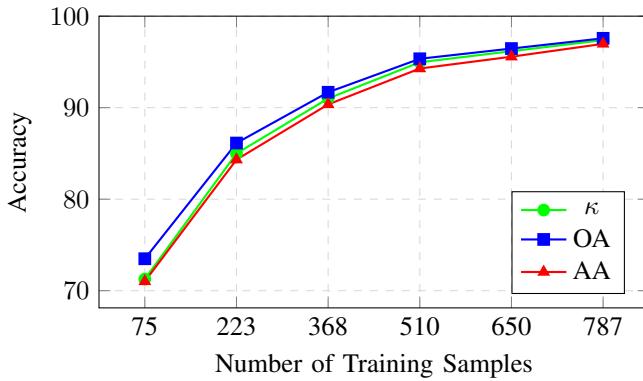| Class	Per-Class Accuracy for Training Samples					
	75	223	368	510	650	787
Healthy grass	86.4217	95.5271	96.3258	97.2843	98.0830	98.4025
Stressed grass	97.4481	99.5215	98.8835	97.9266	99.0430	99.5215
Synthetic grass	93.1034	95.1149	91.0919	93.1034	98.2758	100
Trees	95.0160	97.2668	94.3729	97.1061	99.1961	97.9099
Soil	100	99.8389	100	100	100	100
Water	49.3827	83.9506	90.7407	93.8271	95.6790	98.1481
Residential	57.8864	73.1861	90.2208	93.8485	94.4794	94.4794
Commercial	62.3794	61.2540	83.9228	94.0514	95.1768	95.9807
Road	60.8626	90.5750	96.3258	96.1661	96.4856	97.9233
Highway	93.3224	83.3876	88.9250	98.5342	98.8599	100
Railway	35.5987	95.1456	92.8802	96.4401	98.8673	99.0291
Parking Lot 1	53.8087	75.8508	85.0891	91.0858	91.0858	97.5688
Parking Lot 2	1.7094	23.0769	46.5811	66.6666	68.8034	77.7777
Tennis Court	85.0467	91.1214	100	98.1308	99.5327	100
Running Track	93.3333	100	100	100	100	97.8787
Train (s)	50.52	44.63	44.98	44.34	44.96	47.39
Test (s)	4.43	3.45	3.45	3.35	3.44	3.44
Flops	110336					
Param	836559					

Fig. 2: **UH Dataset**: OA, AA, and κ accuracy as a function of training samples in each iteration as shown in Table II.
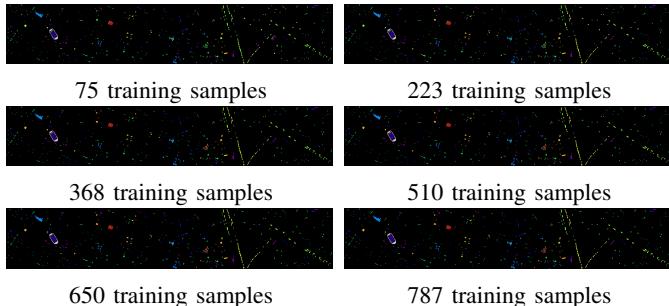

Fig. 3: **UH dataset**: Ground truth maps corresponding to varying numbers of training samples.

indicating robust performance across most classes while maintaining reliability. The steady upward trend demonstrates the model's ability to leverage additional training data effectively.

Qualitative improvements are visually evident in Figure 3, which presents ground truth maps for varying training sample sizes. With fewer training samples (e.g., 75 and 223), the maps exhibit substantial noise and misclassification. As the training

sample size increases, the maps become progressively refined, reflecting improved spatial and spectral consistency. Classes such as Road and Water demonstrate better spatial coherence at higher sample sizes (650 and 787), while sparse classes like Parking Lot 2 clearly reduce misclassification.

The results presented in Table III and Figures 4 and 5 collectively highlight the model's performance progression on the PU dataset as training samples increase. Table III provides a detailed per-class accuracy breakdown, showing substantial accuracy improvements for most classes as sample counts rise, especially for challenging classes like Gravel and Shadows, which see increases of over 20 percentage points. This indicates that the model's learning efficacy improves with a larger training dataset, supporting robust classification for diverse material types. Notably, Metal Sheets achieves perfect classification (100%) consistently as sample size grows. Training and testing times remain relatively stable across sample sizes, indicating computational efficiency; a total of 108,800 FLOPs and 835,017 parameters suggests a well-optimized model that balances accuracy and computational overhead. The accompanying Figure further substantiates these findings by plotting OA, AA, and κ scores against sample counts, illustrating a clear upward trend with convergence near 99% as sample sizes approach 2245. This consistent improvement across all three metrics emphasizes that the model achieves high precision and reliability across iterations, which is critical for effective hyperspectral image classification applications in remote sensing.

TABLE III: **PU Dataset**: Per-class accuracy for training samples, showing accuracy improvements as the sample count increases per iteration. Additionally, average training and testing times (in seconds) and the total FLOPs and parameter count provide a comprehensive overview of computational requirements.

Class	Per-Class Accuracy for Training Samples					
	213	636	1051	1457	1855	2245
Asphalt	87.7563	96.8938	96.5319	96.8938	98.5524	98.7937
Meadows	96.9540	98.3805	99.3350	99.7211	99.7640	99.9034
Gravel	69.5900	82.1734	80.8388	84.9380	87.8932	91.3250
Trees	84.3342	91.8407	95.1697	98.1070	98.6292	99.5430
Metal Sheets	99.8514	99.5542	100	100	100	100
Bare Soil	72.7128	80.7080	97.0564	98.9260	99.3237	99.6420
Bitumen	70.0751	87.5187	97.8947	96.2406	99.3984	99.5488
Bricks	75.0678	82.1835	92.0695	94.0249	94.7311	95.8718
Shadows	66.0337	92.6160	97.0464	98.3122	97.8902	98.9451
Train (s)	118.11	117.25	115.99	119.21	119.78	124.78
Test (s)	9.87	9.26	9.08	8.79	9.07	9.09
Flops	108800					
Param	835017					

Table IV and Figures 6 and 7 provide a thorough analysis of the model's performance, computational complexity, and the scalability of its classification accuracy on the SA dataset. Table IV presents per-class accuracy for various sample sizes, progressively increasing from 270 to 2842 samples. The results highlight the model's robust performance across most classes, achieving near-perfect accuracy as the number of training samples increases, particularly for classes like Broccoli 1, Celery, and Stubble. However, for more challenging classes, such as Grapes and Vinyard Untrained, a noticeable improvement in accuracy is observed with larger sample sizes. This demonstrates the model's need for a substantial amount of
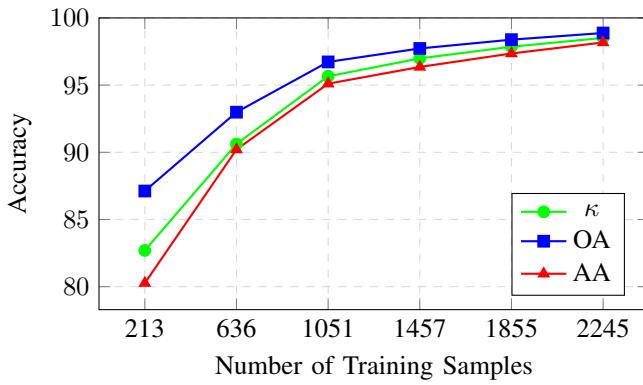
Fig. 4: **PU Dataset**: OA, AA, and κ accuracy as a function of training samples in each iteration as shown in Table III.
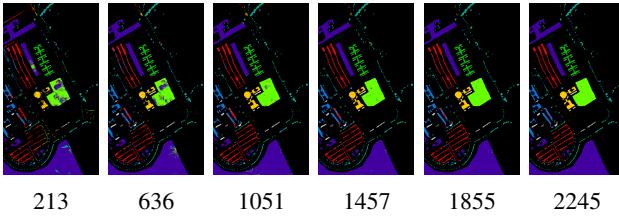

Fig. 5: **PU dataset**: Ground truth maps corresponding to varying numbers of training samples.

data for better performance in complex classifications. Additionally, the table includes average training and testing times to emphasize computational efficiency. Although training times gradually increase as sample sizes grow, testing times remain relatively stable, underscoring the model's scalability and efficiency. Furthermore, Table IV provides computational metrics, including FLOPs and parameter counts, which quantify the model's complexity at approximately 110,592 FLOPs and 836,816 parameters. These values confirm the model's high computational efficiency, which is an essential consideration for large-scale applications.
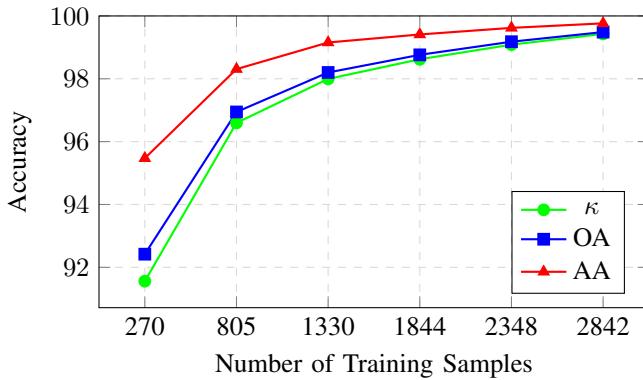

Fig. 6: **SA Dataset**: OA, AA, and κ accuracy as a function of training samples in each iteration as shown in Table IV.

