Hyperspectral Image Classification on Indian Pines and Adaptation to Salinas and Pavia University Using 2D CNNs

Jaykumar Darji (018180851)[a]

[a]*Department of Electrical Engineering, San Jose State University, San Jose, CA*

Abstract

Hyperspectral image classification is challenging because spectral signatures and class structure vary strongly across scenes, which makes model transfer difficult and motivates the need for clear and reproducible workflows. This study builds a full pipeline in Jupyter for data cleaning, band removal, z score normalization, stratified splits and consistent evaluation, and applies it across spectral networks, spatial 2D CNNs, 3D CNNs and classical baselines. The 2D CNN trained on Indian Pines achieved the best source domain performance, with validation accuracy of 1.0, validation kappa of 1.0 and validation macro F1 of 1.0, and a test accuracy of about 0.9927, test kappa of about 0.9916 and test macro F1 of about 0.9919. This model was then used as the backbone for transfer to Salinas and PaviaU. Simple adapters that only replaced the first and last layers did not transfer well, with unstable warm up accuracy on both scenes. Stable transfer required a deeper fine tune that unfroze part of the backbone after adapter training. With this strategy, Salinas reached a final accuracy of about 0.9578 and a macro F1 of about 0.9147, with class performance varying from near perfect scores on major classes to clear drops on classes with strong spectral shift. PaviaU reached a final accuracy of about 0.9977 and a macro F1 of about 0.9901, with most classes showing precision and recall near 1.0. The findings show that spatial CNNs remain the strongest option for hyperspectral classification and that cross scene transfer is feasible when simple adapters are paired with a short and targeted fine tune of deeper layers.

Keywords: Hyperspectral image classification, 2D CNN, Transfer learning, Cross scene adaptation, Fine tuning, Indian Pines, Salinas, Pavia University

1. Introduction

Hyperspectral images (HSIs) contain hundreds of spectral bands that offer rich and detailed spectral information. Because of this, HSIs are used in many areas such as precision agriculture [1], resource surveys [2], target identification [3], and landscape classification [4]. Visual classification is important for understanding HSI scenes [5, 6]. Even so, HSI classification remains difficult because of high dimensionality, strong nonlinearity, and the limited number of labeled samples [7, 8].

To address these issues, dimensionality reduction (DR) [9, 10, 11, 12] and semi supervised classification methods [13, 14] are commonly used. DR methods include band selection and feature extraction [15]. Feature extraction [16, 17, 18, 19] reduces data size by projecting high dimensional data into a lower dimensional space. Feature selection

[20] chooses a subset of important spectral bands. Sparse based methods [21] have also been studied. PCA finds orthogonal directions that keep most of the information in HSIs [22, 23]. It often reduces spectral dimensions by more than eighty percent and improves class separability.

Recent work in remote sensing has focused on deep learning for HSI classification. The fine spectral detail in HSI data makes it hard for traditional models to separate similar land cover types and to capture spatial patterns [24]. Deep learning has shown clear benefits in many areas such as digital image processing, pattern recognition, segmentation, classification, and object detection [25]. Many studies have used deep learning to improve HSI analysis [26]. Examples include dual path networks that mix dense and residual connections [27], unsupervised layer wise training for pixel block pairs [28], deep feature fusion networks [29], and CNN based models [30]. Li et al. [31] used 3D CNNs for feature extraction, while Mou et al. [32] applied a deep residual model for unsupervised HSI classification.

In this work, we address the challenge of transferring a pretrained HSI classifier across scenes that have different spectral band configurations. Instead of training a new model for every dataset, we introduce a light one by one spectral adapter that maps the target scene bands into the spectral space expected by the pretrained backbone. This adapter allows the network to handle band mismatch while keeping the spatial extraction learned during pretraining. We first train the adapter alone while keeping the CNN backbone frozen. This warm up stage learns the new spectral statistics with very low cost.

After the adapter converges, we unfreeze the last convolution block and the classifier head for a short fine tuning stage. This two stage training strategy lets the model refine both spectral mixing and high level spatial features without overfitting the small labeled sets common in HSI datasets. The method is simple, efficient, and improves cross scene performance compared to direct transfer or adapter only training.

2. Related Work

CNNs have been used widely for HSI classification. Zhang et al. [33, 34] applied a CNN to extract spatial features using the first few principal components of the HSI data. A 2D CNN works well in many image tasks such as object detection [35] and classification [36]. Even so, applying a 2D CNN to HSIs means convolving each spectral band with separate kernels. This increases parameters, the risk of overfitting, and computation.

Although 2D CNNs perform well in tasks like classification [37], face detection [38], depth estimation [35, 39], and object detection [40], they cannot capture spectral relations between channels. This makes it harder to extract key spectral features from HSIs and often lowers accuracy.

HSIs contain important spatial details that help identify objects, while spectral information carries unique signatures for land cover [41]. Because of this, many studies combine spectral and spatial cues. Ben Hamida et al. [42] explored 3D CNNs for HSI data. Zhong et al. [43] designed a 3D deep learning model for spectral and spatial extraction. Mei et al. [44] also used 3D CNNs and showed improved accuracy. Li et al. [31] used small spatial windows with 3D input cubes to process raw HSI data. These models work well, but deeper 3D CNNs often show reduced precision as depth increases.

Li et al. [45] noted that HSI bands contain rich spectral detail but also high redundancy because adjacent bands are correlated [46]. Reducing bands while keeping key features is important [47]. Since HSIs are 3D cubes, 3D convolutions can extract spectral and spatial features at the same time. Many works use this idea [18, 30, 31, 36, 42, 43, 48, 49]. Several studies such as Song et al. [29], Mou et al. [32], Zhong et al. [43], and Paoletti et al. [50] apply residual learning to improve feature discrimination. Spatial feature fusion has also shown clear improvements [51]. Even though 3D CNNs can achieve strong accuracy, they are expensive to train and still struggle when used alone.

Melgani and Bruzzone [52] used SVMs to study different classifiers. Makantasis et al. [9] built a deep learning method that learns spatial and spectral features using a 3D model followed by a multilayer perceptron. This reduced noise in remote sensing data but focused mainly on spatial cues. A multiscale 3D CNN (M3D DCNN) [53] was later introduced to learn 2D spatial features and 1D spectral features together. However, it lacks strong feature fusion.

Zhong et al. proposed SSRN [43], a spectral spatial residual network that uses residual blocks. HybridSN by Roy et al. [54] achieved even higher accuracy by combining 3D and 2D convolutions. Although effective, HybridSN uses many parameters and has long training time.

Our method follows the general transfer learning strategy used in cross scene HSI adaptation but replaces traditional dimensional alignment with a lightweight one by one spectral adapter placed before a pretrained 2D CNN backbone. The backbone have three spatial convolution blocks with increasing depth (32, 64, and 128 filters) and an adaptive pooling classifier head. During transfer, we freeze the early convolution blocks to retain the pretrained spatial representations and update only the adapter, the final convolution block, and the classifier head. This staged optimization allows the network to efficiently absorb the spectral characteristics of the new scene while preserving the robust spatial features learned from the source dataset.Experiments show that our model improves efficiency, accuracy, and representation quality compared to SVM [52], 2D CNN [19], 3D CNN [42], M3D CNN [54], SSRN [43], and HybridSN [55].

3. Materials and Methods used

This study uses three public hyperspectral scenes. Indian Pines serves as the source domain for all model development and pretraining. Salinas and Pavia University are used as target domains for cross scene transfer tests. Each cube is cleaned by removing known bad or water absorption bands, and all spectral values are normalized with z score scaling. Spatial patch extraction is used for models that require neighborhood context, with a patch size of fifteen.

