



A 3D-Inception CNN for Hyperspectral Image Classification

Murali Kanthi^{1*} T. Hitendra Sarma² C. Shoba Bindu¹

¹*Department of Computer Science & Engineering, JNTUCEA, Anantapur. A.P., India*

²*Department of Information Technology, Vasavi College of Engineering, Hyderabad, Telangana, India*

* Corresponding author's Email: murali.kanthi@gmail.com

Abstract: The classification of hyperspectral image (HSI) has attracted significant attention from the research community of remote sensing. HSI analysis suffers from overfitting due to the limited number of labelled training samples. As a result, in order to enhance the performance of the HSI classification task, a better efficient neural network architecture should be developed. To tackle this issue, this letter presents a new 3D-Inception CNN (3D-ICNN) model for dynamically extracting features by stacking inception modules in the network that can learn more representative features with fewer training samples by adopting variable spatial size convolutional filters and dynamic CNN architecture. The experimental results exhibit that the presented model can modify the network design adaptively and achieve higher classification performance. To establish the efficiency and robustness of the presented model, the experiments are conducted on the publicly available benchmark data sets and also on the new data sets. The proposed 3D-Inception CNN model obtained accuracies of 86.25% on Ahmedabad-1(AH1) dataset, 80.30% on Ahmedabad-2(AH2) dataset, 99.95% on the Pavia University (PU) dataset, 99.86% on the Salinas (SA) dataset, and 99.89% on the Indian Pines (IP) dataset.

Keywords: Convolutional neural network, Inception model, Hyperspectral image, Deep learning.

1. Introduction

The last few years, remote sensing researchers have given considerable attention to the interpretation of hyperspectral images (HSIs) acquired from space by satellites or aircraft. HSIs are made up of hundreds of electromagnetic spectrum band series and have a high spectral resolution [1]. Each HSI pixel contains spectral data. HSIs could be a rich resource, which allows objects and materials to be identified. HSIs have shown to be useful in a variety of applications like agriculture, mining, urban planning, defence, environmental monitoring and due to the rich spectral data [2-3]. One of the primary challenges in HSI analysis is the HSI classification that allocates each pixel vector to distinct class. For HSI classification, a variety of supervised classification approaches have been proposed. Deep CNN models have shown efficient in extracting features that increase the accuracy of HSI classification [3-5]. The HSIs classification literature

is split into two primary groups. The first group treats each pixel spectrum for a given class independently. The second kind is spectral-spatial, that merges a single pixel with its neighbours to produce a cluster block. Because we evaluate the spectrum of each pixel independently, the proposed work falls under the first category.

Modern HSI analyses are based on deep learning algorithms that extract more discrimination and produce higher performance than traditional [6]. For hyperspectral data processing, deep learning methods, such as 1-D [7], 2-D [8] and 3-D [9] networks, have been suggested. A 1-D method takes spectral data only as input to learn features. A two-CNN model was created by Yang [8] for learning jointly spectral-spatial features. Zhou [10] applied a CNN to classify HSI pixel vectors in conjunction with the random Markov field in a single Bayesian frame. SSUN [11] integrated long-term spectral band model with 2D-CNN, which included a unified neural network with a spatial finite element, spectral FE, classification training.

Deep neural networks are a robust non-linear function that outperforms existing approaches for processing high dimensional HSI data. Another benefit of deep networks is that their receptive regions analyse neighbouring pixels automatically. The spatial feature offers context and aids in HSIs pixel-by-pixel classification. While deep networks take account of the spatial context, traditional CNN models simply take advantage of the fixed kernel size. The distribution of the hyperspectral image over land still is difficult, traditional CNN with fixed kernel size is inadequate flexibility. Convolution with varied spatial context sizes can capture a more discriminatory HSI pixel classification context feature. Deep CNN has proved its power in various computer vision applications like [12, 13] for high-level extraction. Roy [14] proposed a HybridSN model which uses spectral-spatial 3D-CNN followed by spatial 2DCNN. It provides good classification accuracy with a small training sample. Bandar Alotaibi [15] designed Inception and ResNet hybrid network that combines the key principle behind ResNet, which is the usage of residual blocks, and the input sequence behind the Inception model, which is that its topology is superior than each one separately in terms of accuracy. However, the accuracy for some datasets is very low. Haokui Zhang [16] proposed an AI-Net that can utilize the power of deep learning to extract representative features, as well as a transfer learning of data fusion approach for finer model initialization and reduced time of training. Nevertheless, the model improvement has to employ some strategies to overcome the HSI classification imbalance of the data. Yang [17] developed a SyCNN model that blends 2D and 3D CNNs in feature learning with a hybrid module that integrates spectral-spatial HSI data with 3D attention mechanism. However, for the feature extraction process to be improved, optimum model design is necessary.