Figure 6 offers a visual representation of the model's accuracy improvements with increased training samples by plotting OA, AA, and κ. All metrics display an upward trend, achieving

TABLE IV: **SA Dataset**: Per-class accuracy for training samples, showing accuracy improvements as the sample count increases per iteration. Additionally, average training and testing times (in seconds) and the total FLOPs and parameter count provide a comprehensive overview of computational requirements.

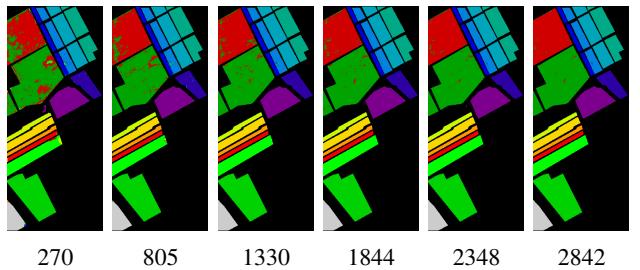| Class	Per-Class Accuracy for Training Samples					
	270	805	1330	1844	2348	2842
Broccoli 1	95.9163	100	100	100	99.5019	100
Broccoli 2	99.9463	100	100	100	100	100
Fallow	99.6963	98.4817	100	100	100	100
Fallow Rough	97.8479	98.5652	100	99.7130	100	100
Fallow Smooth	95.9671	98.2823	97.7595	99.4772	99.4025	99.4025
Stubble	99.4444	99.8484	100	100	100	100
Celery	100	99.4972	99.8882	99.9441	100	100
Grapes	84.6344	94.1447	97.0546	97.4627	98.2966	98.8644
Soil Vinyard	100	100	99.9677	100	100	99.9677
Corn Senesced	94.6918	98.6577	99.8169	100	100	99.9389
Lettuce 4wk	90.4494	100	100	99.4382	100	100
Lettuce 5wk	100	100	100	100	100	100
Lettuce 6wk	97.8165	99.3449	100	99.5633	100	100
Lettuce 7wk	97.7570	97.1962	99.8130	99.8130	99.8130	99.8130
Vinyard Untrained	75.8117	89.1304	92.1849	95.1568	96.9179	98.2388
Vinyard Vertical	97.5636	99.7785	100	100	100	100
Train (s)	149.00	144.89	149.44	152.20	155.20	157.85
Test (s)	12.44	11.40	11.60	11.32	10.69	11.49
Flops	110592					
Param	836816					

Fig. 7: **SA dataset**: Ground truth maps corresponding to varying numbers of training samples.

near-perfect accuracy as the sample size approaches 2842, demonstrating that the model effectively learns and generalizes as it is exposed to more data. Notably, the close convergence of OA, AA, and κ toward 100% signifies the model's balanced accuracy across metrics and low classification error when trained on sufficient data. Finally, Figure 7 provides ground truth maps generated with different training sample sizes, visually showcasing classification performance improvements. The maps show less distinct class boundaries with smaller sample sizes, while with larger sample sizes, the maps display sharper and more accurate distinctions between classes. This visual progression emphasizes the model's enhanced ability to correctly classify and delineate classes as it trains on more data.

Table V provides a comprehensive overview of the HC dataset's classification performance and computational demands across six iterations with increasing training sample sizes. The per-class accuracy shows a consistent improvement for all classes as more samples are added, emphasizing the importance of larger datasets for enhancing model learning. Classes like Strawberry and Sorghum achieve near-perfect accuracy early, suggesting distinct and easily separable spectral features. Conversely, classes such as Plastic and Bright

Object, which start with low accuracy, exhibit significant improvements, indicating the model's ability to learn complex patterns with additional training data. The computational metrics reveal a gradual increase in training time with more samples, reflecting the added processing requirements, while testing times remain relatively stable. Furthermore, the reported FLOPs and parameter counts provide insight into the model's computational efficiency, balancing accuracy with resource usage. Overall, the table highlights the scalability and effectiveness of the model in handling diverse classes and increasing sample sizes.

TABLE V: **HC Dataset**: Per-class accuracy for training samples, showing accuracy improvements as the sample count increases per iteration. Additionally, average training and testing times (in seconds) and the total FLOPs and parameter count provide a comprehensive overview of computational requirements.

Class	Per-Class Accuracy for Training Samples					
	1287	3836	6334	8782	11181	13532
Strawberry	95.1180	95.7662	97.8227	98.6230	99.0164	99.5082
Cowpea	84.8290	93.2934	96.1501	97.7410	98.7254	99.0419
Soybean	74.2951	87.1475	93.3696	96.9861	98.2694	99.2416
Sorghum	96.9742	96.8248	98.0201	98.9914	99.5890	99.7011
Water spinach	30.3333	86.5	89.3333	93.3333	98.8333	100
Watermelon	46.3812	70.9620	79.4792	87.4669	93.2480	95.5428
Greens	71.4769	85.7384	91.6666	94.5121	95.9688	96.8157
Trees	72.1659	85.3153	90.9222	94.5933	96.0952	97.3968
Grass	70.2788	91.5504	95.4795	97.4651	98.4157	98.9860
Red roof	89.4256	95.3974	98.4214	99.4104	99.6006	99.7907
Gray roof	92.8799	94.8078	96.6410	98.2022	98.9355	99.4795
Plastic	24.5108	63.5869	80.5978	92.8804	95.5978	97.5543
Bare soil	60.6186	72.1588	81.8560	87.1434	91.5313	94.0105
Road	84.4504	86.8426	93.2758	97.0689	98.9224	99.3642
Bright object	27.1126	69.8943	92.1849	83.2746	88.7323	95.4225
Water	98.9098	98.6312	99.4694	99.6870	99.8435	99.8992
Train (s)	523.75	544.15	547.76	598.53	618.34	648.47
Test (s)	53.42	50.17	51.15	55.11	54.07	54.52
Flops	110592					
Param	836816					

Fig. 9: **HC dataset**: Ground truth maps corresponding to varying numbers of training samples.

the model to generalize better across all classes. κ, which measures agreement beyond chance, also exhibits steady growth, confirming improved classification consistency. Figure 9 showcases the ground truth maps for each training sample size, visually representing the model's learning progress. With 1,287 samples, the classification appears fragmented, reflecting limited learning. As the sample size increases, the maps exhibit progressively finer details, and class boundaries become more accurate and well-defined. By the final iteration with 13,532 samples, the map closely approximates the true spatial distribution, demonstrating the model's capability to effectively leverage larger datasets for classification.

TABLE VI: **HH Dataset**: Per-class accuracy for training samples, showing accuracy improvements as the sample count increases per iteration.

Class	Per-Class Accuracy for Training Samples					
	1933	5761	9512	13188	16791	20322
Red roof	94.8433	97.7635	98.0056	99.5868	99.7435	99.7578
Road	70.6719	85.5353	91.7425	94.4191	96.0706	95.1025
Bare soil	92.6496	95.4999	97.1038	97.8828	99.0651	99.0926
Cotton	99.3140	99.2087	99.4794	99.4181	99.8873	99.7133
Cotton firewood	46.1241	62.3673	79.0286	92.2161	95.1109	98.4239
Rape	93.5050	95.1658	97.9397	99.0349	99.4120	99.3222
Chinese cabbage	81.3806	93.4782	96.5482	96.5648	98.8217	98.3820
Pakchoi	36.2604	67.8835	81.3024	89.1958	93.7839	96.4479
Cabbage	93.8990	97.1528	99.1865	99.7041	99.6487	99.4638
Tuber mustard	68.4524	83.8631	92.4640	94.9491	97.3535	96.1271
Brassica parachinensis	71.7087	90.5574	94.4252	94.8792	98.4746	97.9299
Brassica chinensis	44.4047	72.9953	87.9606	96.0464	96.3368	98.2800
Small Brassica chinensis	69.7174	86.8757	93.7799	94.5352	97.8141	97.7963
Lactuca sativa	73.4638	86.4328	95.8945	97.1995	98.6677	98.6677
Celtuce	59.4810	91.6167	96.4071	97.2055	96.8063	97.0059
Film covered lettuce	88.2401	95.2630	98.4301	98.1823	99.6144	97.6590
Romaine lettuce	82.5249	86.8438	94.9501	99.4019	98.6046	99.8671
Carrot	74.3781	94.4651	95.0248	95.2114	98.0721	99.0049
White radish	85.2617	91.8273	96.5564	95.9825	98.4848	98.7144
Garlic sprout	58.5771	88.6976	91.9678	95.0659	97.8772	94.1480
Broad bean	13.8554	53.7650	71.2349	79.0662	94.2771	94.8795
Tree	66.2871	90.9900	92.3267	97.9702	99.1584	98.8118
Train (s)	272.05	273.63	277.96	285.60	291.45	291.68
Test (s)	23.30	22.56	22.42	22.73	23.42	22.89
Flops	112128					
Param	838358					