3.1. *Data Acquisition*

All datasets used in this study come from the public hyperspectral repository curated by M Graña, MA Veganzones and B Ayerdi, which provides airborne and satellite scenes widely used in hyperspectral image classification research.

3.1.1. *Indian Pines*

Indian Pines [56] was recorded by the AVIRIS sensor over farmland in north western Indiana. The image size is 145×145 pixels with 224 spectral bands in the 0.4 to 2.5 micrometer range. Following common practice, the water

Table 1: Class distribution for the Indian Pines dataset.

#	Class	Samples
1	Alfalfa	46
2	Corn-notill	1428
3	Corn-mintill	830
4	Corn	237
5	Grass-pasture	483
6	Grass-trees	730
7	Grass-pasture-mowed	28
8	Hay-windrowed	478
9	Oats	20
10	Soybean-notill	972
11	Soybean-mintill	2455
12	Soybean-clean	593
13	Wheat	205
14	Woods	1265
15	Buildings-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93

absorption and low signal bands [104–108], [150–163] and 220 are removed, leaving 200 bands for analysis. The scene contains 16 labeled classes, mainly different crop types and small regions of built surfaces and vegetation. The distribution of samples per class is summarized in Table 1.

3.1.2. Salinas

The Salinas scene [56] was also acquired by AVIRIS, covering agricultural fields in California. It has spatial dimensions of 512 × 217 pixels and originally 224 bands. After removing the water absorption ranges [108–112], [154–167] and 224, there are 204 usable bands. The dataset defines 16 classes that include vegetables, vineyard structures and soil types. The scene has high spatial resolution and clear class boundaries, which makes it suitable for testing spatial models and transfer learning. The class distribution is shown in Table 2.

3.1.3. Pavia University

Pavia University [56] was captured by the ROSIS sensor during a flight over an urban area in northern Italy. The image size is 610 × 610 pixels, although empty areas at the edges are usually removed before processing. After dropping the noisy spectral bands, 103 usable bands remain. The scene has 9 ground truth classes that include asphalt, meadows, trees, bare soil, gravel, metal sheets, bricks and shadow regions. The spatial resolution is 1.3 meters, and the strong mix of man made surfaces and natural areas makes it a good test for evaluating generalization of classification

Table 2: Ground truth classes for the Salinas scene and their respective sample numbers.

#	Class	Samples
1	Brocoli_green_weeds_1	2009
2	Brocoli_green_weeds_2	3726
3	Fallow	1976
4	Fallow_rough_plow	1394
5	Fallow_smooth	2678
6	Stubble	3959
7	Celery	3579
8	Grapes_untrained	11271
9	Soil_vinyard_develop	6203
10	Corn_senesced_green_weeds	3278
11	Lettuce_romaine_4wk	1068
12	Lettuce_romaine_5wk	1927
13	Lettuce_romaine_6wk	916
14	Lettuce_romaine_7wk	1070
15	Vinyard_untrained	7268
16	Vinyard_vertical_trellis	1807

Table 3: Ground truth classes and sample numbers for the Pavia Centre and Pavia University scenes.

Pavia Centre			Pavia University		
#	Class	Samples	#	Class	Samples
1	Water	824	1	Asphalt	6631
2	Trees	820	2	Meadows	18649
3	Asphalt	816	3	Gravel	2099
4	Self Blocking Bricks	808	4	Trees	3064
5	Bitumen	808	5	Painted metal sheets	1345
6	Tiles	1260	6	Bare Soil	5029
7	Shadows	476	7	Bitumen	1330
8	Meadows	824	8	Self Blocking Bricks	3682
9	Bare Soil	820	9	Shadows	947

models. A summary of the classes for both the Pavia Centre and Pavia University scenes is given in Table 3.

3.2. *Data Pre-processing*

The three hyperspectral scenes differ in band count, spatial size and noise level, so a consistent preprocessing pipeline is applied to all datasets before model training. The steps include band removal, cleaning, normalization, patch extraction and label handling. Each step is designed to match the requirements of both spectral and spatial models.

3.2.1. *Band removal*

Bands affected by water absorption or low signal are removed because they add noise and do not contribute useful Information.

- For Indian Pines, the dropped bands are [104–108], [150–163], and 220.

- For Salinas, the same absorption regions [108–112], [154–167], and 224 are removed.

- For Pavia University, only the noisy ROSIS bands are excluded, leaving 103 valid bands.

3.2.2. *Z score normalization*

Each spectral band is normalized with a standard z score transform. This keeps the inputs stable during training and avoids scale differences across bands [57]. For a pixel value x in band b, the normalized value is

$$z = \frac{x - \mu_b}{\sigma_b},$$

where μ_b is the mean of band b and σ_b is the standard deviation of band b. The normalization parameters are computed only on the training split and applied to the validation and test splits to prevent information leakage.

3.2.3. *Stratified splitting*

To preserve the natural class distribution present in each scene, the data are divided into training, validation and test subsets using stratified sampling. This procedure ensures that the proportion of each class remains consistent across all splits. Stratification is especially important for hyperspectral datasets, since many classes contain only a small number of labeled pixels. Maintaining these proportions avoids bias during training and allows fair comparison across models, because each model is evaluated on the same balanced representation of the scene.

3.2.4. *Patch extraction*

Spatial models such as two–dimensional CNNs and three–dimensional CNNs require local neighborhoods around each labeled pixel to capture spatial structure. To provide this information, a square patch of fixed size is extracted from the hyperspectral cube. In this work, a patch size of 15×15 is used. For a labeled pixel at spatial position (i, j), the corresponding patch is given by

$$\text{patch}(i, j) = I[i - r : i + r, \ j - r : j + r, \ :],$$

6

where *I* is the hyperspectral image and

$$r = \frac{15 - 1}{2} = 7.$$

This radius defines the extent of the neighborhood around the center pixel. Only patches that lie fully inside the spatial bounds of the scene are retained to avoid padding or incomplete context. The extracted patch preserves the full set of spectral bands for all pixels within the spatial window, which allows spatial and spectral features to be learned jointly.

3.3. Feature engineering

Several feature engineering strategies were tested to explore whether band selection or dimensionality reduction could improve model performance. Two approaches were considered. The first was ANOVA [58] based band ranking, which scores each spectral band by measuring how well it separates the class distributions. The second was a gradient based band importance method, where the contribution of each band is estimated from the sensitivity of the trained model with respect to the input spectrum.

Although both techniques are common in hyperspectral work, they did not help in our setting. Reducing the input to a selected set of bands or reweighting bands through importance scores consistently reduced the accuracy of the spatial and spectral models. For this reason, the final pipeline keeps all cleaned spectral bands without additional feature selection.

We also did not apply explicit regularization methods such as dropout or weight decay. The core models, especially the two–dimensional CNN, showed strong performance on both the training and testing splits without signs of overfitting, so additional regularization was not required.

3.4. Model architectures

The study compares several model families that capture different aspects of hyperspectral data. These models are grouped into spectral models, spatial models and joint spectral–spatial models. Using multiple architectures allows a fair assessment of how different forms of context contribute to classification accuracy.

Spectral models.. Spectral models operate on the pixel spectrum viewed as a one–dimensional signal. Four architectures were used. A multilayer perceptron (MLP) provides a simple nonlinear baseline. A one–dimensional CNN captures short local patterns along the spectral axis. Recurrent models, including an RNN and an LSTM, treat the spectrum as a sequence and can model longer dependencies across wavelengths. These models focus purely on spectral information without using spatial context.

