Proceedings of 12th IOE Graduate Conference
Peer Reviewed
ISSN: 2350-8914 (Online), 2350-8906 (Print)
Year: 2022 Month: October Volume: 12

A 3D CNN Approach for Hyperspectral Image Classification

Ravi Giri [a], Dibakar Raj Pant [b]

[a, b] *Department of Electronics and Computer Engineering, IOE, Pulchowk Campus*
[a] 076mscsk009.ravi@pcampus.edu.np, [b] drpant@ioe.edu.np

Abstract
Recently, hyperspectral image (HSI) classification using deep learning methods has become popular and has shown improved performance over traditional machine learning methods. Moreover, convolutional neural network (CNN) has been widely used for automatic feature extraction than manual feature engineering. However, training the deep learning model for the HSI classification task is challenging due to the high dimensionality and few training samples. Furthermore, better extraction of spatial-spectral information from HSI data is also needed for good classification. In order to overcome these challenges, an HSI classification method that uses Factor Analysis (FA) for dimensionality reduction followed by 3D CNN for spatial-spectral feature extraction has been proposed. The proposed method also has relatively less trainable parameters. Finally, evaluation was done using two real datasets: Indian Pines and Salinas. Using only 10% Indian Pines and 5% Salinas data as training samples, the overall accuracy, average accuracy, and cohen's kappa were 97.32%, 96.96%, and 96.95% for Indian Pines and 99.34%, 99.27%, and 99.55% for Salinas respectively.

Keywords
hyperspectral, 3D CNN, classification

1. Introduction

Hyperspectral imaging has recently gained increased attention due to its wide range of applications like land classification, precision agriculture, exploration of oil and gas, environment monitoring, and quality control. A hyperspectral imaging sensor captures multiple images of a given scene in narrow and contiguous spectral bands. It results in each pixel showing a reflectance spectra of the material inside the pixel[1]. Due to such high spectral resolution, each material can uniquely be identified based on its spectral response. Spectroscopic analysis is used to classify objects in the given scene based on their spectral response[2]. However, due to the high between class similarity and within-class differences, classification is challenging[3].

The HSI classification task aims to assign a class to each pixel among a predefined set of classes. Given a pixel $x_i \in R^B$ of the HSI data $X = \{x_1, x_2, ..., x_N\} \in R^{N \times B}$ (B represents the spectral dimension and N represents the total pixels) is assigned a label $y_i \in \{1, ..., C\}$ (C represents the number of classes.)

One of the critical characteristics of HSI is that they contain spatial and spectral information. Therefore, properly using this information with better feature extraction can improve classification performance. Similarly, another characteristic of HSI is high dimensionality. However, as the number of dimensions increases, many training samples are required for classification; otherwise, the learned feature space will be sparse, and the model will fail to generalize. In other words, the learning model's performance can be increased on a fixed amount of training data by increasing the number of features (dimension) until we reach a certain point beyond which the performance degrades if the feature is increased while the training data is not. This is also known as Hughes' phenomenon[4]. Hence, proper dimensionality reduction, which can retain the variability present in data, is required.

This work uses 3D CNN[5] for hyperspectral image classification. For this purpose, the Indian Pines[6] and Salinas datasets are used. In the proposed method, first, the dimensionality reduction of a dataset using factor analysis[7] is done. Then, the data are sent to 3D CNN for spatial-spectral feature extraction, and these features are flattened and applied to a fully connected layer with softmax activation to perform classification.

Finally, the classification performance is measured by overall accuracy, average accuracy, and cohen's kappa metric.

2. Related Works

The classification of hyperspectral images has recently been an exciting research topic. Different supervised and unsupervised techniques have been proposed to accomplish the task. Still, two challenges need to be addressed for HSI classification. The first one is the high dimensionality of the narrow spectral bands, and the second is the limited amount of labeled training data. To overcome the problem of high dimensional data PCA[8] and LDA have been successfully used for dimensionality reduction.

