Pre-trained Deep Learning Networks for Analyzing Hyperspectral Image Data

Content

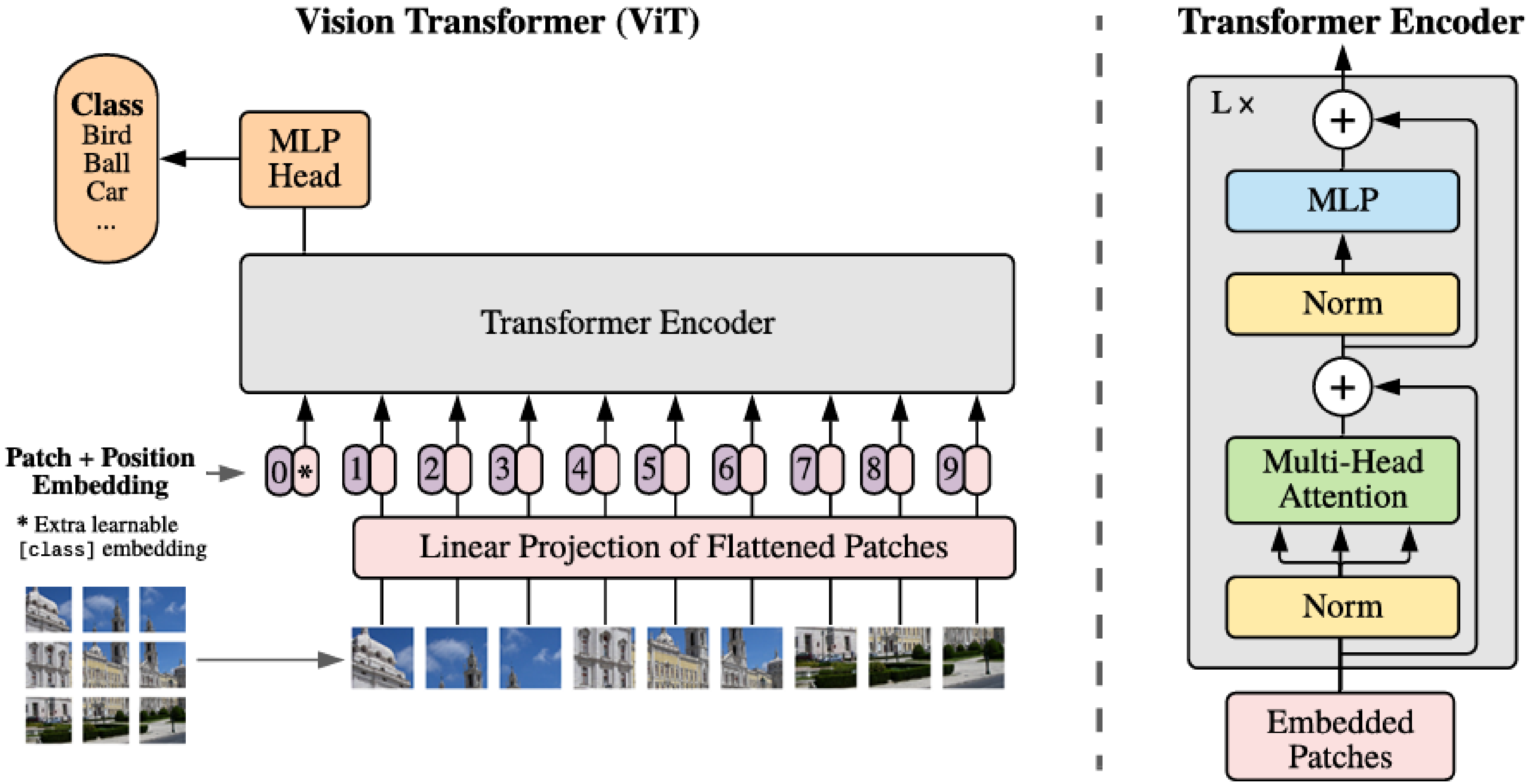
- Introduction
- Research Papers
- Project Timeline
- References

1. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale (VIT) (Vision Transformer)

Goal
Apply the Transformer architecture (originally for NLP) to image classification

Working

- Patch Splitting
    - - Input image is divided into fixed-size patches (e.g. 16×16).
    - - Each patch is flattened and linearly embedded.
- [CLS] Token + Positional Embedding
    - - A learnable [CLS] token is prepended to the patch sequence.
    - - Positional embeddings are added to retain spatial order.
- Transformer Encoder
    - - Sequence is passed through multi-head self-attention layers.
    - - [CLS] token gathers global information via attention.
- Classification Head
    - - Final [CLS] embedding is fed into a fully connected layer.
    - - Output: class probabilities via softmax.