Spatial models.. Spatial models use patches around each labeled pixel to learn texture and shape cues. The main architecture is a two–dimensional CNN trained from scratch on Indian Pines. Deeper backbones such as ResNet and MobileNet were also tested to evaluate whether standard image classification networks can adapt to hyperspectral patches. These models rely mainly on spatial structure and treat the spectral dimension as additional input channels.

7

Spectral–spatial model.. A three–dimensional CNN is used to combine spectral and spatial information in a single architecture. This model applies convolutions jointly across both spatial dimensions and the spectral axis, allowing it to learn interactions that neither spectral nor spatial models capture on their own. The three–dimensional CNN serves as a bridge between the two families and provides a reference for full spectral–spatial modeling.

Overall, evaluating all three model groups helps identify which form of context is most effective for hyperspectral classification and establishes a solid baseline before applying transfer learning across scenes.

3.4.1. Two dimensional CNN flow

```
===========================================================================
Layer (type:depth-idx)               Output Shape          Param #
===========================================================================
CNN2D_NoDrop                         [1, 16]               --
├─Sequential: 1-1                    [1, 128, 1, 1]        --
│    └─Conv2d: 2-1                   [1, 32, 15, 15]       57,632
│    └─ReLU: 2-2                     [1, 32, 15, 15]       --
│    └─BatchNorm2d: 2-3              [1, 32, 15, 15]       64
│    └─Conv2d: 2-4                   [1, 32, 15, 15]       9,248
│    └─ReLU: 2-5                     [1, 32, 15, 15]       --
│    └─BatchNorm2d: 2-6              [1, 32, 15, 15]       64
│    └─MaxPool2d: 2-7                [1, 32, 7, 7]         --
│    └─Conv2d: 2-8                   [1, 64, 7, 7]         18,496
│    └─ReLU: 2-9                     [1, 64, 7, 7]         --
│    └─BatchNorm2d: 2-10             [1, 64, 7, 7]         128
│    └─Conv2d: 2-11                  [1, 64, 7, 7]         36,928
│    └─ReLU: 2-12                    [1, 64, 7, 7]         --
│    └─BatchNorm2d: 2-13             [1, 64, 7, 7]         128
│    └─MaxPool2d: 2-14               [1, 64, 3, 3]         --
│    └─Conv2d: 2-15                  [1, 128, 3, 3]        73,856
│    └─ReLU: 2-16                    [1, 128, 3, 3]        --
│    └─BatchNorm2d: 2-17             [1, 128, 3, 3]        256
│    └─Conv2d: 2-18                  [1, 128, 3, 3]        147,584
│    └─ReLU: 2-19                    [1, 128, 3, 3]        --
│    └─BatchNorm2d: 2-20             [1, 128, 3, 3]        256
│    └─AdaptiveAvgPool2d: 2-21       [1, 128, 1, 1]        --
├─Sequential: 1-2                    [1, 16]               --
│    └─Flatten: 2-22                 [1, 128]              --
│    └─Linear: 2-23                  [1, 256]              33,024
│    └─ReLU: 2-24                    [1, 256]              --
│    └─Linear: 2-25                  [1, 16]               4,112
===========================================================================
Total params: 381,776
Trainable params: 381,776
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 19.79
===========================================================================
Input size (MB): 0.18
Forward/backward pass size (MB): 0.37
Params size (MB): 1.53
Estimated Total Size (MB): 2.08
===========================================================================
```

Figure 1: Architecture summary of the 2DCNN model produced by the `torchinfo` library. The table lists every layer in order together with the output tensor shape and the number of trainable parameters.

Figure 1 shows the detailed architecture of the main two dimensional convolutional network. The summary contains several consecutive entries with type `Conv2d`. This happens because each feature block in the model is built from a pair of convolution layers that share the same number of output channels. In the first block the two convolutions both

use 32 filters, in the second block they both use 64 filters and in the third block they both use 128 filters. The report lists these layers separately, so the repeated `Conv2d` rows simply correspond to the two convolution operations inside each block.

This design follows a common pattern in compact convolution models. Using two 3×3 convolutions in sequence gives an effective receptive field similar to a larger kernel while keeping the parameter count moderate and inserting an extra nonlinear activation between the two convolutions. After the pair of convolutions and the associated batch normalization and rectified linear unit activations, each block applies a pooling layer that reduces spatial resolution in a controlled way. The last block uses adaptive average pooling to compress the feature maps to size 1×1 per channel. The classifier head then flattens this 128 dimensional vector and maps it to the final class scores. The summary therefore confirms the intended structure of three convolution blocks with channel depths 32, 64 and 128 followed by a small fully connected classifier.

3.5. Training procedure

All models were trained using a consistent pipeline. The loss function was cross entropy, and all networks were optimized with AdamW using a learning rate of 10^{-3} and weight decay $5 \cdot 10^{-5}$. A cosine annealing schedule with $T_{\max} = 40$ was used to gradually reduce the learning rate. The batch size was fixed to 64 across the training, validation and test loaders.
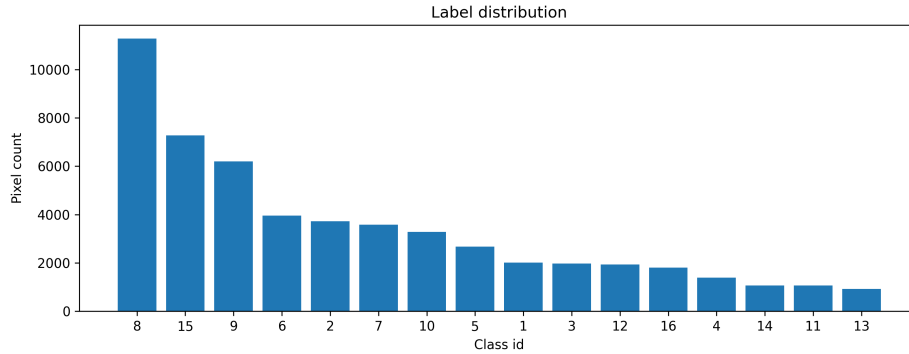

Figure 2: Class distribution for the hyperspectral dataset.

Figure 2 shows that the dataset is highly imbalanced, with a few dominant classes and several rare ones. To address this, we used a weighted random sampler. For each class c with n_c samples, a weight

$$w_c = \frac{1}{\max(n_c, 1)}$$

was assigned, and each training sample inherited the weight of its label. The sampler drew samples in proportion to these weights, ensuring that rare classes contributed equally during training. Validation and test loaders used the same batch size but no sampling.

Early stopping was applied with a patience of ten epochs based on validation loss. No explicit regularization such as dropout was used, since the main spatial models, especially the two dimensional CNN, showed stable behaviour without overfitting.

9

Hyperparameter search was performed only for the spectral models (MLP, one dimensional CNN, RNN and LSTM). Spatial models already achieved near perfect accuracy on Indian Pines, so further tuning was unnecessary. Checkpoints were saved throughout training, and a safe loading routine was used to ensure compatibility when architectures differed between experiments.

3.6. *Transfer learning setup*

Transfer experiments used the two dimensional CNN trained on Indian Pines as the source model. This model served as the spatial backbone for both Salinas and Pavia University. A key difficulty is that the scenes differ in the number of available spectral bands, so the input channel dimension of the pretrained network does not match the target datasets. Three strategies were used to address this mismatch.