Initially, the HSI classification was done using machine learning-based methods like Support Vector Machine, Multi-nomial Logistic Regression, and K-means clustering[9]. However, most of these methods had just considered exploiting the spectral feature for classifying pixels and disregarded the spatial feature. It was followed by deep learning-based techniques like stacked autoencoders, sparse autoencoders, deep belief networks, and CNN. Due to better feature extraction, the deep learning methods outperformed the classical machine learning methods, which required manual feature engineering. However, even the early deep learning techniques were based on spectral information only. Nevertheless, S. Paul et al. [10] showed improved classification performance of stacked autoencoder by considering spatial and spectral information. Finally, 3D CNNs were used in HSI classification to exploit spatial-spectral information, which improved the classification results.

There are three kinds of CNN-based HSI classification: (1) Extracting only spectral information: They aim to classify each pixel separately using 1D CNN. (2) Extracting only spatial information: They consider a pixel's neighborhood and apply 2D CNN to classify the central pixel. (3) Extracting spatial-spectral information: These model may fuse features extracted by 1D and 2D CNN and was used by H. Zhang et al. [11]. They may also fuse features extracted by other techniques like Attribute Profiles and Morphological Profiles. They may also use 3D CNN. M. Ahmad et al. [12] proposed a 3D CNN-based method for fast classification of HSI using PCA for dimensionality reduction. However, it still has a relatively large number of trainable parameters, which might cause overfitting when training using limited labeled samples. Similarly, S. Mei et al. [13] performed unsupervised training of a 3D CNN autoencoder and used the latent representation for classification.

Based on the literature, it can be seen that making better use of spatial-spectral features and proper dimensionality reduction is necessary for obtaining a model capable of classifying high-dimensional HSI data when the training samples are limited. So, a model that uses FA for dimensionality reduction and 3D CNN for spatial-spectral feature extraction has been proposed for HSI classification in this work.

3. Methodology

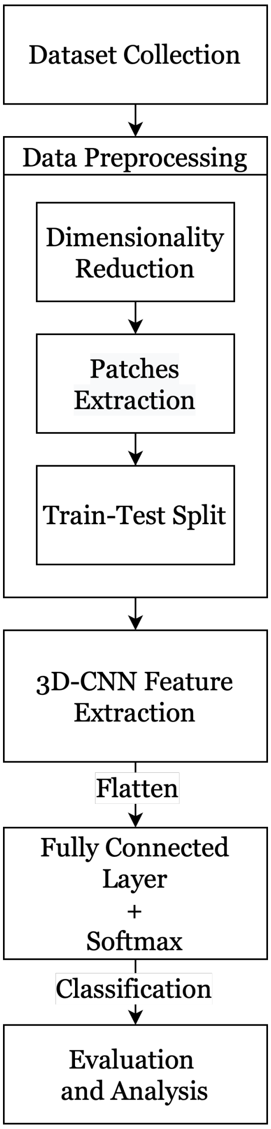


Figure 1: Methodology

The proposed methodology for training and testing the 3D CNN for HSI classification is shown in Figure 1. Data collection, preprocessing, extracting spatial-spectral features using 3D CNN, and finally classifying using a fully connected layer with softmax activation compromise the steps used for good hyperspectral image classification.

3.1 Dataset Description

Indian Pines and Salinas, which are the most popular datasets for hyperspectral classification research, are used. The images were captured by the AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) sensor with 224 spectral channels, which are captured in the spectral range of 0.4-2.5 µm.

3.1.1 Indian Pines

The HSI size is 145 x 145 pixels. 20 m/pixel is the spatial resolution of the dataset. The ground truth labels provided for the dataset consist of 16 classes.

Table 1: Indian Pines Dataset Detail

Class	No. of Samples
1	46
2	1,428
3	830
4	237
5	483
6	730
7	28
8	478
9	20
10	972
11	2,455
12	593
13	205
14	1265
15	386
16	93

3.1.2 Salinas

The HSI size is 512 x 217 pixels. It has a high spatial resolution of 3.17 m/pixel. Table 2 shows the ground truth classes provided for the dataset.