	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** $\pm$ 0.04	87.76 $\pm$ 0.03	85.30 $\pm$ 0.02	87.54 $\pm$ 0.02	88.4/88.5*
ImageNet ReaL	**90.72** $\pm$ 0.05	90.54 $\pm$ 0.03	88.62 $\pm$ 0.05	90.54	90.55
CIFAR-10	**99.50** $\pm$ 0.06	99.42 $\pm$ 0.03	99.15 $\pm$ 0.03	99.37 $\pm$ 0.06	–
CIFAR-100	**94.55** $\pm$ 0.04	93.90 $\pm$ 0.05	93.25 $\pm$ 0.05	93.51 $\pm$ 0.08	–
Oxford-IIIT Pets	**97.56** $\pm$ 0.03	97.32 $\pm$ 0.11	94.67 $\pm$ 0.15	96.62 $\pm$ 0.23	–
Oxford Flowers-102	99.68 $\pm$ 0.02	**99.74** $\pm$ 0.00	99.61 $\pm$ 0.02	99.63 $\pm$ 0.03	–
VTAB (19 tasks)	**77.63** $\pm$ 0.23	76.28 $\pm$ 0.46	72.72 $\pm$ 0.21	76.29 $\pm$ 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

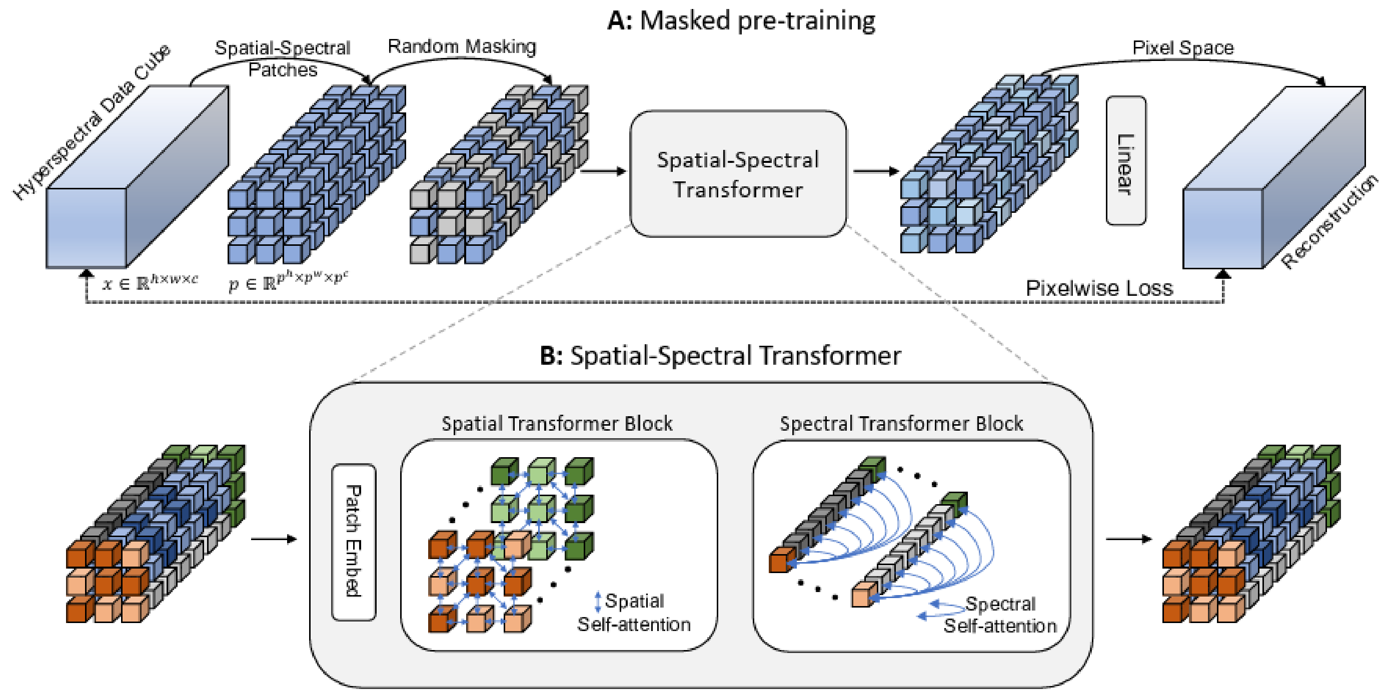
2. Masked Vision Transformers for Hyperspectral Image Classification