Only less than 10% of the samples may be used for HSIs classification training in addition to the aforementioned ones. However, in general, deep learning models have huge training parameters. If the sample size is too little, deep models will probably overfit. Recent proposals have been made for deep network architectures like DensNet [18] and ResNet [19] and they are quite profound, even in over 100 layers. The networks function in massive amounts of data such as Coco and Imagenet. These networks are, however, too complex for HSI images. The training samples are insufficient, thus there are overfitting issues with these models.

We build a 3D-Inception CNN model termed 3D-ICNN to overcome the above-mentioned issues. The contributions of the work can be summed up as follows:

- 1) Proposed a new 3D-inception network which can learn deep spatial data to improve HSIs classification performance by adapting convolution kernel size with different spatial context.
- 2) The proposed approach tackles the problem of overfitting by the use of CNN architecture with inception modules.

The remainder of this article is organized in the following manner. The background on CNN and inception model is presented in Section 2. The suggested model for feature learning and HSI classification is demonstrated in Section 3. Experimental investigation is presented in Section 4, which contains data descriptions, experimental setup, and classification findings. Finally, Section 5 presents the results.

2. Background on CNN and inception model

In this part, we mostly study CNN's background and the inception module.

2.1 CNN

CNN uses a multi-layer trained architecture consisting of a stacked, pooling layers and non-linearity which is used for learning characteristics such as textures and edges and high-level features with more discriminating data [20-22]. Fig. 1 depicts a typical CNN architecture.

Every hidden layer entity is coupled by shared weights to the local responsive field over the input instead of being entirely connected to the input in the converting layer, which may be 2-dimensional size $m \times n$ feature maps in the convolution layer. The convolution layer produces a concentration of the X^i input feature maps with a K^i kernel of the size $k \times k \times m$ and a nonlinear element activation feature, which subsequently applies to the X^i output maps. To predict classification labels, fully connected with Softmax layers are applied after stacked layers have been completed.

2.2 GoogleNet (inception)

Many convolution layers are followed by one or more fully connected (FC) layers in a standard CNN model. In the traditional meaning, the FC layer is a