Table 2: Salinas Dataset Detail

Class	No. of Samples
1	2,009
2	3,726
3	1,976
4	1,394
5	2,678
6	3,959
7	3,579
8	11,271
9	6,203
10	3,278
11	1,068
12	1,927
13	916
14	1,070
15	7268
16	1,807

3.2 Data Preprocessing

Various kinds of noise may be present in real-world data. Due to water absorption, noisy channels in the Indian Pines and Salinas dataset were also removed. Then, Z-Score normalization was done on data as:

$$x^{'} = \frac{x-\mu}{\sigma} \tag{1}$$

This ensures that the feature distribution has a mean of zero and a standard deviation of one.

3.2.1 Dimensionality Reduction

HSI is generally represented in 3D cube form with 2D spatial information and 1D spectral information. As HSI has high spectral dimension, training with a limited amount of data leads to overfitting and poor generalization as the feature space representation will be sparse. Hence, dimensionality reduction is necessary. Factor Analysis has been used for dimensionality reduction.

Factor Analysis (FA)

FA reduces variables into lower dimensional factors, which are generally less in numbers than variables. Hence, factors are also termed latent variables. This technique explains the variance in all observed variables using a few unobserved factors. For example, two factors might be sufficient to explain the variation in five observed variables. So, the observed variables are a linear combination of factors and error terms.

Given N samples of flattened HSI data cube with B bands $X \in R^{B\times N}$, factors $F \in R^{K\times N}$ matrix which relates factors to observation $L \in R^{B\times K}$, error term matrix $\varepsilon \in R^{B\times N}$ and mean matrix $M \in R^{B\times N}$, then according to factor analysis,

$$X = LF + M + \varepsilon \tag{2}$$

The maximum-likelihood estimate is done to get L, and the latent variables F are transformed using Singular Value Decomposition (SVD).

Hence, FA reduces the number of spectral dimensions in HSI to K while the spatial resolution is unaffected.

Then, the HSI with $Y \times Z$ spatial dimension and K spectral dimension was divided into $N(N = Y \times Z)$ 3D data patches of shape $n \times n \times K$ using the sliding window of size $n \times n$ along the spatial dimension. This was done as training on the large HSI data cube is computationally expensive. Each pixel is considered for patch creation by placing it in the center of the sliding window, and these patches are labeled based on the label of that central pixel.

10% of patches were used for training, and the remaining patches were used for testing in Indian Pines. Similarly, for Salinas, 5% of the patches were used for training, and the remaining patches were used for testing. The train-test split was done using stratified sampling.

3.3 3D CNN

While 1D CNN can extract spectral information, it fails to consider spatial, and 2D CNN fails to capture better spectral information. Nevertheless, in 3D CNN, as a filter is applied along the spectral dimension in addition to the spatial dimension of the 3D input patches, it can better extract spatial-spectral features. Moreover, recent works have shown that considering spatial and spectral information improves HSI classification performance.

The proposed method consists of four layers of 3D CNN for extracting spatial-spectral features. The filter size and the number of filters in each layer are given in Table 3. The first three 3D CNN layers are employed to extract spatial-spectral information, and the final layer is used to refine spectral information. A dropout layer is added between each convolutional layer after non-linear activation, so the model does not overfit when there are few training data. The activation in the l^{th} feature map of k^{th} layer, for position (x,y,z) is given by,

$$v_{k,l}^{x,y,z} = f(\sum_{c}\sum_{h=0}^{H_k-1}\sum_{w=0}^{W_k-1}\sum_{d=0}^{D_k-1} W_{k,l,c}^{h,w,d} . v_{(k-1),c}^{(x+h),(y+w),(z+d)} + b_{k,l}) \tag{3}$$

where H_k, W_k, D_k represents the size of the convolutional kernel, c is the index of the previous layer feature map, and f is the PReLU activation function given by,

$$f(x) = \begin{cases} x, & \text{if } x > 0 \\ ax, & \text{otherwise} \end{cases} \tag{4}$$

where a is a parameter that is learned during training. The features extracted by 3D CNN are flattened and applied to the fully connected layer, having output neurons equal to the number of classes to be classified. The softmax activation in this layer gives the vector of class probabilities as output. Then, the largest probability is considered as an output class.