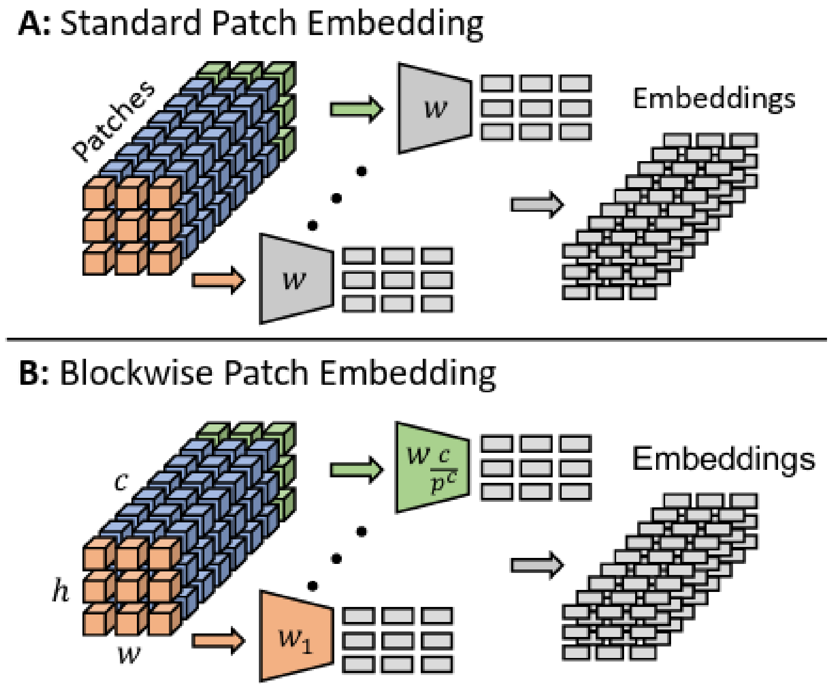
Goal

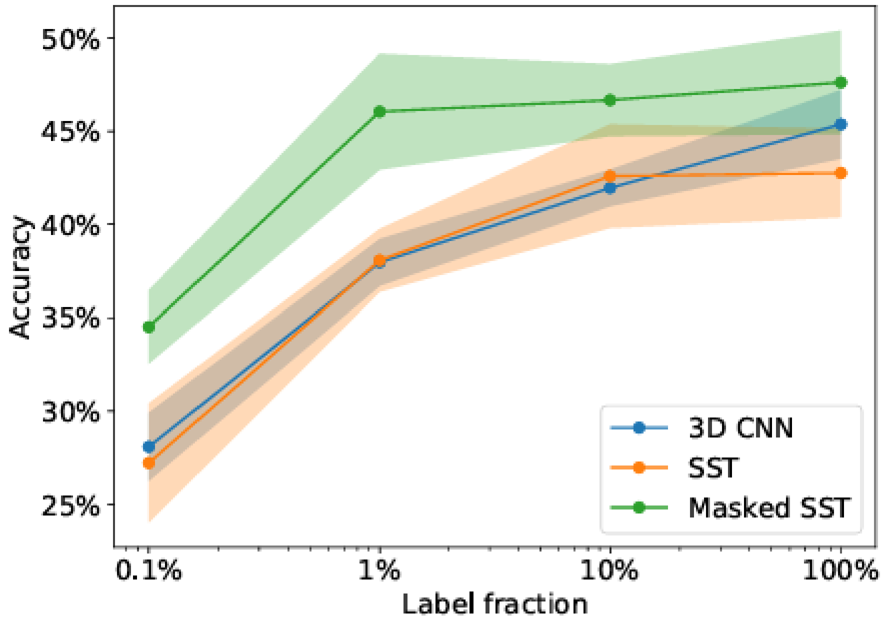
Leverage self-supervised masked image modeling to pretrain transformers on unlabeled hyperspectral data