Fig. 8: **HC Dataset**: OA, AA, and κ accuracy as a function of training samples in each iteration as shown in Table 8.
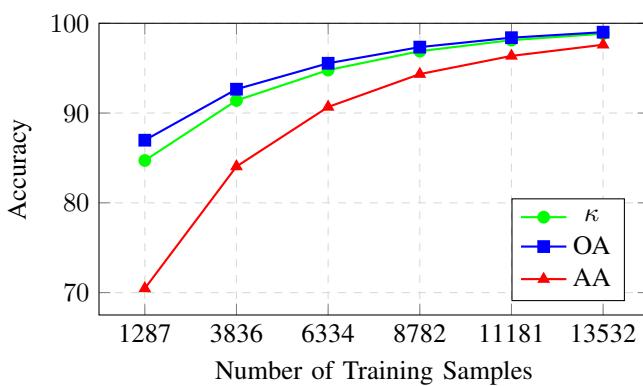
Figure 8 illustrates the OA, AA, and κ as functions of training sample size across six iterations. The trends clearly demonstrate that increasing the number of training samples significantly improves all three metrics, with the model achieving near-optimal performance as the sample count reaches 13,532. The steep initial rise in accuracy, particularly for AA, indicates that even a moderate increase in sample size enables
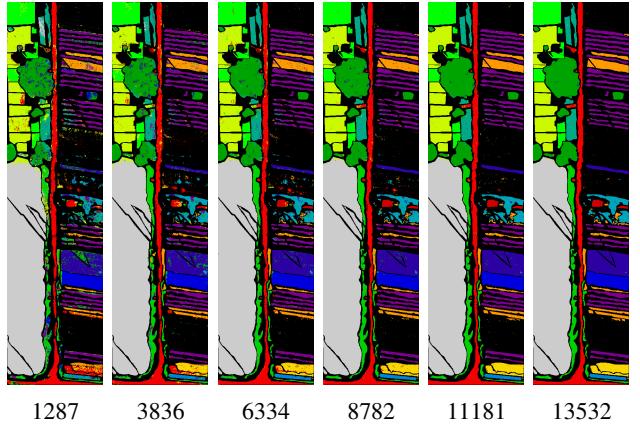
Table VI illustrates the per-class accuracy for a variety of land cover categories as the number of training samples rises from 1933 to 20,322. Significant accuracy gains are observed in challenging classes such as Cotton firewood, Pakchoi, and Broad bean, where accuracy improved from 46.12% to 98.42%, 36.26% to 96.44%, and 13.85% to 94.88%,
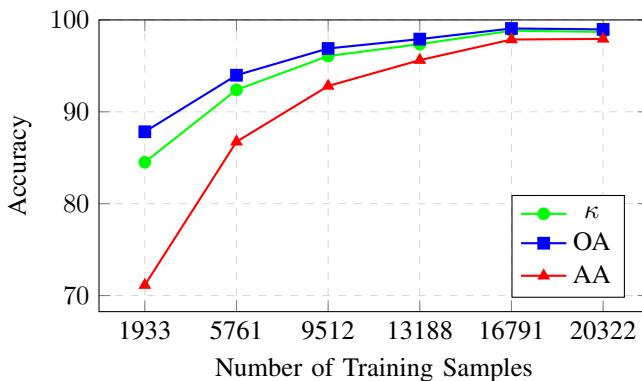
Fig. 10: **HH Dataset**: OA, AA, and κ accuracy as a function of training samples in each iteration as shown in Table VI.

Fig. 11: **HH dataset**: Ground truth maps corresponding to varying numbers of training samples.

TABLE VII: **QUH Dataset**: Per-class accuracy for training samples, showing accuracy improvements as the sample count increases per iteration.

Class	Per-Class Accuracy for Training Samples					
	4774	14227	23491	32570	41467	50186
Trees	91.8324	97.0037	97.3201	98.8538	99.0666	99.3291
Concrete building	92.0907	97.9266	98.9772	98.8401	99.5365	99.6869
Car	18.3401	81.5728	90.1334	92.7742	91.4828	96.6773
Ironhide building	98.7302	98.9350	96.5185	99.4675	99.5699	99.6108
Plastic playground	93.9587	97.1718	98.5064	98.9886	99.1788	99.6509
Asphalt road	91.9654	94.5551	96.3140	98.5950	98.6989	98.7598
Train (s)	2000.15	2122.98	2152.50	2248.08	2296.25	2360.75
Test (s)	219.09	227.46	231.50	239.75	239.00	234.79
Flops	108032					
Param	834246					

Fig. 12: **QUH Dataset**: OA, AA, and κ accuracy as a function of training samples in each iteration as shown in Table VII.
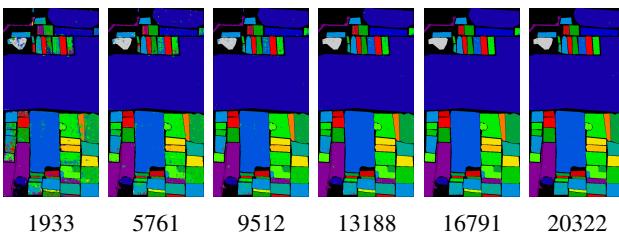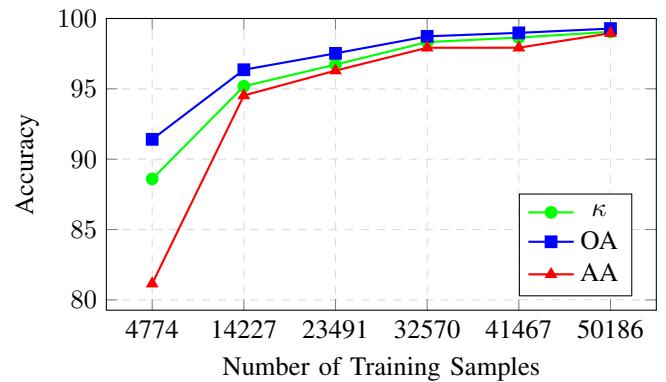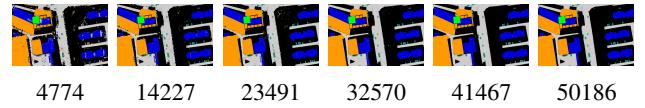

Fig. 13: **QUH dataset**: Ground truth maps corresponding to varying numbers of training samples.

respectively. These improvements highlight the model's ability to learn complex patterns with more training data. In contrast, classes with inherently distinct features, such as Cotton and Red roofs, already achieved high accuracy with fewer samples and exhibited marginal improvements. Table VI also includes computational metrics such as training and testing times and the total number of FLOPs and parameters. Despite increasing data, the training and testing times remain relatively stable, showcasing the model's computational efficiency. For example, training times range from 272.05 seconds to 291.68 seconds, and testing times fluctuate minimally between 22.42 seconds and 23.42 seconds. On the aggregate level, the OA, AA, and κ demonstrated consistent growth across iterations, as shown in Figure 10. For instance, OA increased from 87.83% to 98.97%, while AA rose from 71.14% to 97.94%, reflecting the model's robust performance across diverse classes. The κ followed a similar upward trajectory, indicating reduced misclassification rates and alignment with ground truth. Figure 11 presents ground truth maps corresponding to the training sample counts. These visualizations show the gradual enrichment of spatial details and class boundaries as training data increases. The progression from 1933 to 20,322 samples demonstrates the model's growing ability to capture intricate spatial and spectral variations. Notably, the maps for higher sample counts, such as 16,791 and 20,322, closely resemble the actual ground truth, highlighting the efficacy of the dataset's size in improving model generalization.

Table VII presents the per-class accuracy of the QUH dataset across six iterations of training samples, highlighting the progressive improvement in accuracy as the number of

samples increases. Classes such as "Trees" and "Concrete Building" exhibit significant gains, with accuracies exceeding 99% in the final iterations. Notably, the "Car" class starts with lower accuracy (18.34%) due to limited initial sample representation but shows remarkable improvement, reaching 96.68% by the last iteration. The table also reports training and testing times, which grow modestly with larger sample counts, and provides the model's computational metrics, such as FLOPs and parameters, indicating efficient scaling. Figure 12 visualizes the OA, AA, and κ as functions of training sample size, consistent with the trends in Table VII. All three metrics show steady improvement, converging towards optimal values as sample size increases, underlining the model's robustness and scalability. Figure 13 illustrates ground truth maps for the dataset, corresponding to each training sample size. These maps visually demonstrate the enhancement in spatial coverage and label consistency as the number of training samples increases, corroborating the quantitative improvements observed in Table VII and Figure 12.

VI. ACTIVE TRANSFER LEARNING EFFECTS ON ACROSS DATASET

TABLE VIII: The OA, AA, and κ accuracies for cross-dataset classification were obtained using the actively trained SST on the SA dataset and evaluated across other datasets.