Table 3: Architecture of 3D CNN for HSI Classification

Layer	Kernel Size
Conv1	$5 \times 5 \times 8$, 8
Conv2	$3 \times 3 \times 8$, 16
Conv3	$3 \times 3 \times 8$, 32
Conv4	$1 \times 1 \times 8$, 64

$$\sigma(Z) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{5}$$

where σ is softmax activation, and Z is the input vector.

The focal loss is used to handle class imbalance, which penalizes hard misclassified examples more than the easy ones.

$$FL(p) = -(1-p)^{\gamma} log(p) \tag{6}$$

where FL is focal loss, γ is modulating factor, and p is the class probability vector.

3.4 Evaluation

First, the confusion matrix $M = (a_{ij})_{n\times n}$ is constructed based on the actual and predicted class for each pixel, where n is the number of predefined classes and a_{ij} is the number of pixels which are predicted as class i but are class j. Then, assuming T is the total number of test samples, the proposed method is evaluated using the metrics discussed below.

3.4.1 Overall Accuracy(OA)

OA is the ratio of samples correctly identified to the total number of samples. It does not consider class imbalance.

$$OA = \frac{\sum_{i=1}^{n} a_{ii}}{T} \tag{7}$$

3.4.2 Average Accuracy(AA)

AA is the average of each class's accuracy. AA can deal with an imbalanced dataset.

$$AA = \frac{1}{n}\sum_{i=1}^{n} \frac{a_{ii}}{\sum_{j=1}^{n} a_{ji}} \tag{8}$$

3.5 Cohen's Kappa (k)

Cohen's Kappa score gives the agreement between two raters when classifying the items into mutually exclusive groups. In addition, it measures how often two raters will agree while considering agreement by chance.

$$k = \frac{\sum_{i=1}^{n} a_{ii} - \frac{\sum_{i=1}^{n}(a_{i\_}a_{\_i})}{T}}{T - \frac{\sum_{i=1}^{n}(a_{i\_}a_{\_i})}{T}} \tag{9}$$

where $a_{i\_}$ and $a_{\_i}$ represent all the elements of row i and all the elements of column i, respectively.

4. Results and Analysis

First, the HSI data cubes were reduced to 40 spectral dimensions using FA. Then, a total of 10 runs were done for each dataset. Training samples were randomly selected for each run by stratified sampling (to ensure that minority classes were also present in training data). Each run consisted of 50 epochs with a batch size of 64 and a learning rate of 0.001 using the Adam optimizer.

Table 4 shows the summary of the model parameters used for the experiments.

4.1 Accuracy and Loss Curves

A run's loss and accuracy curves for the Indian Pines dataset are shown in Figure 2 and Figure 3, and for the Salinas dataset are shown in Figure 4 and Figure 5.

Table 4: Model Parameters

Parameter	Value
Input size	9 x 9 x 40
Dropout	0.5
Epoch	50
Batch Size	64
Learning Rate	0.001
Optimizer	Adam
Momentum	$\beta_1 = 0.9$ $\beta_2 = 0.999$
Loss Modulating Factor	2

4.1.1 Indian Pines Dataset

As the training progresses, it is observed that loss is decreasing and accuracy is increasing rapidly up to 20 epochs. After that, the loss decreases steadily and has saturated within 50 epochs. The gap between the test and training curve is also small, which indicates that the model has good generalization ability.