Working

- Patch Extraction
    - Divide hyperspectral cubes into spatial–spectral patches (e.g. 8×8×bands).
    - Apply random masking to hide some patches.
- Blockwise Patch Embedding
    - Spectral bands grouped into blocks; each block gets its own embedding layer.
    - Preserves wavelength-specific characteristics.
- Spatial–Spectral Factorized Attention
    - Alternates between spatial and spectral self-attention.
    - Efficiently models global context across space and spectrum.
- Masked Pretraining
    - Train transformer to reconstruct masked patches.
    - Learns rich representations without labels.
- Fine-Tuning
    - Attach classification head.
    - Use small labeled datasets (e.g. Houston2018) for supervised training.







Houston2018 dataset.

3. Contrastive Learning Based on Transformer for Hyperspectral Image Classification

1. Create two "views" of each cube

For each 27×27×N patch:

• Flip it (horizontally or vertically).

• Randomly erase (zero-out) either some individual pixels or one small rectangle—but never the center pixel.

That gives you View A and View B of the same original patch.

2. Two networks: Online vs. Target

Both networks have the same shape, but different weights:

Encoder (a tiny 2-layer Vision Transformer)

Projector (a small 2-layer feed-forward head)

Predictor (another small head, but only in the Online network)

3. Forward pass & loss

Feed View A into the **Online** network → get prediction vector p_1.

Feed View B into the **Target** network → get projection vector z_2.

Compare p_1 and z_2 with a simple "make-them-close" loss (cosine similarity).

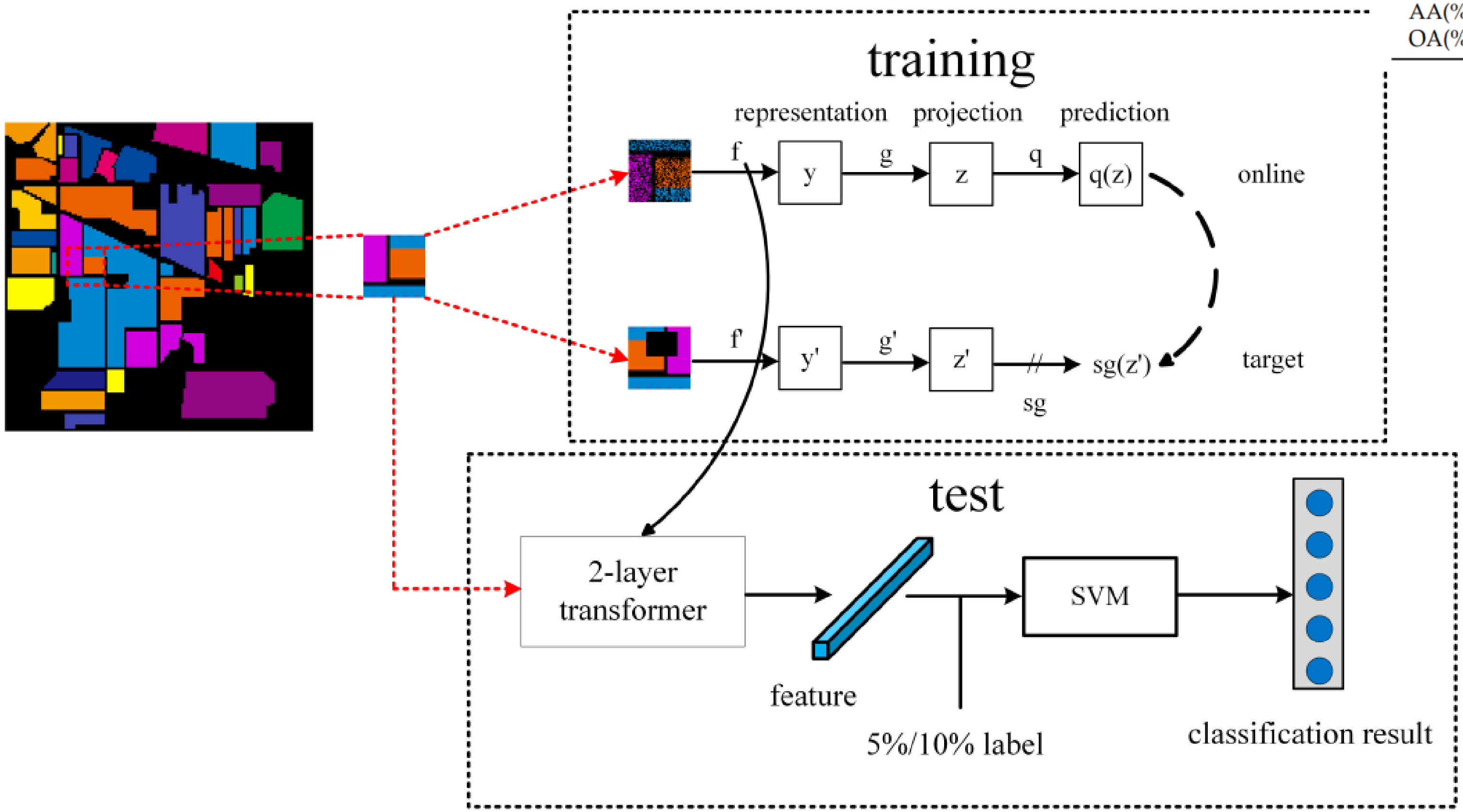
Swap roles (View B→Online, View A→Target) and compute the same loss again.