Metric	SST Trained	SST Fine-Tuned using 10% Samples				
	SA	PU	UH	HC	HH	QUH
κ	99.4281	96.4279	97.9568	93.9511	93.9427	95.7008
OA	99.4864	97.3069	98.1104	94.8339	95.2070	96.7533
AA	99.7641	95.4779	97.8930	90.1400	89.3021	93.9454
Train (s)	157.85	126.02	47.29	745.22	1552.11	2733.52
Test (s)	11.49	10.31	5.46	61.11	128.04	252.74

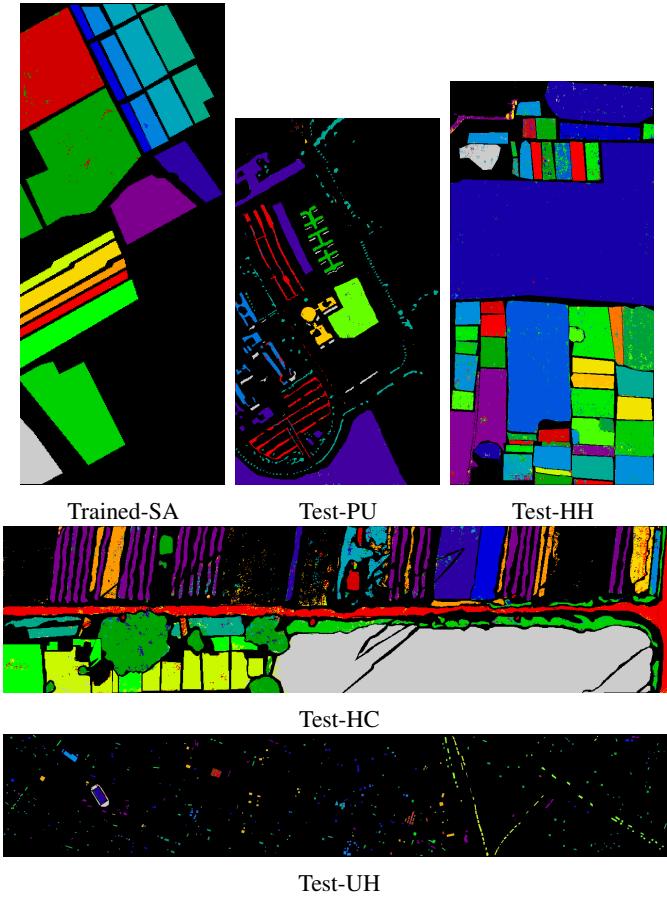Trained-SA Test-PU Test-HH

Test-HC

Test-UH

Fig. 14: **SST trained on SA** was fine-tuned and tested on PU, HC, HH, and UH datasets, respectively.

This section explores the impact of active transfer learning (ATL) in enabling effective across-dataset classification. By leveraging the actively trained spatial-spectral transformer (SST) on one dataset and fine-tuning it with a limited fraction of samples from other datasets, we assess its ability to generalize and adapt to varying spectral-spatial distributions. The results highlight the significance of ATL in achieving high accuracy with minimal fine-tuning across diverse datasets.

Table VIII presents the OA, AA, and κ coefficient for cross-dataset classification, where the ATL-SST is initially trained on the SA dataset and subsequently fine-tuned using 10% of the samples from other datasets (PU, UH, HC, HH, and QUH). The results indicate that the model achieves high accuracy across datasets, with κ values consistently above 93%,

demonstrating effective transfer learning. Training and testing times vary significantly across datasets due to differences in data volume and complexity, with QUH requiring the longest computation times. Figure 14 provides visual representations of classification outcomes, showcasing the trained SST on SA and its fine-tuned performance on PU, HC, HH, and UH datasets. Each subfigure illustrates distinct patterns, emphasizing the model's adaptability to different data distributions. Together, the table and figure underscore the SST's robust cross-dataset generalization capabilities, particularly with minimal fine-tuning.

TABLE IX: The OA, AA, and κ accuracies for cross-dataset classification were obtained using the actively trained SST on the PU dataset and evaluated across other datasets.

Metric	SST Trained	SST Fine-Tuned using 10% Samples				
	PU	SA	UH	HC	HH	QUH
κ	98.5130	97.7532	96.1730	91.8089	92.4317	94.8548
OA	98.8778	97.9826	96.4604	93.0097	94.0118	96.1137
AA	98.1747	98.8340	96.0646	87.5593	87.7129	92.7979
Train (s)	124.78	156.22	49.0509	717.08	1107.30	2698.77
Test (s)	9.09	12.83	3.71	59.67	89.13	239.48

Trained-PU Test-SA Test-HH

Test-HC

Test-UH
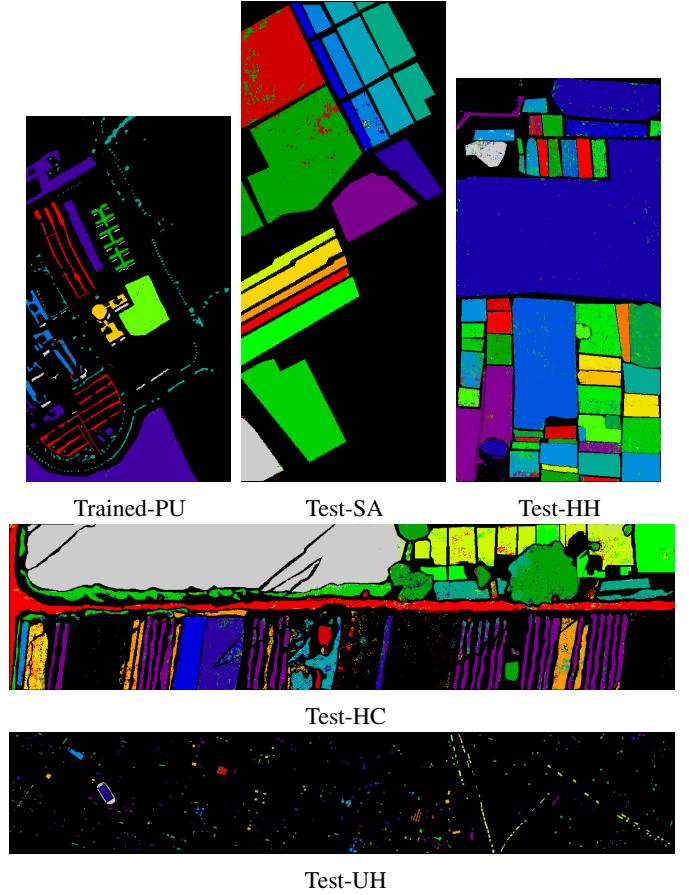
Fig. 15: **SST trained on PU** was fine-tuned and tested on SA, HC, HH, and UH datasets, respectively.

Table IX and Figure 15 illustrate the performance of the SST trained on the PU dataset and subsequently fine-tuned and evaluated on other datasets, including SA, UH, HC, and HH. Table IX provides detailed metrics—highlighting the transferability and generalization ability of the SST model.

The actively trained SST on PU achieves a κ of 98.51% and an OA of 98.88% on its test data, demonstrating robust performance on the source dataset. When fine-tuned with only 10% samples from other datasets, the model maintains competitive accuracy, with OA values exceeding 96% for most target datasets. Training and testing times are also reported, indicating computational efficiency during model fine-tuning and evaluation. While the training time scales with dataset complexity, testing times remain relatively low, demonstrating the efficiency of the SST architecture. Figure 15 visually complements these results by showcasing the SST model's classification maps on the source dataset (PU) and various target datasets (SA, HC, HH, and UH). The maps indicate a strong generalization capability, as the fine-tuned model effectively captures spectral-spatial features across diverse datasets with minimal sample utilization. This analysis underscores the SST's potential for cross-dataset classification, even with limited target domain data.

TABLE X: The OA, AA, and κ accuracies for cross-dataset classification were obtained using the actively trained SST on the UH dataset and evaluated across other datasets.

Metric	SST Trained	SST Fine-Tuned using 10% Samples				
	UH	SA	PU	HC	HH	QUH
κ	97.3807	98.2996	97.0131	93.2679	94.0447	95.6066
OA	97.5781	98.4740	97.7463	94.2523	95.2929	96.6787
AA	96.9746	99.1954	96.3262	90.4259	89.4059	94.3506
Train (s)	47.39	161.59	129.94	741.44	1167.53	2808.13
Test (s)	3.44	12.8567	10.81	69.98	97.30	242.65

Table X and Figure 16 present the performance of the spatial-spectral transformer (SST) actively trained on the UH dataset and fine-tuned with 10% samples from other target datasets, including SA, PU, HC, and HH. Table X outlines key metrics—highlighting the SST model's capability to generalize and adapt to new datasets with minimal fine-tuning. The actively trained SST on UH achieves high baseline performance, with a κ of 97.38% and an OA of 97.58% on its test data. When fine-tuned with samples from other datasets, the model demonstrates strong transferability, achieving an OA of 98.47% on SA and 97.75% on PU. Although the performance is slightly reduced for HC and HH, the OA values remain above 94%, underscoring the model's robustness. Training and testing times scale with dataset complexity, with SA and PU requiring less computation compared to HC and HH, as seen in the recorded durations. Figure 16 complements the table by visualizing classification results. The maps indicate that the fine-tuned SST effectively captures spatial and spectral features across diverse datasets. Specifically, the SST retains consistent classification quality on target datasets despite using only 10% of the samples for fine-tuning. This outcome demonstrates the SST's adaptability and its potential for cross-dataset applications in classification.