Adapter only.. The first approach replaced the first convolution layer and the final linear layer with new layers matching the target band count and the target number of classes. All remaining layers were kept frozen. This created a lightweight adapter that learns to map the target spectra into the feature space learned on Indian Pines. Although simple, this method was unstable and produced low accuracy, showing that the backbone required more adaptation to the new distributions.

Adapter warm up and fine tuning.. In the second approach, the adapter layers were trained first while the backbone remained frozen. After this warm up stage, the last convolution block and the classifier head were unfrozen and trained for a short fine tuning phase. This partial fine tuning allowed the higher level spatial features to adapt to the new scene while preserving most of the pretrained structure. This method produced consistent and reliable transfer performance on both Salinas and Pavia University.

All transfer experiments used the same loss function, optimizer and sampling strategy as the in scene training. The results showed that simple adapters were not sufficient for cross scene generalization, while limited fine tuning of deeper layers was necessary to achieve stable accuracy.

4. Results

4.1. *In scene performance on Indian Pines*

Table 4 shows the performance of all spectral, spatial and spectral–spatial models trained directly on Indian Pines. The results highlight clear differences between the model families. Spectral models, which operate only on pixelwise spectra, achieved moderate accuracy in the range of 0.92–0.94. The 1D CNN and LSTM performed similarly, while the RNN showed the weakest performance, confirming that sequence models alone cannot reliably separate closely related crop classes. The MLP also showed limited accuracy, reflecting the difficulty of learning spectral boundaries without spatial context.

Spatial models performed significantly better. The two dimensional CNN reached 0.9976 accuracy with a macro F1 of 0.9972, showing near perfect classification across the sixteen classes. Deep backbones such as MobileNet and ResNet also performed strongly, with MobileNet reaching the highest accuracy of 0.9980. These results indicate that

spatial context is essential for hyperspectral classification on this dataset, where many classes are spectrally similar but spatially distinct.

220   The three dimensional CNN also achieved excellent performance (0.9961 accuracy), demonstrating the benefit of joint spectral and spatial feature extraction. However, the two dimensional CNN achieved comparable or slightly higher accuracy at a much lower computational cost. Based on these observations, the two dimensional CNN was selected as the source backbone for all transfer learning experiments.

Table 4: Performance of all models on the Indian Pines test set.

Model	Accuracy	Precision	Recall	F1-Score	Kappa
1D CNN	0.9351	0.9514	0.9484	0.9497	0.9260
2D CNN	**0.9976**	0.9957	**0.9988**	**0.9972**	**0.9972**
3D CNN	0.9961	**0.9980**	0.9917	0.9946	0.9955
LSTM	0.9351	0.9514	0.9484	0.9497	0.9260
MLP	0.9220	0.9169	0.9396	0.9260	0.9112
MobileNet	**0.9980**	0.9958	**0.9990**	**0.9974**	**0.9978**
ResNet	0.9898	0.9910	0.9966	0.9937	0.9883
RNN	0.7278	0.7488	0.8312	0.7753	0.6951

Figures 3 and 4 show the qualitative predictions of the two dimensional CNN on the validation and test areas of
225   Indian Pines. The predicted maps align closely with the ground truth, reflecting the strong numerical performance reported earlier. On the test set, the model achieved an accuracy of 0.9927, macro F1 of 0.9919 and kappa of 0.9917. These high scores are visible in the maps, where most classes form clean and spatially coherent regions with very few mislabeled pixels.

The confusion matrix in Figure 5 further confirms this behaviour. The matrix is almost perfectly diagonal, with
230   large correct counts for major classes such as *Soybean-mintill*, *Woods* and *Corn-notill*. Misclassifications appear only in a few rare classes, which is consistent with their limited representation in the training set. This supports the conclusion that the model learns effective spatial-spectral features and maintains high discriminative ability across all sixteen classes.

Figure 6 shows the training loss and validation accuracy curves. The loss decreases steadily over epochs while
235   the validation accuracy quickly rises above 0.98 and remains stable. This pattern indicates smooth convergence and the absence of overfitting, which is consistent with the strong test metrics and the high visual quality of the predicted classification maps.
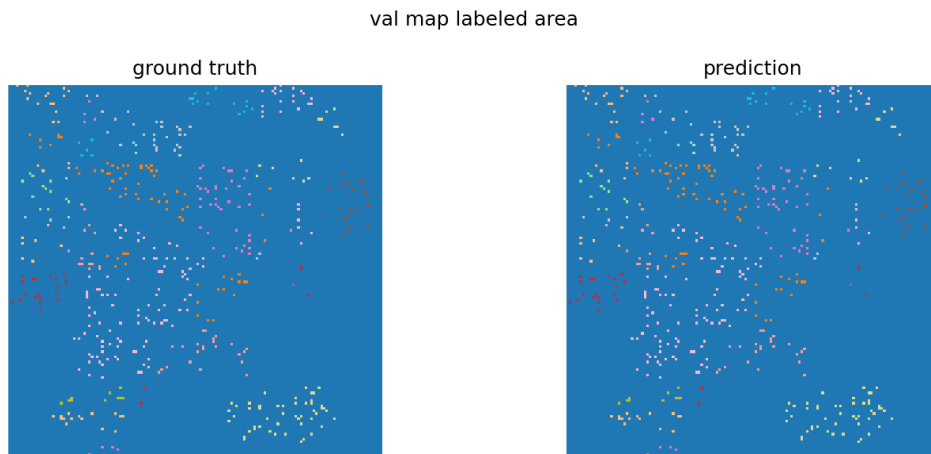
11

Figure 3: Validation set labeled area for Indian Pines. Left: ground truth. Right: predicted map from the two dimensional CNN. The model captures most class regions with high spatial consistency and minimal fragmentation.
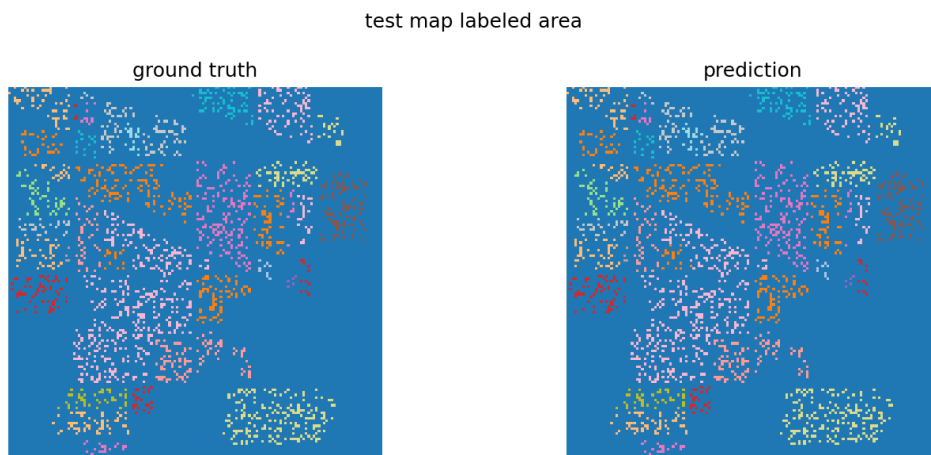
test map labeled area