Total loss = $Loss_1 + Loss_2$.

Class	Supervised Feature Extraction			Unsupervised Feature Extraction			
	LDA	1D-CNN	S-CNN	3DCAE	AAE	VAE	Proposed
1	58.54	43.33	83.33	90.48	**100.00**	**100.00**	97.05
2	69.88	73.13	81.41	92.49	81.63	78.78	**96.73**
3	65.86	65.52	74.02	90.37	95.27	92.37	**95.34**
4	73.71	51.31	71.49	86.90	**99.22**	97.34	98.97
5	90.32	87.70	90.11	94.25	**95.17**	93.87	94.82
6	92.09	95.10	94.06	97.07	**98.73**	98.27	93.85
7	96.00	56.92	84.61	91.26	96.00	**98.67**	95.00
8	98.14	96.64	98.37	97.79	**99.84**	99.77	98.85
9	11.11	28.89	33.33	75.91	96.30	98.15	88.88
10	73.80	75.12	86.05	87.34	87.01	78.86	**95.65**
11	55.41	83.49	82.98	90.24	89.08	81.75	**98.56**
12	76.92	67.55	73.40	**95.76**	93.51	90.64	94.31
13	91.30	96.86	87.02	97.49	98.56	98.56	89.37
14	93.32	96.51	94.38	96.03	95.73	93.24	**98.44**
15	67.72	39.08	75.57	90.48	97.31	97.02	**99.70**
16	90.36	89.40	79.76	98.82	98.02	**98.81**	94.73
AA(%)	76.89	71.66	84.44	92.04	95.09	93.51	**95.64**
OA(%)	76.88	79.66	80.72	92.35	91.80	88.03	**96.78**

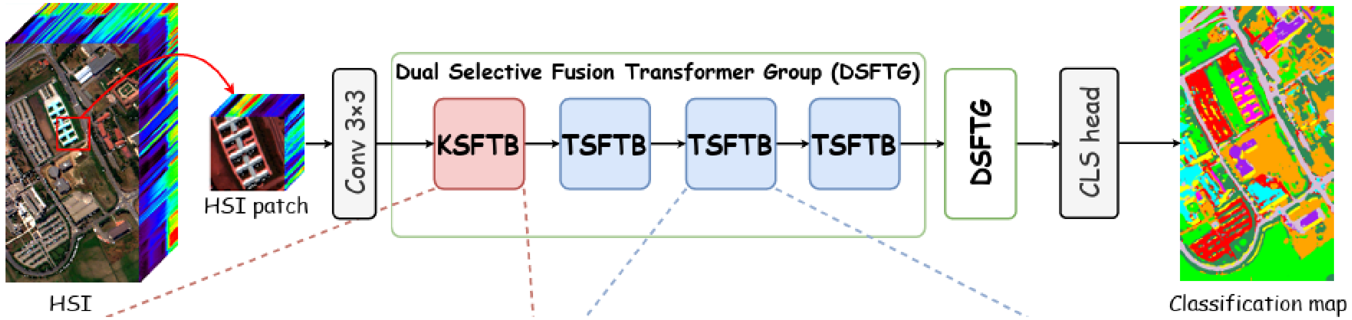
Extract features and classify

- After training keep only the transformer encoder
- Run every pixel patch through it to get a feature vector
- Train a simple SVM on just 5-10% of labeled pixels in that feature space
- Use the SVM to label the rest of your image