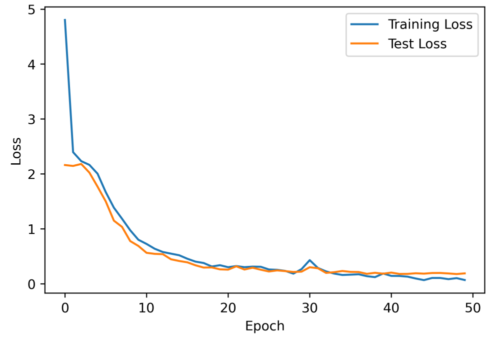


Figure 2: Loss Curve of Indian Pine Dataset

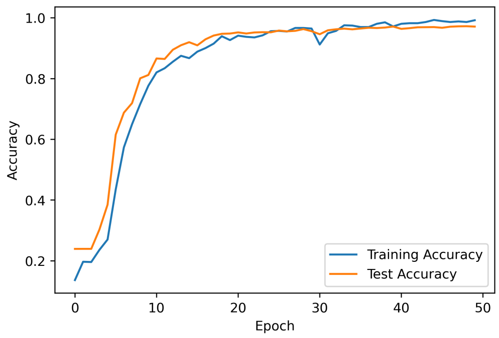


Figure 3: Accuracy Curve of Indian Pine Dataset

4.1.2 Salinas Dataset

Due to the large training data available for the Salinas dataset compared to Indian Pines, even while sampling 5%, the model converged earlier. Other possible reasons are less intra-class variability, less inter-class similarity, and higher spatial resolution than Indian Pines.

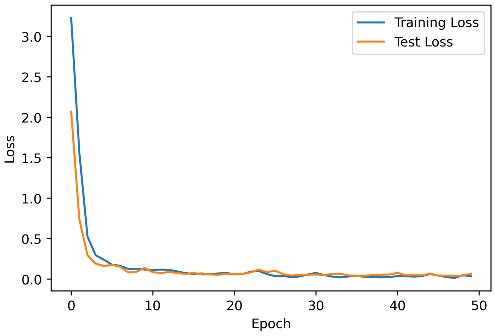


Figure 4: Loss Curve of Salinas Dataset

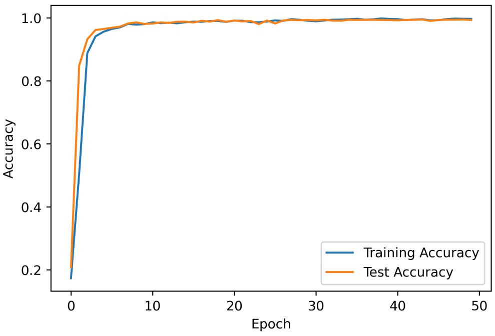
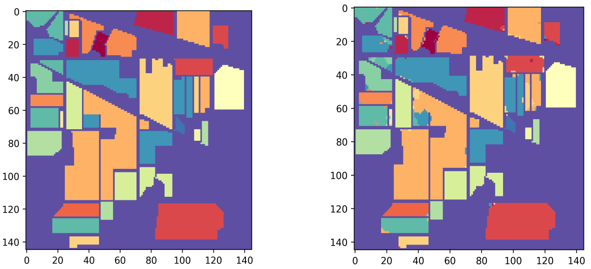


Figure 5: Accuracy Curve of Salinas Dataset

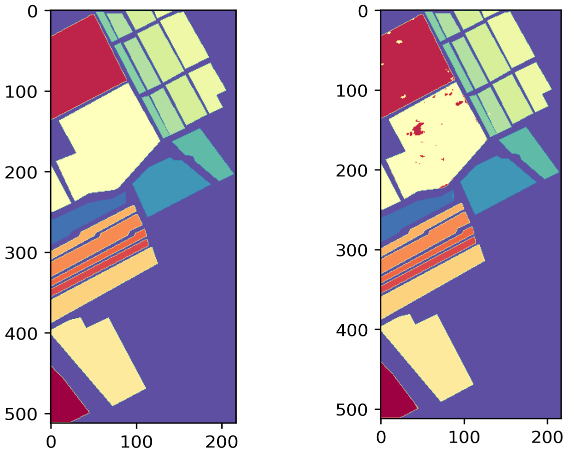
4.2 Qualitative Results

The classification map from the proposed method is compared to the ground truth classification map for the two datasets in Figure 6 and Figure 7. The comparison shows that the classification map from the proposed method has few misclassifications and is almost identical to the ground truth. Most misclassification for Indian Pines seems to happen at the pixel whose neighborhood mainly contains background. This highlights the importance of spatial information. Nevertheless, the method performed good classification despite high inter-class similarity and intra-class variability usually prevalent in the hyperspectral dataset.