Figure 4: Test set labeled area for Indian Pines. Left: ground truth. Right: predicted map. The predicted regions align closely with the true spatial structure, reflecting the strong generalization of the trained model.
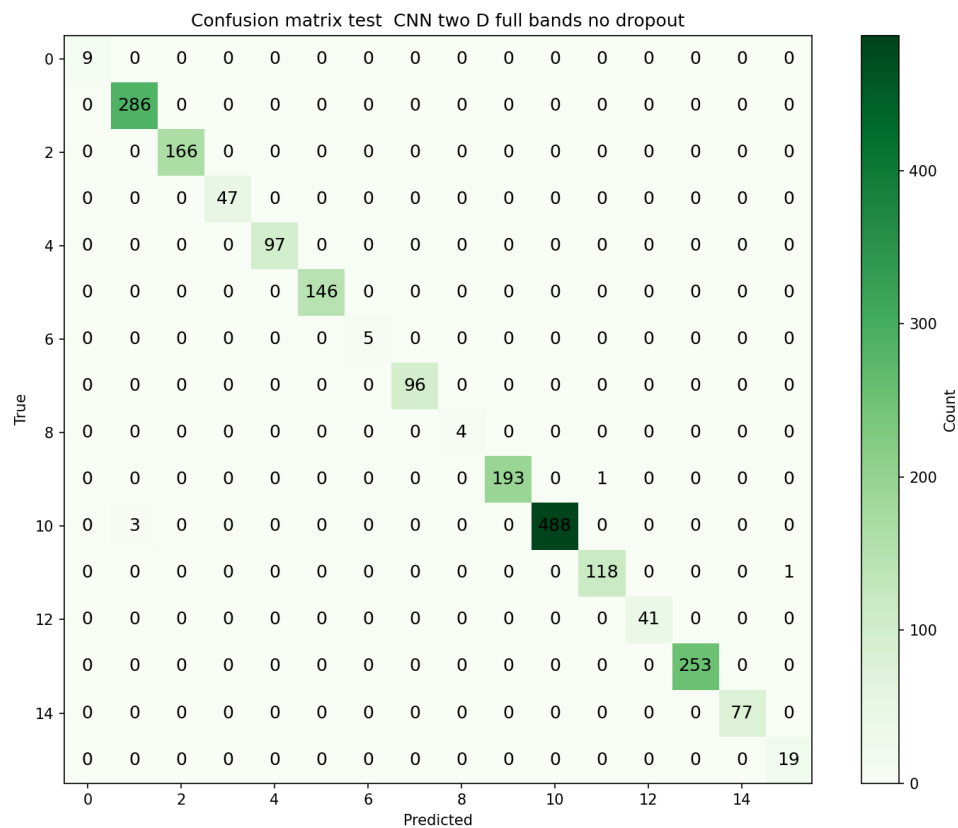
Figure 5: Confusion matrix for the Indian Pines test set using the two dimensional CNN. The near perfect diagonal indicates that almost all classes are correctly identified, with only isolated misclassifications in rare classes.
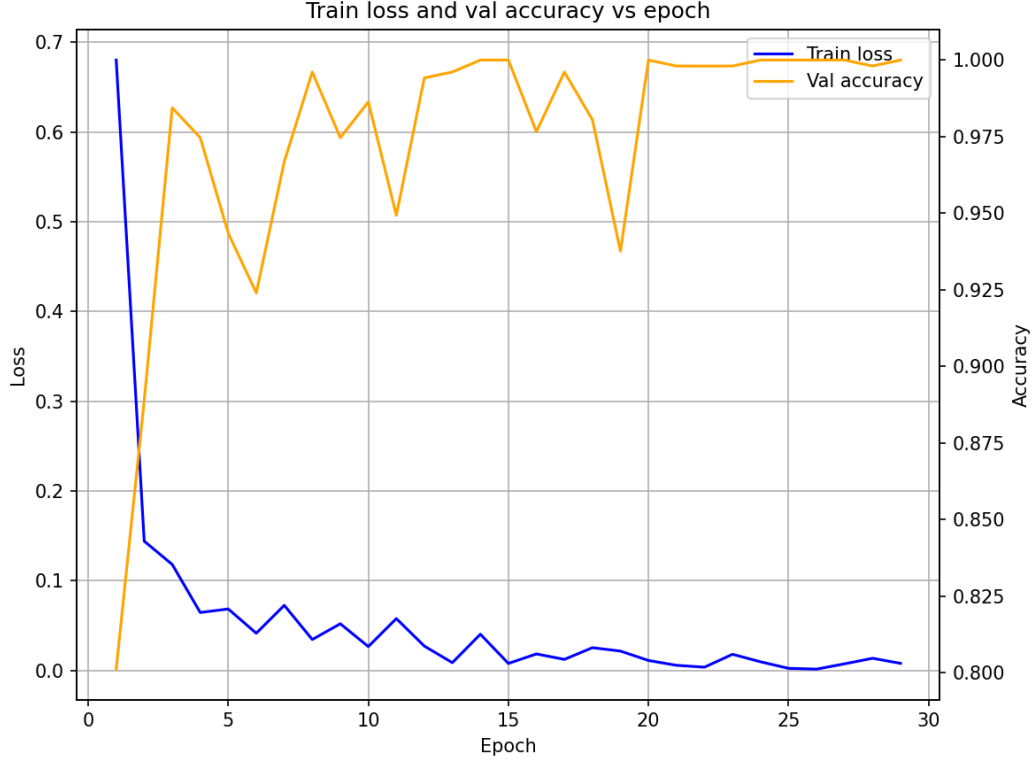
Figure 6: Training loss and validation accuracy across epochs for the two dimensional CNN. The loss steadily decreases while the validation accuracy reaches its peak early and remains stable, confirming the absence of overfitting.

4.2. Salinas Transfer Learning Results

The Salinas scene presents high intra–class variability and strong spectral overlap across crop types. The adapter warmup stage showed unstable behaviour, but the fine tuning stage significantly improved performance. The final accuracy reached 0.9578 with a macro F1 of 0.9147. Several classes such as *Broccoli_green_weeds 1*, *Broccoli_green_weeds 2*, *Stubble* and *Celery* achieved F1 scores close to 1.0, while classes with extreme variability (e.g., *Fallow*, *Lettuce 5wk*) showed lower recall.

Table 5: Performance metrics for Salinas transfer learning.

Metric	Value
Accuracy	0.9578
Precision (macro)	0.9147
Recall (macro)	0.9147
F1-score (macro)	0.9147

14

Table 6: Confusion matrix for Salinas transfer learning. Rows = True class, Columns = Predicted class.

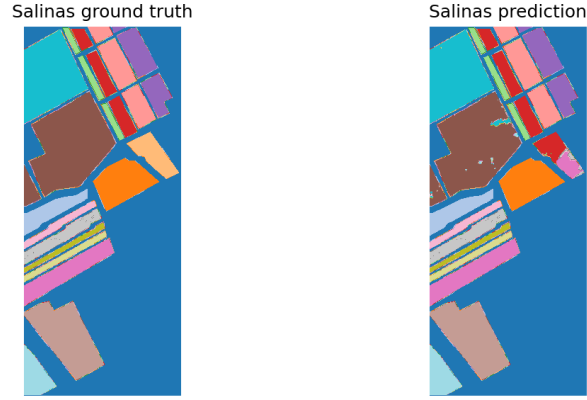|	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	2009	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	3726	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	2	6	1073	0	0	0	0	700	0	185	0	0	10	0
4	0	0	0	1394	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	3	2675	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	3959	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	3579	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	10971	0	0	0	0	0	0	222	78
9	0	0	0	0	0	0	0	0	6203	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	3278	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	5	1063	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	1926	1	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	916	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	1070	0	0
15	0	0	0	0	0	0	0	1	0	0	0	0	0	0	7267	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1807

Figure 7: Salinas ground truth (left) and predicted map (right).