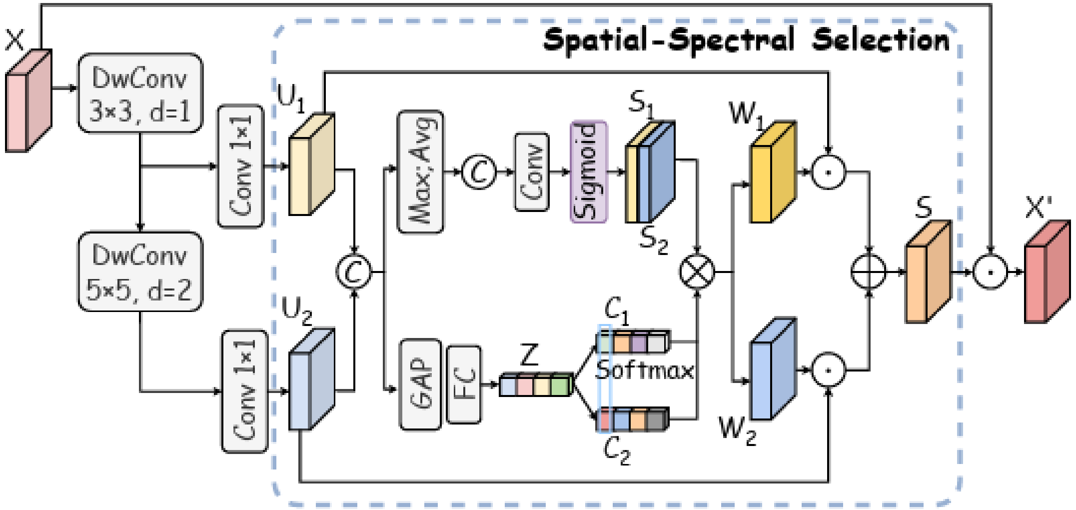


4. Dual Selective Fusion Transformer Network for Hyperspectral Image Classification

Goal : Instead of treating every feature equally, DSFormer focuses only on what truly matters — selecting the best scales and features for each scene.