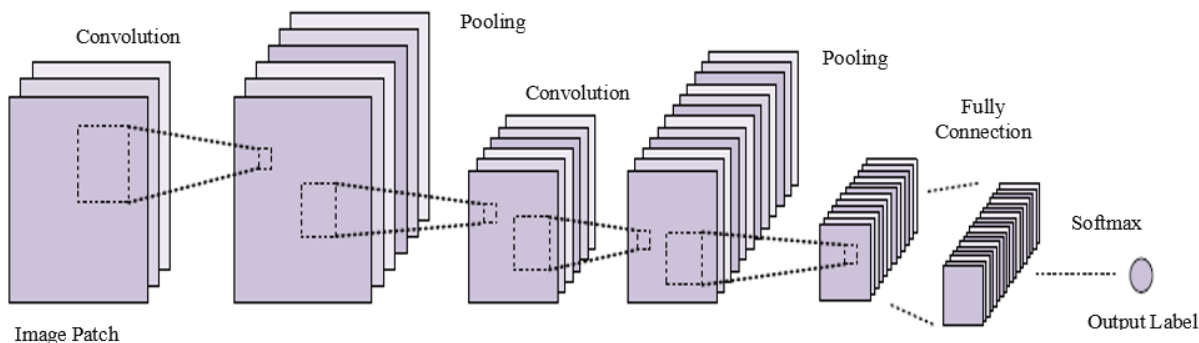


Figure. 1 Typical CNN structure illustration

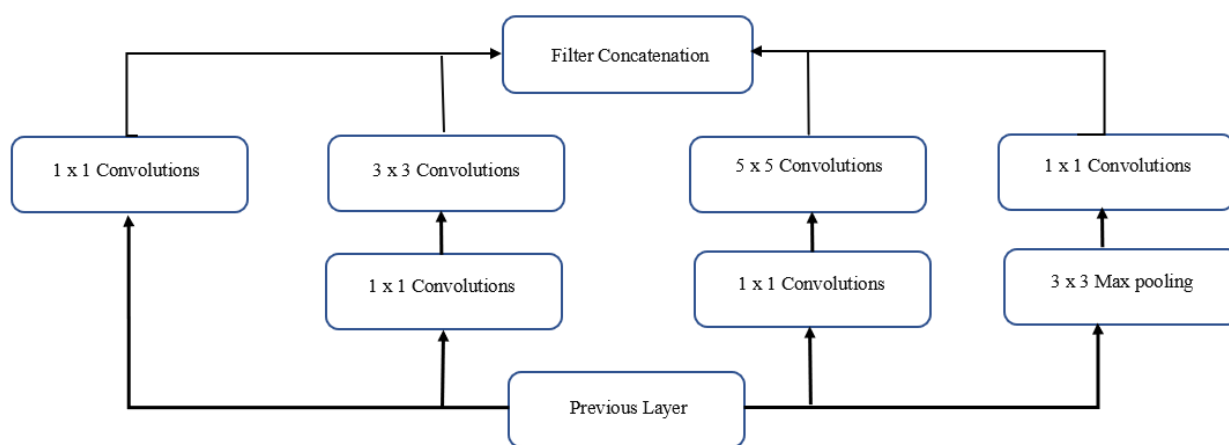


Figure. 2 Core building block of inception model

multilayer neural network. The last FC layer stores the output. The convolutional layer's goal is to use many filters (learnable weights) to convolve the input image, whereas the pooling layer's goal is to down sample the data. The pooling layer typically contains two sorts of functions: max-pooling and average-pooling. From the original raw pixels to the final class scores, the CNN transforms the input image through multiple stacked layers. CNN architectures have served as the foundation for numerous semantic segmentation models.

Goog-LeNet [23] is powerful and advanced deep learning model. As seen in Fig. 2, GoogleNet is deep CNN architecture developed by researchers of Google and presented in the year 2014 to public. With 93.3 percent accuracy, this design was named to the ILSVRC's top-five list. With 22 layers, the GoogleNet is extremely complex, and it includes a unique building element called as the Inception model. This architecture uses a network in a pooling layer, a network layer, and large and small convolution layers, which are calculated simultaneously and not in the conventional sequence manner. After that, a 1 x 1 convolution operation is

done in order to reduce the dimensionality. As a result of the parallelism and dimensionality reduction implemented in this design, the number of parameters and operations has been considerably decreased, resulting in significant memory and computational cost reductions [24].

3. Methodology

Since the disparity cover classes in HSI are complicated, most CNN models typically employ a fixed convolution kernel size, which is inadequately adaptable. Convolution with different spatial context sizes may extract more discriminative context features for HSI classification.

3D-Inception CNN (3D-ICNN) is proposed to capture more discriminatively relevant information, as detailed in this section. As shown in Fig. 3, the proposed model 3D-ICNN takes 3D patches as input to obtain deep spatial features from HSI data cube for pixel classification. The following section presents the proposed model architecture details.

Consider a HSI X , which is expressed as three-dimensional cube with dimensions $W \times H \times B$, where

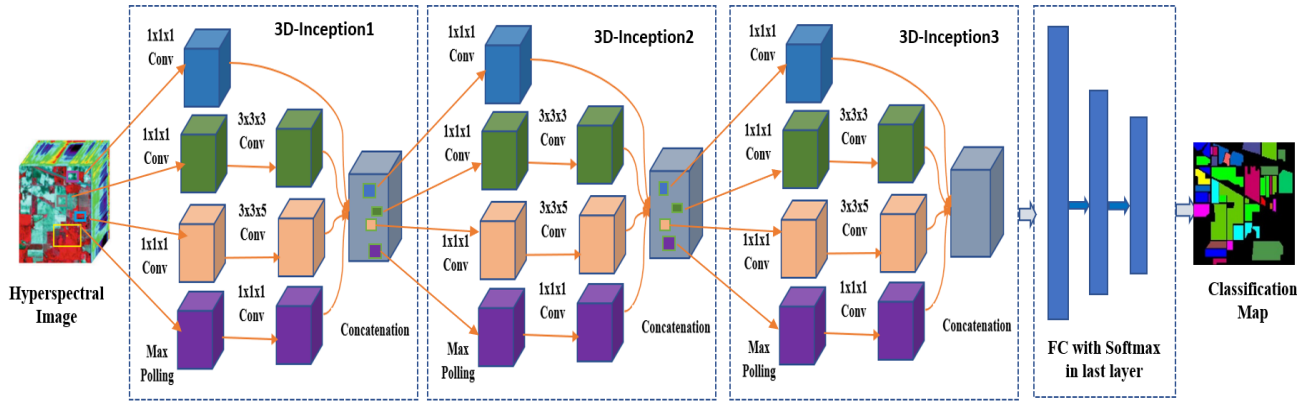


Figure. 3 Overview of the proposed 3D-Inception CNN (3D-ICNN) model

W, H shows image's spatial height and width, and spectral bands is denoted by B . Before processing, every CNN model requires a normalized image. To reduce spectral correlation and redundancy, PCA is first applied to the original hyperspectral image.