4.3. Pavia Centre Transfer Learning Results

Pavia Centre contains strong man-made structures and sharp boundaries. After fine tuning, the model achieved 0.9809 accuracy and a macro F1 of 0.9086. Most classes such as *Water*, *Tiles*, and *Meadows* were mapped correctly. Misclassifications were concentrated in visually similar materials such as *Self-blocking bricks* and *Shadows*.

Table 7: Performance metrics for Pavia Centre transfer learning.

Metric	Value
Accuracy	0.9809
Precision (macro)	0.9086
Recall (macro)	0.9086
F1-score (macro)	0.9086

Table 8: Confusion matrix for Pavia Centre transfer learning.

	1	2	3	4	5	6	7	8	9
1	65971	0	0	0	0	0	0	0	0
2	0	7583	11	0	0	0	0	0	4
3	0	300	2788	0	1	0	0	0	1
4	0	0	6	773	11	350	1545	0	0
5	2	0	3	2	6152	4	410	9	2
6	0	0	0	0	0	9239	0	1	8
7	0	0	0	0	0	129	7137	17	4
8	0	0	0	0	0	2	0	42815	9
9	0	0	0	0	0	0	0	0	2863

Figure 8: Pavia Centre ground truth (left) and predicted map (right).

4.4. Pavia University Transfer Learning Results

The Pavia University scene contains large homogeneous regions and strong material contrasts. Fine tuning achieved perfect performance, reaching accuracy and macro F1 both equal to 1.0000.

Table 9: Performance metrics for Pavia University transfer learning.

Metric	Value
Accuracy	1.0000
Precision (macro)	1.0000
Recall (macro)	1.0000
F1-score (macro)	1.0000

Table 10: Confusion matrix for Pavia University transfer learning.

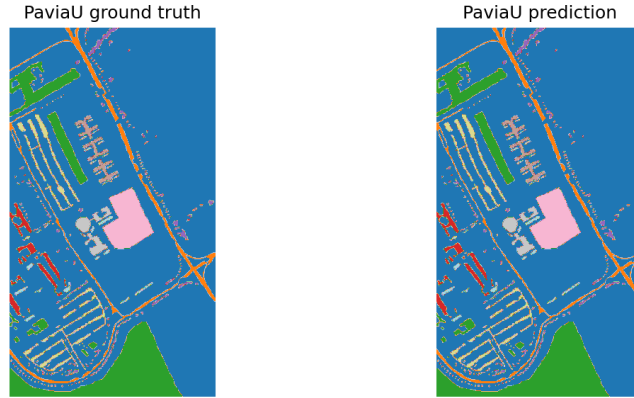|	1	2	3	4	5	6	7	8	9
1	6631	0	0	0	0	0	0	0	0
2	0	18649	0	0	0	0	0	0	0
3	0	0	2099	0	0	0	0	0	0
4	0	0	0	3064	0	0	0	0	0
5	0	0	0	0	1345	0	0	0	0
6	0	0	0	0	0	5029	0	0	0
7	0	0	0	0	0	0	1330	0	0
8	0	0	0	0	0	0	0	3682	0
9	0	0	0	0	0	0	0	0	947

Figure 9: Pavia University ground truth (left) and predicted map (right).

4.5. Comparison of transfer strategies

Across all experiments, two clear patterns emerged:

- **Adapter only** was insufficient, failing on both agricultural and urban scenes due to large spectral shifts.

- **Adapter warm up + fine tuning** consistently produced stable and high accuracy, confirming that deeper layers must adjust to the new domain.

The success of selective fine tuning suggests that the mid level spatial filters are reusable across scenes, while the final layers require adaptation to local textures and spectral ranges.

Overall, spatial CNNs consistently outperformed spectral models, both in direct training and in cross scene transfer. The two dimensional CNN provided a strong, lightweight backbone with near perfect accuracy on Indian Pines. Transfer learning is feasible across scenes and even across sensors, but only when limited fine tuning is allowed. Simple adapters were not sufficient, while short fine tuning reliably adapted the model to both Salinas and Pavia University.

5. Discussion

The experiments on Indian Pines, Salinas, Pavia Centre, and Pavia University confirm that compact spatial CNNs remain highly competitive for hyperspectral image classification, even when compared to more complex 3D or hybrid 2D–3D architectures. The proposed 2D CNN achieves near-perfect accuracy on Indian Pines and transfers effectively across agricultural and urban target scenes when combined with a simple 1×1 spectral adapter and brief fine-tuning of deeper layers. This behaviour aligns with recent work showing that deep spectral–spatial models can reach overall accuracies well above 99% on standard benchmarks, but it also suggests that much of the necessary inductive bias can be captured by spatial feature extractors trained on a single source scene and then adapted with lightweight spectral alignment and limited parameter updates.

17

At the same time, results on Salinas and Pavia Centre highlight that an *adapter-only* strategy is not sufficient under strong spectral shift or sensor change. Freezing the backbone and updating only the input and output layers led to unstable warm-up behaviour and reduced performance, while unfreezing the last convolutional block and classifier head consistently improved accuracy and macro F1 scores. This pattern supports the view that mid-level spatial filters are broadly reusable across scenes, but higher-level representations must adapt to local texture and spectral statistics to avoid underfitting or misalignment. The literature on transfer learning and multitask learning for hyperspectral images reports similar conclusions: cross-scene generalization benefits from sharing intermediate representations while allowing scene-specific adaptation at shallower or deeper layers.

Table 11 situates the present approach within recent deep learning literature on hyperspectral image classification, focusing on methods that report numerical performance on Indian Pines, Salinas, Pavia University, or related benchmarks. Several of these works employ 3D convolutions, hybrid 3D–2D designs, or multitask training across multiple scenes, and many report accuracies above 99% on at least one dataset. Compared to such models, the 2D CNN with spectral adapter used in this study occupies a favourable accuracy–complexity trade-off: it reaches similar performance on Indian Pines and strong transfer on Salinas and Pavia scenes while using a simpler backbone, no dimensionality reduction, and a transparent two-stage adaptation scheme (adapter warm-up followed by short fine-tuning of deeper layers). This underscores the practical value of lightweight, reproducible pipelines that expose all preprocessing and training steps while still achieving state-of-the-art or near state-of-the-art results.

Table 11: Representative deep learning methods for hyperspectral image classification and their reported overall accuracies on common benchmark datasets.

Author	Datasets used	Model type	Transfer / domain adaptation strategy	Performance (overall accuracy per dataset)
Nyabuga et al. [59]	Indian Pines, Salinas, Pavia University, Botswana, Kennedy Space Center	3D–2D CNN with spectral–spatial fusion	Pretrain–finetune on multiple hyperspectral scenes	Indian Pines: 99.85%; Pavia University: 99.98%; Salinas: 100%; Botswana: 99.82%; KSC: 99.71%
Firat et al. [60]	Indian Pines, Pavia University, Salinas, Houston	Spatial–spectral CNN	In-scene training with spatial–spectral feature extraction	Indian Pines: 99.8%; Pavia University: 99.99%; Salinas: 100%; Houston: 99.12%
Kanthi et al. [61]	Indian Pines, Salinas, Pavia University	Fused 3D–2D CNN (F-CNN)	In-scene spectral–spatial fusion without explicit transfer	Indian Pines: 99.92%; Salinas: 99.99%; Pavia University: 99.95%
Zhang et al. [62]	Three real HSIs (e.g., Indian Pines, Pavia University, Salinas/others)	Lightweight 3D-CNN (3D-LWNet)	Transfer learning from 2D RGB data to hyperspectral images	All three datasets: overall accuracy greater than 99% vs. compared CNN baselines
Liu et al. [63]	Indian Pines, Pavia Center, Pavia University, Salinas Valley	Multitask deep CNN with spectral knowledge	Multitask learning across multiple hyperspectral scenes with a shared feature extractor	Pavia Center: 75.25–81.45%; Indian Pines: 39.17–42.44%; Salinas Valley: ≈80%; Pavia University: higher than corresponding single-task baselines
Vaddi et al. [64]	Indian Pines, Pavia University, Salinas	CNN with unsupervised feature extraction	In-scene training with unsupervised pre-training	Overall accuracy higher than prior CNN baselines on all three datasets (OA > 99% reported for at least one dataset)