(a) Ground Truth **(b)** Classified by 3D CNN

Figure 6: Comparison of (a) Ground Truth and (b) 3D CNN Classification Map for Indian Pines Dataset



(a) Ground Truth **(b)** Classified by 3D CNN

Figure 7: Comparison of (a) Ground Truth and (b) 3D CNN Classification Map for Salinas Dataset

4.3 Quantitative Results

The model's performance has been evaluated using OA, AA, and Cohen's Kappa. The average value of 10 runs for the two datasets are shown in Table 5 and Table 6. From Table 5, it is observed that even for the minority class, the performance is good without doing any class balancing due to the focal loss function used. Increasing the modulating term in focal loss might even further boost the performance of the minority class.

A comparison with existing methods based on the results reported in the companion paper that uses a similar data split for training and testing has also been made in Table 8, and the proposed method has shown good performance.

Table 5: Classification Results on Indian Pines Dataset with 10% Training Samples

Class	Accuracy (%)
1	97.09
2	96.88
3	95.93
4	94.51
5	96.89
6	99.81
7	96.81
8	99.95
9	91.66
10	96.3
11	97.3
12	96.1
13	98.69
14	98.72
15	95.53
16	99.16

Table 6: Classification Results on Salinas Dataset with 5% Training Samples

Class	Accuracy (%)
1	99.84
2	99.99
3	1
4	99.78
5	99.20
6	99.97
7	99.97
8	99.17
9	99.94
10	99.93
11	1
12	99.95
13	99.80
14	98.64
15	97.33
16	99.28

Table 7: Performance Metric on Indian Pines and Salinas Dataset with 10% and 5% Training Samples respectively

	OA (%)	AA (%)	Kappa (%)
Indian Pines	97.32	96.96	96.95
Salinas	99.34	99.27	99.55

Table 8: Comparison with Other Method

	Indian Pines		Salinas	
	OA	*AA*	*OA*	*AA*
AE 3DCNN[13]	92.35	92.04	95.81	97.45
Fast 3DCNN[12]	97.75	94.54	98.06	98.80
Proposed	97.32	96.96	99.34	99.27

4.4 Effect of Training Samples Size

The above results have shown that the proposed method performs well even when small training data are available. Here the effect of the training data size on the model is further analyzed.

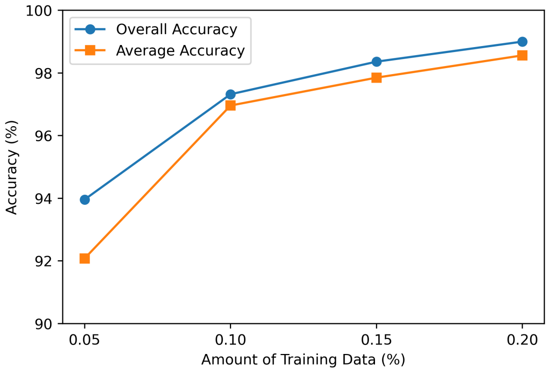


Figure 8: Effect of Training Samples Size on Indian Pines Dataset

Figure 8 clearly shows that the increase in the number of training samples leads to better performance. The performance improvement for the first 0.05% is observed to be more significant. After that, the performance improvement is steady. During the extreme case of using a 0.05% sample, the difference between OA and AA is also high, which is due to the model's inability to correctly classify minority classes, as the training samples for those classes will be scarce.

4.5 Comparison of Dimensionality Reduction Technique

This section compares FA and PCA for dimensionality reduction on the Indian Pines dataset with varying training data sizes.