VII. COMPARISON WITH STATE-OF-THE-ART METHODS

The following Tables and Figure illustrate a comparative evaluation of various HSI classification methods on the University of Houston (UH), Pavia University (PU), and Salina (SA) datasets. The methods compared include Attention Graph CNN (AGCN), Hybrid Spatial-Spectral
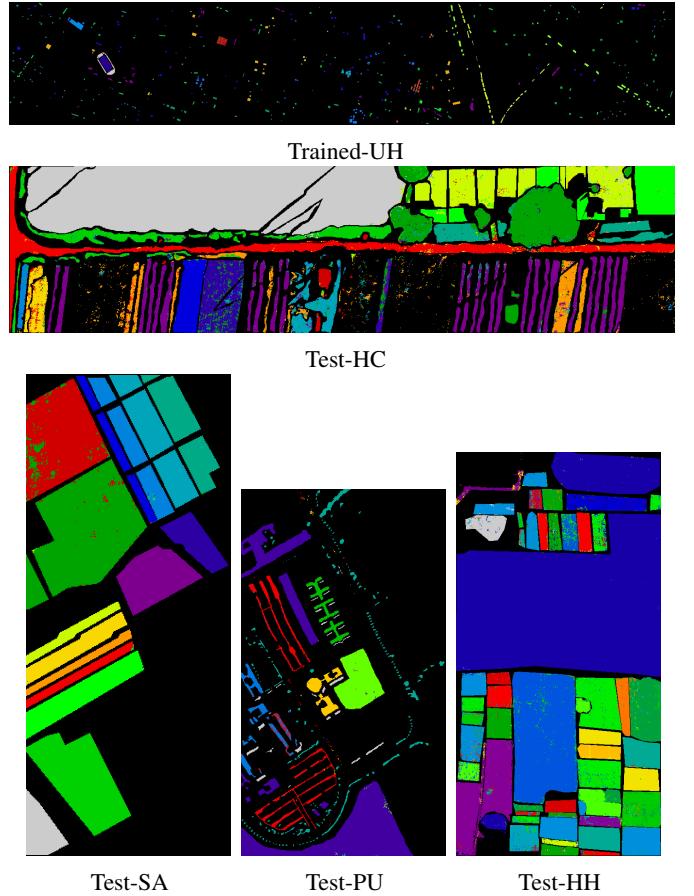

Trained-UH

Test-HC

Test-SA   Test-PU   Test-HH

Fig. 16: **SST trained on UH** was fine-tuned and tested on SA, HC, HH, and PU datasets, respectively.

Transformer (HybViT), Spatial-Spectral Mamba (SSMamba), Spatial-Spectral Transformer (SST), Wavelet-Based Spatial-Spectral Transformer (WaveFormer), Pyramid-Based Spatial-Spectral Transformer (PyFormer), and Wavelet-Based Spatial-Spectral Mamba (WaveMamba) [50]. These state-of-the-art approaches are benchmarked against the proposed method, ATL-SST, under consistent training, validation, and testing configurations, as summarized in the Tables.

As shown in Table XI, the results highlight the superior performance of ATL-SST across most experimental settings. For OA, ATL-SST consistently achieves the highest values, with a maximum of 100% in multiple scenarios, underscoring its robustness and effectiveness. Similarly, ATL-SST outperforms all comparative methods in terms of κ and AA, achieving top scores of 97.38% and 96.97%, respectively. This reflects the method's ability to effectively classify diverse classes within the hyperspectral dataset. Notably, ATL-SST demonstrates significant improvement over earlier methods like AGCN and WaveMamba, with AGCN yielding lower performance metrics (e.g., 91.35% OA and 90.64% κ) in many cases. While HybViT and SST exhibit strong results, particularly in individual settings, their performance is generally less consistent compared to ATL-SST. The training time analysis also reveals ATL-SST's computational efficiency, with an average training time comparable to SST and WaveFormer,

TABLE XI: **UH dataset:** Comparison of various HSI classification methods (Attention Graph CNN = AGCN, Hybrid Spatial-spectral Transformer = HybViT, Spatial-spectral Mamba = SSMamba) highlighting performance metrics.

Samples			State-of-the-art Comparative Methods							ATL-SST
Tr	Va	Te	AGCN	HybViT	SST	WaveFormer	PyFormer	SSMamba	WaveMamba	
43	582	626	95.3674	93.6102	94.5686	94.8881	85.3035	91.8530	93.9297	98.4025
42	585	627	99.8405	99.3620	99.2025	99.0430	100	96.6507	98.4051	99.5215
26	323	348	99.1379	100	99.7126	98.5632	95.6896	93.6781	97.7011	100
44	578	622	94.6945	97.5884	97.2668	97.5884	98.8745	96.6237	94.8553	97.9099
42	579	621	100	99.8389	100	100	94.5249	96.6183	96.9404	100
17	146	162	87.0370	80.2469	98.1481	88.8888	88.8888	80.8641	87.6543	98.1481
106	528	634	89.7476	91.9558	92.2712	94.4794	89.2744	80.4416	85.4889	94.4794
81	541	622	87.2990	90.3536	92.7652	96.4630	87.6205	83.9228	90.9967	95.9807
81	545	626	69.9680	91.2140	93.7699	90.0958	71.7252	65.0159	83.2268	97.9233
69	544	614	96.0912	96.7426	99.5114	99.0228	84.8534	80.2931	89.7394	100
70	547	618	94.6601	95.9546	96.7637	94.8220	90.9385	86.0841	92.8802	99.0291
68	548	617	81.0372	99.0275	98.8654	97.4068	80.8752	92.7066	92.5445	97.5688
55	180	234	87.6068	82.0512	79.4871	72.2222	57.6923	34.6153	73.5042	77.7777
15	199	214	95.3271	96.2616	97.1962	97.1962	97.1962	92.5233	98.5981	100
28	302	330	94.5454	95.4545	100	100	100	81.5151	95.7575	97.8787
Train (s)			101.61	50.85	46.74	47.73	397.21	67.21	563.52	47.39
κ			90.6479	94.6165	96.0141	95.3937	87.6230	84.2658	91.1925	**97.3807**
OA			91.3506	95.0232	96.3140	95.7418	88.5562	85.4690	91.8562	**97.5781**
AA			91.4906	93.9775	95.9686	94.7120	88.2305	83.5604	91.4815	**96.9746**

AttentionGCN

SST

HybViT

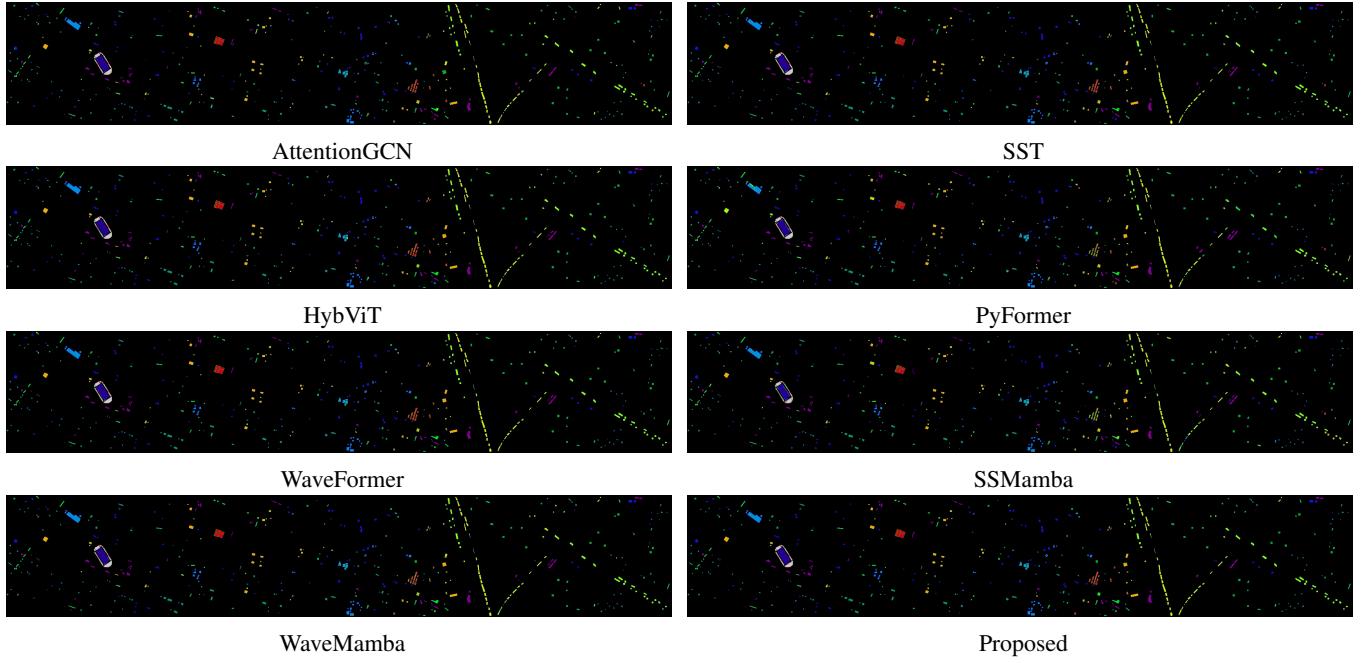PyFormer

WaveFormer

SSMamba

WaveMamba

Proposed

Fig. 17: **UH dataset:** Ground truth maps for comparative methods alongside the proposed ATL-SST.