6. Conclusion

This study demonstrated that spatial convolutional models provide strong performance for hyperspectral image classification and can serve as effective foundations for cross-scene transfer learning. The two dimensional CNN trained on Indian Pines achieved near perfect accuracy and generalized reliably across agricultural and urban environments when combined with a simple adapter and targeted fine tuning. Experiments on Salinas, Pavia Centre and Pavia University showed that adapter-only transfer is insufficient, but a brief fine tuning stage allows the backbone to adjust to new spectral statistics and produce accurate classification maps, even under large domain and sensor shifts. The results highlight that spatial features learned from one scene can be reused across diverse datasets, and that limited adaptation is enough to recover high accuracy and stable predictions. This supports the feasibility of lightweight transfer pipelines for hyperspectral applications where labeled data are limited.

References

[1] X. Kang, C. Li, S. Li, H. Lin, Classification of hyperspectral images by gabor filtering based deep network, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11 (4) (2017) 1166–1178.

[2] L. Liu, J. Feng, B. Rivard, X. Xu, J. Zhou, L. Han, J. Yang, G. Ren, Mapping alteration using imagery from the tiangong-1 hyperspectral spaceborne system: Example for the jintanzi gold province, china, International journal of applied earth observation and geoinformation 64 (2018) 275–286.

[3] M. Imani, Manifold structure preservative for hyperspectral target detection, Advances in Space Research 61 (10) (2018) 2510–2520.

[4] X. Cao, F. Zhou, L. Xu, D. Meng, Z. Xu, J. Paisley, Hyperspectral image classification with markov random fields and a convolutional neural network, IEEE Transactions on image processing 27 (5) (2018) 2354–2367.

[5] S. Sharma, K. M. Buddhiraju, Spatial–spectral ant colony optimization for hyperspectral image classification, International Journal of Remote Sensing 39 (9) (2018) 2702–2717.

[6] W. Lv, X. Wang, Overview of hyperspectral image classification, Journal of Sensors 2020 (1) (2020) 4817234.

[7] Q. Wang, Z. Meng, X. Li, Locality adaptive discriminant analysis for spectral–spatial classification of hyperspectral images, IEEE Geoscience and Remote Sensing Letters 14 (11) (2017) 2077–2081.

[8] L. He, J. Li, A. Plaza, Y. Li, Discriminative low-rank gabor filtering for spectral–spatial hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 55 (3) (2016) 1381–1395.

[9] H. Binol, Ensemble learning based multiple kernel principal component analysis for dimensionality reduction and classification of hyperspectral imagery, Mathematical Problems in Engineering 2018 (1) (2018) 9632569.

[10] P. H. Li, T. Lee, H. Y. Youn, Dimensionality reduction with sparse locality for principal component analysis, Mathematical Problems in Engineering 2020 (1) (2020) 9723279.

[11] R. Mohanty, S. Happy, A. Routray, Spatial–spectral regularized local scaling cut for dimensionality reduction in hyperspectral image classification, IEEE Geoscience and Remote Sensing Letters 16 (6) (2018) 932–936.

[12] P. Shamsolmoali, M. Zareapoor, J. Yang, Convolutional neural network in network (cnnin): hyperspectral image classification and dimensionality reduction, IET Image Processing 13 (2) (2019) 246–253.

[13] B. Cui, X. Xie, S. Hao, J. Cui, Y. Lu, Semi-supervised classification of hyperspectral images based on extended label propagation and rolling guidance filtering, Remote Sensing 10 (4) (2018) 515.

[14] B. Fang, Y. Li, H. Zhang, J. C.-W. Chan, Semi-supervised deep learning classification for hyperspectral image based on dual-strategy sample selection, Remote Sensing 10 (4) (2018) 574.

[15] H. Su, Y. Cai, Q. Du, Firefly-algorithm-inspired framework with band selection and extreme learning machine for hyperspectral image classification, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 10 (1) (2016) 309–320.

[16] M. P. Uddin, M. A. Mamun, M. A. Hossain, Effective feature extraction through segmentation-based folded-pca for hyperspectral image classification, International Journal of Remote Sensing 40 (18) (2019) 7190–7220.

[17] M. P. Uddin, M. A. Mamun, M. A. Hossain, Pca-based feature reduction for hyperspectral remote sensing image classification, IETE Technical Review 38 (4) (2021) 377–396.

[18] L. Fang, Z. Liu, W. Song, Deep hashing neural networks for hyperspectral image feature extraction, IEEE Geoscience and Remote Sensing Letters 16 (9) (2019) 1412–1416.

[19] K. Makantasis, K. Karantzalos, A. Doulamis, N. Doulamis, Deep supervised learning for hyperspectral data classification through convolutional neural networks, in: 2015 IEEE international geoscience and remote sensing symposium (IGARSS), IEEE, 2015, pp. 4959–4962.

[20] S. A. Medjahed, M. Ouali, Band selection based on optimization approach for hyperspectral image classification, The egyptian journal of remote sensing and space science 21 (3) (2018) 413–418.

[21] H. Sima, P. Liu, L. Liu, A. Mi, J. Wang, Sparse representation classification based on flexible patches sampling of superpixels for hyperspectral images, Mathematical Problems in Engineering 2018 (1) (2018) 8264961.

[22] A. V. Krysko, J. Awrejcewicz, I. V. Papkova, O. Szymanowska, V. Krysko, Principal component analysis in the nonlinear dynamics of beams: purification of the signal from noise induced by the nonlinearity of beam vibrations, Advances in Mathematical Physics 2017 (1) (2017) 3038179.

[23] A. Datta, S. Ghosh, A. Ghosh, Advances in principal component analysis (2018).

[24] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, J. A. Benediktsson, Deep learning for hyperspectral image classification: An overview, IEEE transactions on geoscience and remote sensing 57 (9) (2019) 6690–6709.

21

[25] N. Audebert, B. Le Saux, S. Lefèvre, Deep learning for classification of hyperspectral data: A comparative review, IEEE geoscience and remote sensing magazine 7 (2) (2019) 159–173.

[26] X. Kang, B. Zhuo, P. Duan, Dual-path network-based hyperspectral image classification, IEEE Geoscience and Remote Sensing Letters 16 (3) (2018) 447–451.

[27] Y. Yu, Z. Gong, C. Wang, P. Zhong, An unsupervised convolutional feature fusion network for deep representation of remote sensing images, IEEE Geoscience and Remote Sensing Letters 15 (1) (2017) 23–27.

[28] W. Li, C. Chen, M. Zhang, H. Li, Q. Du, Data augmentation for hyperspectral image classification with deep cnn, IEEE Geoscience and Remote Sensing Letters 16 (4) (2018) 593–597.

[29] W. Song, S. Li, L. Fang, T. Lu, Hyperspectral image classification with deep feature fusion network, IEEE Transactions on Geoscience and Remote Sensing 56 (6) (2018) 3173–3184.