From Figure 9, it can be observed that when fewer training data are available, dimensionality reduction using FA has better performance. However, the performance gap between FA and PCA is narrowing with increased training samples. Overall, using FA for

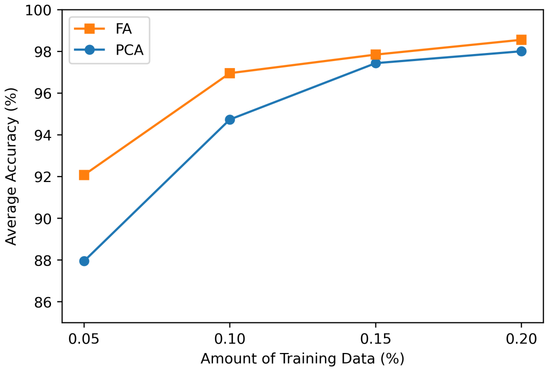


Figure 9: Comparison of Dimensionality Reduction Technique on Indian Pines Dataset

dimensionality reduction in HSI seems to give better classification performance.

5. Conclusion

Hence, a 3D CNN approach that makes use of spatial-spectral information for HSI classification has been proposed in this work. The proposed method was observed to scale well according to the data available for training and is also faster to train as it has relatively less trainable parameters. The method has also shown good classification performance with limited data, which might benefit various applications. The class imbalance problem was tackled using the focal loss function, and the curse of dimensionality was mitigated using FA. A comparison between FA and PCA was also made in which FA was found to perform better than PCA and provided a considerable boost in performance when the training sample was deficient. Evaluating the model trained on 10% Indian Pines and 5% Salinas dataset, OA, AA, and Cohen Kappa were 97.32%, 96.96%, and 96.95% for Indian Pines and 99.34%, 99.27%, and 99.55% for Salinas respectively.

References

[1] Alexander F. H. Goetz, Gregg Vane, Jerry E. Solomon, and Barrett N. Rock. Imaging spectrometry for earth remote sensing. *Science*, 228(4704):1147–1153, 1985.

[2] Chein-I Chang. *Hyperspectral imaging: techniques for spectral detection and classification*, volume 1. Springer Science & Business Media, 2003.

[3] Shaohui Mei, Qianqian Bi, Jingyu Ji, Junhui Hou, and Qian Du. Spectral variation alleviation by low-rank matrix approximation for hyperspectral image analysis. *IEEE Geoscience and Remote Sensing Letters*, 13(6):796–800, 2016.

[4] G. Hughes. On the mean accuracy of statistical pattern recognizers. *IEEE Transactions on Information Theory*, 14(1):55–63, 1968.

[5] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, page 495–502, Madison, WI, USA, 2010. Omnipress.

[6] Marion F. Baumgardner, Larry L. Biehl, and David A. Landgrebe. 220 band aviris hyperspectral image data set: June 12, 1992 indian pine test site 3, Sep 2015.

[7] Richard Arnold Johnson and Dean W. Wichern. *Applied multivariate statistical analysis*. Prentice Hall, Upper Saddle River, NJ, 5. ed edition, 2002.

[8] Xudong Kang, Xuanlin Xiang, Shutao Li, and Jón Atli Benediktsson. Pca-based edge-preserving features for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(12):7140–7151, 2017.

[9] Juan Mario Haut, Mercedes Paoletti, Javier Plaza, and Antonio Plaza. Cloud implementation of the k-means algorithm for hyperspectral image analysis. *J. Supercomput.*, 73(1):514–529, jan 2017.

[10] Subir Paul and D. Nagesh Kumar. Spectral-spatial classification of hyperspectral data with mutual information based segmented stacked autoencoder approach. *ISPRS Journal of Photogrammetry and Remote Sensing*, 138:265–280, 2018.

[11] Haokui Zhang, Ying Li, Yuzhu Zhang, and Qiang Shen. Spectral-spatial classification of hyperspectral imagery using a dual-channel convolutional neural network. *Remote Sensing Letters*, 8(5):438–447, 2017.

[12] Muhammad Ahmad, Adil Mehmood Khan, Manuel Mazzara, Salvatore Distefano, Mohsin Ali, and Muhammad Shahzad Sarfraz. A fast and compact 3-d CNN for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2022.

[13] Shaohui Mei, Jingyu Ji, Yunhao Geng, Zhi Zhang, Xu Li, and Qian Du. Unsupervised spatial–spectral feature learning by 3d convolutional autoencoder for hyperspectral classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(9):6808–6820, 2019.