TABLE XII: **PU dataset:** Comparison of various HSI classification methods highlighting performance metrics.

Samples			State-of-the-art Comparative Methods							ATL-SST
Tr	Va	Te	AGCN	HybViT	SST	WaveFormer	PyFormer	SSMamba	WaveMamba	
422	2893	3316	99.4571	97.1351	97.4065	97.5572	96.8636	88.7515	97.6477	98.7937
535	8790	9324	98.0158	99.1419	99.0240	99.5388	98.9382	94.7447	99.5388	99.9034
218	832	1049	69.0181	84.6520	84.7473	82.9361	93.5176	74.7378	75.9771	91.3250
161	1371	1532	97.5195	96.2793	96.0182	95.1697	97.3237	92.2323	94.3864	99.5430
15	657	673	97.7711	100	100	100	100	98.9598	98.2169	100
311	2204	2514	99.2044	97.7326	97.1360	96.2609	94.3516	93.3571	98.6077	99.6420
114	551	665	87.9699	88.1203	92.9323	91.2781	75.0375	82.2556	92.3308	99.5488
405	1436	1841	93.9163	89.5708	90.0597	92.2868	63.0635	81.9663	88.4845	95.8718
64	409	474	97.0464	93.4599	96.8354	94.9367	89.4514	89.4514	90.5063	98.9451
Train (s)			72.49	117.58	114.56	112.68	1181.18	154.76	1382.70	124.78
κ			95.0080	95.3372	95.5679	95.6819	91.6276	88.1644	94.9488	**98.5130**
OA			96.2268	96.4840	96.6570	96.7505	93.6880	91.0183	96.1941	**98.8778**
AA			93.3243	94.0102	94.9066	94.4405	89.8386	88.4952	92.8551	**98.1747**
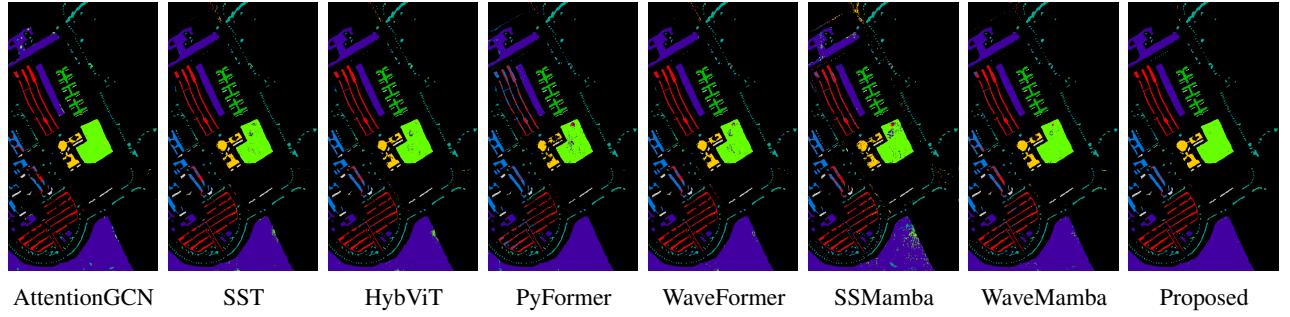
Fig. 18: **PU dataset:** Ground truth maps for comparative methods alongside the proposed ATL-SST.

AttentionGCN SST HybViT PyFormer WaveFormer SSMamba WaveMamba Proposed

and significantly lower than methods like WaveMamba and PyFormer. This balance of performance and efficiency makes ATL-SST a promising solution for practical HSI classification applications. The ground truth maps preseneted in Figure 17 visually depicts the classification results for the UH dataset. It is evident from the maps that ATL-SST provides more precise and coherent classification results, with fewer misclassified pixels and better-defined class boundaries compared to the comparative methods. For instance, methods like AGCN and WaveMamba exhibit noticeable misclassifications and noise, whereas ATL-SST maintains high spatial and spectral accuracy. The enhanced performance can be attributed to ATL-SST's hybrid query mechanism and spatial-spectral feature integration, which enable more effective discrimination of classes within the hyperspectral data.

The results presented in Table XII and Figure 18 comprehensively compare the performance of the proposed ATL-SST method with state-of-the-art techniques for HSI classification on the PU dataset. Quantitative results in the Table highlight the superiority of ATL-SST across key performance metrics, while the qualitative analysis in the figure illustrates its effectiveness in generating accurate classification maps.

In terms of accuracy, ATL-SST consistently outperforms all other methods, achieving the highest OA of 98.8778%, AA of 98.1747%, and κ of 98.5130%. These results are a significant improvement over the next best-performing methods, such as WaveMamba, which achieves an OA of 96.1941%. The robust performance of ATL-SST is evident across varying data splits, including scenarios with limited training samples. For instance, with only 15 training samples, ATL-SST achieves 100% accuracy, matching the best results from methods like WaveFormer while surpassing models such as AGCN, which only reaches 97.7711%. This demonstrates ATL-SST's ability to generalize effectively across diverse data distributions and sample sizes. From a computational perspective, ATL-SST maintains competitive efficiency, with a training time of 124.78 seconds, comparable to HybViT and SST while significantly outperforming computationally intensive models like Wave-Mamba and WaveFormer, which require 1382.70 and 112.68 seconds, respectively.

Qualitative results presented in Figure 18 further validate the quantitative findings. The classification maps generated by ATL-SST are visually superior, exhibiting precise boundary delineation and accurate region classifications that closely match the ground truth. In contrast, competing methods, such

as AGCN and SSMamba, show noticeable errors, particularly in regions with complex class boundaries or high spectral variability. Transformer-based models like HybViT and Py-Former produce smoother classifications but still fail to capture the finer details that ATL-SST excels at. While wavelet-based methods such as WaveFormer and WaveMamba improve spectral representation, their outputs show slight boundary smoothing, leading to less accurate segmentation compared to ATL-SST.

The comparison of HSI classification methods on the SA dataset, as detailed in Table XIII and illustrated in Figure 19, highlights the superior performance of the proposed ATL-SST model across multiple metrics and qualitative benchmarks. Table XIII provides a comprehensive analysis of accuracy, computational efficiency, and robustness under various training, validation, and testing splits, while Figure 19 visually corroborates these findings with ground truth and classification maps.

Quantitatively, ATL-SST demonstrates outstanding accuracy, achieving the highest OA of 99.4864%, AA of 99.7641%, and κ of 99.4281%. These metrics significantly surpass the next best-performing models, such as HybViT and SST, which attain OAs of 97.4912% and 97.5318%, respectively. Importantly, ATL-SST maintains this performance consistency across different data splits, achieving perfect accuracy (100%) in multiple scenarios with varying training sample sizes, which underscores its generalization capability. From a computational perspective, ATL-SST achieves a balanced trade-off between accuracy and training efficiency. Its training time of 157.85 seconds is competitive with models like HybViT and SST, while being significantly faster than WaveMamba and PyFormer, which require 1734.50 and 1469.81 seconds, respectively.

Qualitative results in Figure 19 further validate ATL-SST's efficacy. The classification maps produced by ATL-SST exhibit superior boundary delineation and region specificity, closely mirroring the ground truth. In comparison, models such as SS-Mamba and WaveMamba show noticeable misclassifications in regions with complex spectral patterns, while transformer-based models like HybViT and PyFormer, although performing well overall, display minor boundary smoothing. ATL-SST, on the other hand, achieves a fine balance between detail preservation and smoothness, resulting in the most accurate visual outputs.

TABLE XIII: **SA** Comparison of various HSI classification methods highlighting performance metrics.