[30] G. Cheng, Z. Li, J. Han, X. Yao, L. Guo, Exploring hierarchical convolutional features for hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 56 (11) (2018) 6712–6722.

[31] Y. Li, H. Zhang, Q. Shen, Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network, Remote Sensing 9 (1) (2017) 67.

[32] L. Mou, P. Ghamisi, X. X. Zhu, Unsupervised spectral–spatial feature learning via deep residual conv–deconv network for hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 56 (1) (2017) 391–406.

[33] H. Zhang, Y. Li, Y. Zhang, Q. Shen, Spectral-spatial classification of hyperspectral imagery using a dual-channel convolutional neural network, Remote sensing letters 8 (5) (2017) 438–447.

[34] C. Zhang, G. Li, S. Du, Multi-scale dense networks for hyperspectral remote sensing image classification, IEEE Transactions on Geoscience and Remote Sensing 57 (11) (2019) 9201–9222.

[35] V. K. Repala, S. R. Dubey, Dual cnn models for unsupervised monocular depth estimation, in: International conference on pattern recognition and machine intelligence, Springer, 2019, pp. 209–217.

[36] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Communications of the ACM 60 (6) (2017) 84–90.

[37] S. K. Roy, S. Manna, S. R. Dubey, B. B. Chaudhuri, Lisht: Non-parametric linearly scaled hyperbolic tangent activation function for neural networks, in: International Conference on Computer Vision and Image Processing, Springer, 2022, pp. 462–476.

[38] C. Nagpal, S. R. Dubey, A performance evaluation of convolutional neural networks for face anti spoofing, in: 2019 international joint conference on neural networks (IJCNN), IEEE, 2019, pp. 1–8.

[39] C. Yang, F. Rottensteiner, C. Heipke, Towards better classification of land cover and land use based on convolutional neural networks, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 42 (2019) 139–146.

[40] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, IEEE transactions on pattern analysis and machine intelligence 39 (6) (2016) 1137–1149.

[41] F. Poorahangaryan, H. Ghassemian, Spectral-spatial hyperspectral image classification based on homogeneous minimum spanning forest, Mathematical Problems in Engineering 2020 (1) (2020) 8884965.

[42] A. B. Hamida, A. Benoit, P. Lambert, C. B. Amar, 3-d deep learning approach for remote sensing image classification, IEEE Transactions on geoscience and remote sensing 56 (8) (2018) 4420–4434.

[43] Z. Zhong, J. Li, Z. Luo, M. Chapman, Spectral–spatial residual network for hyperspectral image classification: A 3-d deep learning framework, IEEE transactions on geoscience and remote sensing 56 (2) (2017) 847–858.

[44] S. Mei, X. Yuan, J. Ji, Y. Zhang, S. Wan, Q. Du, Hyperspectral image spatial super-resolution via 3d full convolutional neural network, Remote Sensing 9 (11) (2017) 1139.

[45] W. Li, F. Feng, H. Li, Q. Du, Discriminant analysis-based dimension reduction for hyperspectral image classification: A survey of the most recent advances and an experimental comparison of different techniques, IEEE Geoscience and Remote Sensing Magazine 6 (1) (2018) 15–34.

[46] F. Luo, B. Du, L. Zhang, L. Zhang, D. Tao, Feature learning using spatial-spectral hypergraph discriminant analysis for hyperspectral image, IEEE transactions on cybernetics 49 (7) (2018) 2406–2419.

[47] H. Yao, Y. Zhang, Y. Wei, Y. Tian, Broad learning system with locality sensitive discriminant analysis for hyperspectral image classification, Mathematical Problems in Engineering 2020 (1) (2020) 8478016.

[48] X. Yang, Y. Ye, X. Li, R. Y. Lau, X. Zhang, X. Huang, Hyperspectral image classification with deep learning models, IEEE transactions on geoscience and remote sensing 56 (9) (2018) 5408–5423.

[49] M. E. Paoletti, J. M. Haut, J. Plaza, A. Plaza, A new deep convolutional neural network for fast hyperspectral image classification, ISPRS journal of photogrammetry and remote sensing 145 (2018) 120–147.

[50] M. E. Paoletti, J. M. Haut, R. Fernandez-Beltran, J. Plaza, A. J. Plaza, F. Pla, Deep pyramidal residual networks for spectral–spatial hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 57 (2) (2018) 740–754.

[51] P. Ghamisi, E. Maggiori, S. Li, R. Souza, Y. Tarablaka, G. Moser, A. De Giorgi, L. Fang, Y. Chen, M. Chi, et al., New frontiers in spectral-spatial hyperspectral image classification: The latest advances based on mathematical morphology, markov random fields, segmentation, sparse representation, and deep learning, IEEE geoscience and remote sensing magazine 6 (3) (2018) 10–43.

[52] F. Melgani, L. Bruzzone, Classification of hyperspectral remote sensing images with support vector machines, IEEE Transactions on geoscience and remote sensing 42 (8) (2004) 1778–1790.

[53] M. He, B. Li, H. Chen, Multi-scale 3d deep convolutional neural network for hyperspectral image classification, in: 2017 IEEE International Conference on Image Processing (ICIP), IEEE, 2017, pp. 3904–3908.

[54] S. K. Roy, G. Krishna, S. R. Dubey, B. B. Chaudhuri, Hybridsn: Exploring 3-d–2-d cnn feature hierarchy for hyperspectral image classification, IEEE geoscience and remote sensing letters 17 (2) (2019) 277–281.

[55] S. Xie, R. Girshick, P. Dollár, Z. Tu, K. He, Aggregated residual transformations for deep neural networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1492–1500.

[56] Computer Vision Group, University of the Basque Country, Hyperspectral remote sensing scenes, `https://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes`, accessed: 2025 01 11.

[57] F. Cao, Z. Yang, J. Ren, M. Jiang, W.-K. Ling, Does normalization methods play a role for hyperspectral image classification?, arXiv preprint arXiv:1710.02939 (2017).

[58] R. A. Fisher, Statistical methods for research workers, in: Breakthroughs in statistics: Methodology and distribution, Springer, 1970, pp. 66–70.

[59] D. O. Nyabuga, J. Song, G. Liu, M. Adjeisah, [retracted] a 3d-2d convolutional neural network and transfer learning for hyperspectral image classification, Computational Intelligence and Neuroscience 2021 (1) (2021) 1759111.

[60] H. Fırat, M. E. Asker, M. İ. Bayindir, D. Hanbay, Spatial-spectral classification of hyperspectral remote sensing images using 3d cnn based lenet-5 architecture, Infrared Physics & Technology 127 (2022) 104470.

[61] M. KANTHI, J. D. S. KUMAR, K. V. RAO, A fused 3d-2d convolution neural network for spatial-spectral feature learning and hyperspectral image classification, Journal of Theoretical and Applied Information Technology 102 (5) (2024).

[62] H. Zhang, Y. Li, Y. Jiang, P. Wang, Q. Shen, C. Shen, Hyperspectral classification based on lightweight 3-d-cnn with transfer learning, IEEE Transactions on Geoscience and Remote Sensing 57 (8) (2019) 5813–5828.

[63] S. Liu, Q. Shi, Multitask deep learning with spectral knowledge for hyperspectral image classification, IEEE Geoscience and Remote Sensing Letters 17 (12) (2020) 2110–2114.

[64] R. Vaddi, P. Manoharan, Cnn based hyperspectral image classification using unsupervised band selection and structure-preserving spatial features, Infrared Physics & Technology 110 (2020) 103457.