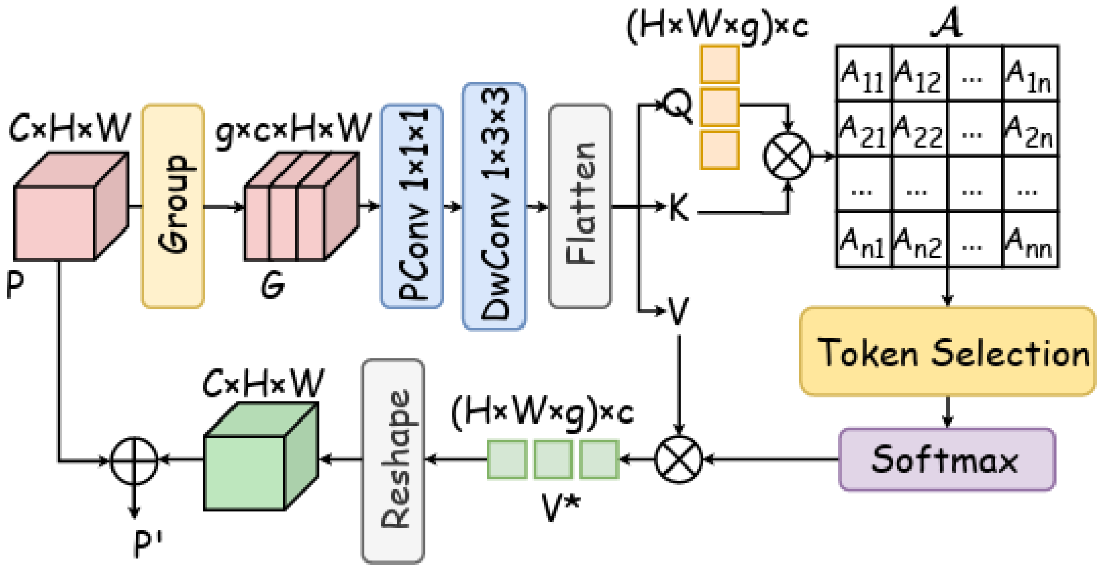


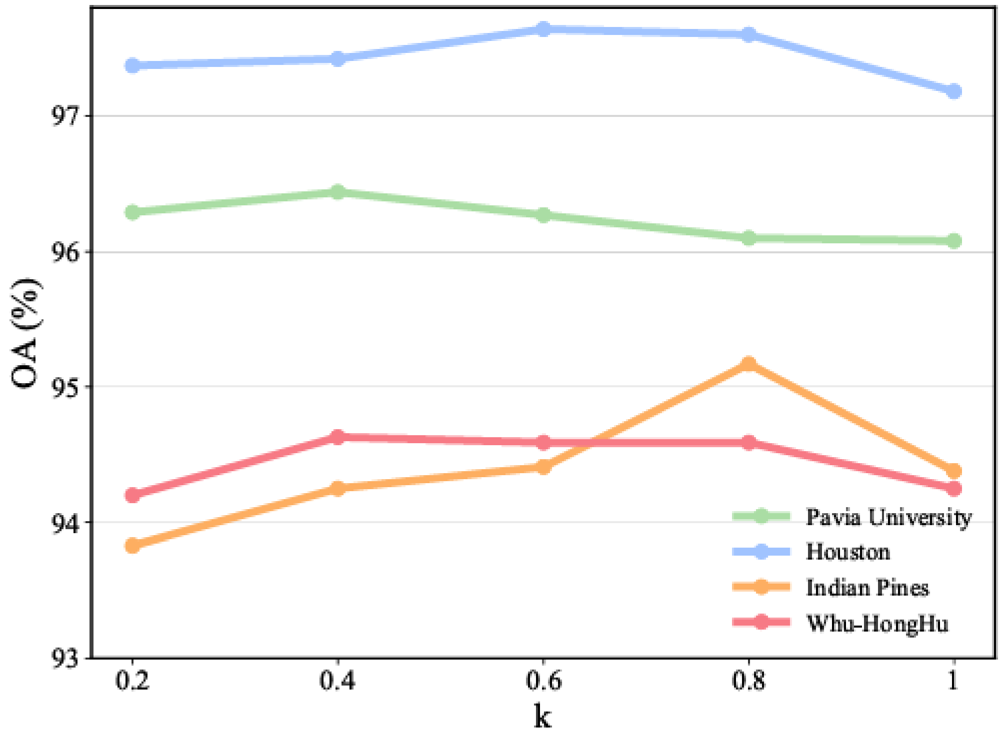
KSFTB (Kernel Selective Fusion Attention)
KSFTB is trying to decide **how much to "zoom in or out"** when looking at a pixel in a hyperspectral image



(TSFA) Token Selective Fusion Transformer Block

TSFTB lets the model **focus only on the most relevant tokens**, skipping the distractions





General Observations

- With lots of data, self-supervised or contrastive pre-training from scratch often yields superior representations.
- Transformers outperform CNNs, But require significantly more memory.
- Contrastive learning yield better results and provide strong unsupervised pre-training.

Proposed method

- Contrastive Pretraining with CNNs: Use a CNN backbone with SimCLR-style contrastive learning to learn general HSI features from unlabeled patches through augmentations.
- Cross-Domain Generalization: Train on HSI data from multiple domains (different sensors or regions) to create a model that captures domain-invariant spectral-spatial patterns.
- Supervised Fine-tuning: Fine-tune the pretrained model on labeled data from a target domain to achieve high classification accuracy with minimal extra training.

Project Phase	Jun-25		Jul-25				Aug-25				Sep-25				Oct-25				Nov-25				Dec	
	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2
Literature review																								
Research Paper Gathering																								
Summarizing LR																								
Thesis Proposal																								
Deciding Implementation																								
Implementation																								
Test and Finalize																								
Evaluation of Performance																								
Report Writing																								
Proof Reading/ Changes																								
Review and Submission																								

References

1. https://arxiv.org/abs/2010.11929

2. L. Scheibenreif, M. Mommert and D. Borth, "Masked Vision Transformers for Hyperspectral Image Classification," 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Vancouver, BC, Canada, 2023, pp. 2166-2176, doi: 10.1109/CVPRW59228.2023.00210. keywords: {Solid modeling;Three-dimensional displays;Training data;Computer architecture;Transformers;Data models;Convolutional neural networks},

3. https://doi.org/10.3390/app11188670
4. https://arxiv.org/abs/2410.03171

Thank you