Samples			State-of-the-art Comparative Methods							ATL-SST
Tr	Va	Te	AGCN	HybViT	SST	WaveFormer	PyFormer	SSMamba	WaveMamba	
41	964	1004	99.5019	99.9003	99.0039	98.5059	100	99.4023	100	100
93	1770	1863	100	100	100	100	100	100	100	100
80	908	988	100	99.3927	99.5951	99.4939	100	97.3684	98.6842	100
62	635	697	96.4131	99.8565	99.7130	100	100	98.4218	99.1391	100
118	1221	1339	86.0343	99.8506	99.7012	99.1784	94.6975	98.6557	98.2076	99.4025
77	1902	1980	100	100	100	100	100	99.8484	99.8484	100
64	1725	1790	99.8882	99.8882	100	100	100	98.8268	99.2178	100
1004	4631	5636	86.1249	93.9850	94.9609	95.2803	88.2540	87.6330	94.2512	98.8644
109	2992	3102	100	100	100	100	99.9677	99.8388	100	99.9677
94	1545	1639	98.1696	99.5118	99.1458	99.7559	93.8987	97.3764	97.6205	99.9389
58	476	534	91.9475	100	98.8764	99.8127	88.9513	96.8164	98.3146	100
56	908	963	97.9231	100	99.5846	99.8961	100	98.6500	99.8961	100
54	404	458	96.7248	100	100	100	96.2882	99.1266	100	100
40	495	535	99.6261	99.6261	100	99.0654	46.1682	94.2056	98.1308	99.8130
842	2792	3634	92.9003	91.2493	90.7264	90.0385	92.4325	76.6373	91.1667	98.2388
50	854	903	99.4462	100	99.6677	100	100	99.6677	99.5570	100
Train (s)			91.0756	147.54	139.82	144.16	1469.81	194.06	1734.50	157.85
κ			94.3561	97.2066	97.2511	97.2427	93.9481	92.8092	96.8405	**99.4281**
OA			94.9233	97.4912	97.5318	97.5244	94.5575	93.5451	97.1623	**99.4864**
AA			96.5437	98.9538	98.8109	98.8142	93.7911	96.4047	98.3771	**99.7641**

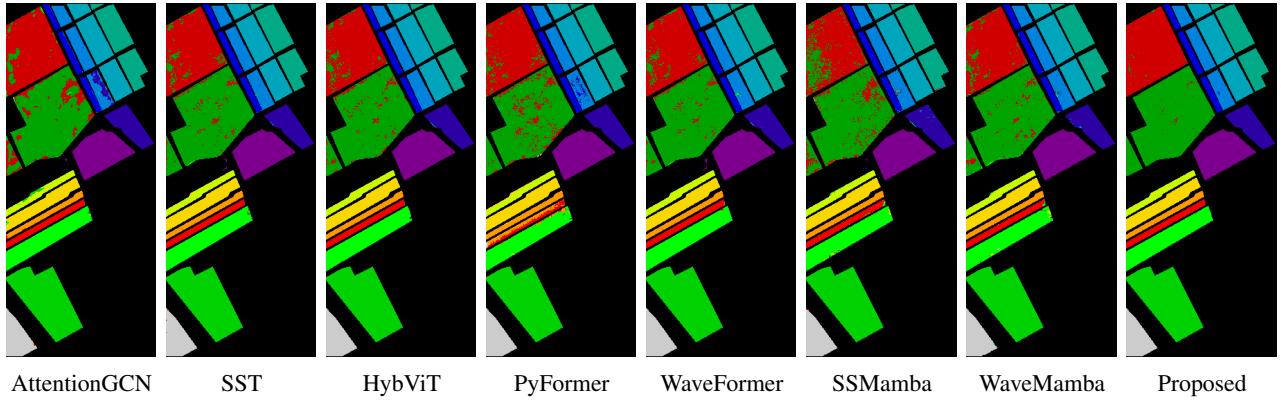| AttentionGCN | SST | HybViT | PyFormer | WaveFormer | SSMamba | WaveMamba | Proposed |

Fig. 19: **SA dataset:** Ground truth maps for comparative methods alongside the proposed ATL-SST.

VIII. CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

This work proposes a novel multi-stage ATL framework that integrates the Spatial-Spectral Transformer (SST) for efficient HSI classification. The proposed framework leverages the strengths of transfer learning, active learning, and spectral-spatial attention mechanisms to address key challenges in HSI classification, such as high spectral dimensionality and limited labeled data. A major contribution of this work is developing an uncertainty-diversity querying mechanism that adaptively selects the most informative and diverse samples for iterative model refinement. This not only optimizes labeling efficiency but also improves the model's ability to generalize across varying spectral profiles. Additionally, this work introduced a dynamic freezing strategy to selectively freeze and unfreeze SST layers during the transfer learning process, ensuring an optimal balance between computational efficiency and adaptability to new spectral variations. This mechanism significantly reduces computational overhead while preserving critical learned representations, making the approach scalable for real-time or resource-constrained environments. Moreover, the proposed framework incorporates a self-calibration mechanism for spectral and spatial attention layers, which further improves the model's robustness.

Extensive experimental validation on benchmark HSI datasets demonstrates that the proposed framework consistently outperforms state-of-the-art models, including CNNs, and baseline SSTs, across key metrics such as classification accuracy, robustness, and computational efficiency. The results confirm the effectiveness of integrating ATL with transformer-based models for HSI classification, showcasing its potential for real-world applications where labeling data is expensive and spectral profiles are complex.

While this study makes substantial progress in advancing HSI classification through active transfer learning and spectral-spatial transformers, several avenues for future exploration remain: for instance, future extensions could focus on semi-supervised or self-supervised learning approaches to reduce reliance on labeled data. This would be particularly beneficial in HSI classification, where obtaining labeled samples is costly. Leveraging self-supervised pretext tasks or pseudo-labeling strategies could further enhance the model's learning capabilities in scenarios with limited labeled data. Although the current uncertainty-diversity-based querying has proven effective, dynamic querying strategies could be explored. For example, combining reinforcement learning with AL may

allow the system to learn optimal sampling strategies over time, potentially improving both sample selection and overall learning efficiency. As HSI data can vary significantly between different sensors or geographical regions, exploring domain adaptation techniques within the ATL framework would be a valuable direction. This would allow models to adapt better to unseen datasets or shifts in data distribution, improving the robustness of the classification process in real-world scenarios.

REFERENCES

[1] M. Ahmad, S. Shabbir, S. K. Roy, D. Hong, X. Wu, J. Yao, A. M. Khan, M. Mazzara, S. Distefano, and J. Chanussot, "Hyperspectral image classification—traditional to deep models: A survey for future prospects," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 968–999, 2022.

[2] F. Ullah, I. Ullah, R. U. Khan, S. Khan, K. Khan, and G. Pau, "Conventional to deep ensemble methods for hyperspectral image classification: A comprehensive survey," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 3878–3916, 2024.

[3] P. Ghamisi, N. Yokoya, J. Li, W. Liao, S. Liu, J. Plaza, B. Rasti, and A. Plaza, "Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art," *IEEE Geoscience and Remote Sensing Magazine*, vol. 5, no. 4, pp. 37–78, 2017.

[4] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geoscience and Remote Sensing Magazine*, vol. 1, no. 2, pp. 6–36, 2013.

[5] M. H. F. Butt, J. P. Li, M. Ahmad, and M. A. F. Butt, "Graph-infused hybrid vision transformer: Advancing geoai for enhanced land cover classification," *International Journal of Applied Earth Observation and Geoinformation*, vol. 129, p. 103773, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1569843224001274

[6] M. B. Stuart, M. Davies, M. J. Hobbs, T. D. Pering, A. J. McGonigle, and J. R. Willmott, "High-resolution hyperspectral imaging using low-cost components: Application within environmental monitoring scenarios," *Sensors*, vol. 22, no. 12, p. 4652, 2022.

[7] Z. Saleem, M. H. Khan, M. Ahmad, A. Sohaib, H. Ayaz, and M. Mazzara, "Prediction of microbial spoilage and shelf-life of bakery products through hyperspectral imaging," *IEEE Access*, vol. 8, pp. 176 986–176 996, 2020.

[8] M. H. Khan, Z. Saleem, M. Ahmad, A. Sohaib, H. Ayaz, M. Mazzara, and R. A. Raza, "Hyperspectral imaging-based unsupervised adulterated red chili content transformation for classification: Identification of red chili adulterants," *Neural Computing and Applications*, vol. 33, no. 21, pp. 14 507–14 521, 2021.

[9] M. H. Khan, Z. Saleem, M. Ahmad, A. Sohaib, H. Ayaz, and M. Mazzara, "Hyperspectral imaging for color adulteration detection in red chili," *Applied Sciences*, vol. 10, no. 17, p. 5955, 2020.

[10] H. Ayaz, M. Ahmad, A. Sohaib, M. N. Yasir, M. A. Zaidan, M. Ali, M. H. Khan, and Z. Saleem, "Myoglobin-based classification of minced meat using hyperspectral imaging," *Applied Sciences*, vol. 10, no. 19, p. 6862, 2020.

[11] H. Ayaz, M. Ahmad, M. Mazzara, and A. Sohaib, "Hyperspectral imaging for minced meat classification using nonlinear deep features," *Applied Sciences*, vol. 10, no. 21, p. 7783, 2020.

[12] M. H. F. Butt, H. Ayaz, M. Ahmad, J. P. Li, and R. Kuleev, "A fast and compact hybrid cnn for hyperspectral imaging-based bloodstain classification," in *2022 IEEE Congress on Evolutionary Computation (CEC)*, 2022, pp. 1–8.

[13] M. Zulfiqar, M. Ahmad, A. Sohaib, M. Mazzara, and S. Distefano, "Hyperspectral imaging for bloodstain identification," *Sensors*, vol. 21, no. 9, p. 3045, 2021.

[14] M. H. F. Butt, J. P. Li, J. C. Ji, W. Riaz, N. Anwar, F. F. Butt, M. Ahmad, A. Saboor, A. Ali, and M. Y. Uddin, "Intelligent tumor tissue classification for hybrid health care units," *Frontiers in Medicine*, vol. 11, 2024. [Online]. Available: https://www.frontiersin.org/journals/medicine/articles/10.3389/fmed.2024.1385524

[15] N. Wambugu, Y. Chen, Z. Xiao, K. Tan, M. Wei, X. Liu, and J. Li, "Hyperspectral image classification on insufficient-sample and feature learning using deep neural networks: A review," *International Journal of Applied Earth Observation and Geoinformation*, vol. 105, p. 102603, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S030324342100310X

[16] S. Hajaj, A. El Harti, A. B. Pour, A. Jellouli, Z. Adiri, and M. Hashim, "A review on hyperspectral imagery application for lithological mapping and mineral prospecting: Machine learning techniques and future prospects," *Remote Sensing Applications: Society and Environment*, vol. 35, p. 101218, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S235293852400082X

[17] M. F. Guerri, C. Distante, P. Spagnolo, F. Bougourzi, and A. Taleb-Ahmed, "Deep learning techniques for hyperspectral image analysis in agriculture: A review," *ISPRS Open Journal of Photogrammetry and Remote Sensing*, vol. 12, p. 100062, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2667393322400005X

[18] M. Ahmad, M. H. F. Butt, M. Mazzara, S. Distefano, A. M. Khan, and H. A. Altuwaijri, "Pyramid hierarchical spatial-spectral transformer for hyperspectral image classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 17 681–17 689, 2024.

[19] Z. Zhao, X. Xu, S. Li, and A. Plaza, "Hyperspectral image classification using groupwise separable convolutional vision transformer network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–17, 2024.

[20] M. Ahmad, M. Usama, A. M. Khan, S. Distefano, H. A. Altuwaijri, and M. Mazzara, "Spatial–spectral transformer with conditional position encoding for hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 21, pp. 1–5, 2024.

[21] F. Xu, S. Mei, G. Zhang, N. Wang, and Q. Du, "Bridging cnn and transformer with cross-attention fusion network for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–14, 2024.

[22] M. Ahmad, U. Ghous, M. Usama, and M. Mazzara, "Waveformer: Spectral–spatial wavelet transformer for hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 21, pp. 1–5, 2024.

[23] M. Jiang, Y. Su, L. Gao, A. Plaza, X.-L. Zhao, X. Sun, and G. Liu, "Graphgst: Graph generative structure-aware transformer for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–16, 2024.

[24] J. Li, Z. Zhang, R. Song, Y. Li, and Q. Du, "Scformer: Spectral coordinate transformer for cross-domain few-shot hyperspectral image classification," *IEEE Transactions on Image Processing*, vol. 33, pp. 840–855, 2024.

[25] M. Ahmad, M. Usama, M. Mazzara, S. Distefano, H. A. Altuwaijri, and S. L. Ullo, "Fusing transformers in a tuning fork structure for hyperspectral image classification across disjoint samples," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 18 167–18 181, 2024.

[26] D. Hong, Z. Han, J. Yao, L. Gao, B. Zhang, A. Plaza, and J. Chanussot, "Spectralformer: Rethinking hyperspectral image classification with transformers," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[27] J. Chen, C. Yang, L. Zhang, L. Yang, L. Bian, Z. Luo, and J. Wang, "Tccu-net: Transformer and cnn collaborative unmixing network for hyperspectral image," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 8073–8089, 2024.

[28] C. Shi, S. Yue, and L. Wang, "A dual-branch multiscale transformer network for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–20, 2024.

[29] L. Zhang, Y. Wang, L. Yang, J. Chen, Z. Liu, J. Wang, L. Bian, and C. Yang, "A multi-range spectral-spatial transformer for hyperspectral image classification," *Infrared Physics & Technology*, vol. 135, p. 104983, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1350449523004413

[30] M. Ye, J. Chen, F. Xiong, and Y. Qian, "Adaptive graph modeling with self-training for heterogeneous cross-scene hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, 2024.

[31] W. Huang, Y. Deng, S. Hui, Y. Wu, S. Zhou, and J. Wang, "Sparse self-attention transformer for image inpainting," *Pattern Recognition*, vol. 145, p. 109897, 2024.

[32] Y. Sun, X. Zhi, S. Jiang, G. Fan, X. Yan, and W. Zhang, "Image fusion for the novelty rotating synthetic aperture system based on vision transformer," *Information Fusion*, vol. 104, p. 102163, 2024.

[33] Y. Peng, Y. Zhang, B. Tu, Q. Li, and W. Li, "Spatial–spectral transformer with cross-attention for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[34] S. Jia, S. Jiang, Z. Lin, N. Li, M. Xu, and S. Yu, "A survey: Deep learning for hyperspectral image classification with few labeled samples," *Neurocomputing*, vol. 448, pp. 179–204, 2021.

[Online]. Available: https://www.sciencedirect.com/science/article/pii/S09252312210004033

[35] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B. B. Gupta, X. Chen, and X. Wang, "A survey of deep active learning," *ACM Comput. Surv.*, vol. 54, no. 9, Oct. 2021. [Online]. Available: https://doi.org/10.1145/3472291

[36] M. Ahmad, S. Shabbir, D. Oliva, M. Mazzara, and S. Distefano, "Spatial-prior generalized fuzziness extreme learning machine autoencoder-based active learning for hyperspectral image classification," *Optik*, vol. 206, p. 163712, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0030402619316109

[37] H. Wu, Y. Zhong, G. Han, J. Lin, Z. Liu, and C. Han, "Dbral: A novel uncertainty-based active learning based on deep-broad learning for medical image classification," in *Artificial Neural Networks and Machine Learning – ICANN 2024: 33rd International Conference on Artificial Neural Networks, Lugano, Switzerland, September 17–20, 2024, Proceedings, Part VIII*. Berlin, Heidelberg: Springer-Verlag, 2024, p. 260–273. [Online]. Available: https://doi.org/10.1007/978-3-031-72353-7_19

[38] R. Thoreau, V. Achard, L. Risser, B. Berthelot, and X. Briottet, "Active learning for hyperspectral image classification: A comparative review," *IEEE Geoscience and Remote Sensing Magazine*, vol. 10, no. 3, pp. 256–278, 2022.

[39] C. Deng, Y. Xue, X. Liu, C. Li, and D. Tao, "Active transfer learning network: A unified deep joint spectral–spatial feature learning model for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 3, pp. 1741–1754, 2019.

[40] M. Ahmad, M. Usama, S. Distefano, and M. Mazzara, "Hyperspectral image classification with fuzzy spatial-spectral class discriminate information," in *2024 IEEE International Conference on Image Processing (ICIP)*, 2024, pp. 2285–2291.

[41] M. Ahmad, U. Ghous, D. Hong, A. M. Khan, J. Yao, S. Wang, and J. Chanussot, "A disjoint samples-based 3d-cnn with active transfer learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2022.

[42] X. Cao, J. Yao, Z. Xu, and D. Meng, "Hyperspectral image classification with convolutional neural network and active learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 7, pp. 4604–4616, 2020.

[43] S. Liu, Q. Shi, and L. Zhang, "Few-shot hyperspectral image classification with unknown classes using multitask deep learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 6, pp. 5085–5102, 2021.

[44] X. Liao, B. Tu, J. Li, and A. Plaza, "Class-wise graph embedding-based active learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1–13, 2023.

[45] C. Zhao, B. Qin, S. Feng, W. Zhu, W. Sun, W. Li, and X. Jia, "Hyperspectral image classification with multi-attention transformer and adaptive superpixel segmentation-based active learning," *IEEE Transactions on Image Processing*, vol. 32, pp. 3606–3621, 2023.

[46] Z. Wang, Z. Chen, and B. Du, "Active learning with co-auxiliary learning and multi-level diversity for image classification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 8, pp. 3899–3911, 2023.

[47] F. Liu, W. Gao, J. Liu, X. Tang, and L. Xiao, "Adversarial domain alignment with contrastive learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1–20, 2023.

[48] F. Deng, R. Liang, W. Luo, and G. Zhang, "Deep learning segmentation of seismic facies based on proximity constraint strategy: Innovative application of uma-net model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–17, 2024.

[49] H. wu, S. Wu, K. Zhang, X. Liu, S. Shi, and C. Bian, "Unsupervised blind spectral–spatial cross-super-resolution network for hsi and msi fusion," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–14, 2024.

[50] M. Ahmad, M. Usama, M. Mazzara, and S. Distefano, "Wavemamba: Spatial-spectral wavelet mamba for hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters*, pp. 1–1, 2024.