In the 3D-ICNN model, we introduced three inception layers one after another to extract deep spatial spectral features by providing small (25×25) cubes of the original dataset. This is because there is a strong correlation between a particular pixel and its surrounding pixels. The (25×25) dimension was chosen to provide adequate headroom for the 3D-Inception layers to employ various sized filters. Every 3D-Inception layer is made up of multiple convolution and pooling filters of varying sizes, whose proportions with regard to one another are determined by the original GoogLeNet structure. Inception modules employ ($1 \times 1 \times 1$) convolution filters to drastically decrease the number of parameters needed to conduct more complex processes such as (3×3) convolution or (5×5) Max Pooling. This method adds more learning parameters and enables for the use of several convolution and pooling filters in a single layer, allowing the network to use the best of them all. As a result, the network's depth is decreased, and the problem of overfitting is avoided.

As shown in Fig. 3, in each inception layer, 3D-Inception module consists of input layer, ($1 \times 1 \times 1$) convolution layer, ($3 \times 3 \times 3$) convolution layer, ($3 \times 3 \times 5$) convolution layer, max-pooling layer and concatenation layer. Initially 3D image path size of ($25 \times 25 \times 30$) is taken as input to fed to three ($1 \times 1 \times 1$) convolution layers (C_{11}, C_{21} and C_{31}) with three sets of filters $K_{11} = 32, K_{21} = 48$ and $K_{31} = 8$ respectively and to max pooling layer (P) with strides of ($1 \times 1 \times 1$). The features extracted after ($1 \times 1 \times 1$) convolution operation from the layers C_{21}, C_{31} , and P_1 are fed to ($3 \times 3 \times 3$), ($3 \times 3 \times 5$) and ($3 \times 3 \times 3$) convolution layers respectively to get more characteristics from C_{22}, C_{32} ,

and C_p convolution layers with filters $K_{22} = 64, K_{32} = 16$ and $K_p = 16$. The features extracted from the layers C_{11}, C_{22}, C_{32} , and C_p are given to concatenation layer ($Conc_1$) to combine the extracted features to fed to the second 3D-Inception module. In the second 3D-Inception module, the resultant $Conc_1$ feature map is taken as input to fed to three ($1 \times 1 \times 1$) convolution layers same as in the first 3D-Inception module with the same filters and max pooling layer with strides of ($1 \times 1 \times 1$). The features extracted after ($1 \times 1 \times 1$) convolution operation from the layers C_{21}, C_{31} , and P are fed to ($3 \times 3 \times 3$), ($3 \times 3 \times 5$) and ($3 \times 3 \times 3$) convolution layers respectively to get more features. The features extracted in second Inception module are given to concatenation layer ($Conc_2$) to combine the extracted features to fed to the third 3D-Inception module. The same process is repeated with $Conc_2$ feature map in third 3D-Inception module to generate $Conc_3$ feature map for extracting more discriminative context features. Convolution with different spatial context kernel sizes is employed to extract features with different characteristics.

For classification, the extracted feature map $Conc_3$ is flattened to send to the fully connected layers fc_1, fc_2 and fc_3 . In each fully connected layer, the activation function of each neuron is computed as Eq. (1):

$$Act_i(fc) = g(w_i(fc) * act_{i-1}(fc) + b_i) \quad (1)$$

Where, $w_i(fc)$ is the weighted sum of the preceding layer's inputs and b_i is the bias. The ReLU activation function is represented by $g(\cdot)$.

Finally, a soft-max probabilistic model is used to classify the data. Let $L = [L_i]_i$ be a number between 1 and n , and implies learned features after the entire model has been applied.

$$Smax(L)_i = \frac{e^{L_i}}{\sum_{j=1}^k e^{L_j}} \text{ for } i = 1, 2, 3, \dots, n \quad (2)$$

Table 1. Description of popularly used benchmark hyperspectral image datasets

Parameters	Ahmadabad-1	Ahmadabad-2	Salinas	Pavia University	Indian Pines
Spatial Dimension	300 × 200	300 × 200	512 × 217	610 × 340	145 × 145
No. of Spectral Bands	351	370	200	115	200
No. of Classes	5	7	16	9	16
Wavelength Range	0.37–2.5 μm	0.37–2.48 μm	360–2500 μm	0.43–0.86 μm	0.4–2.5μm
Sensor	AVIRIS-NG	AVIRIS-NG	AVIRIS	ROSIS	AVIRIS

Table 2. Classification accuracies (in %) on salinas, pavia university, and indian pines datasets

Model	Salinas			Pavia University			Indian Pines		
	OA	AA	Kappa	OA	AA	Kappa	OA	AA	Kappa
2D-CNN [8]	94.94	94.63	94.23	93.18	92.86	92.03	91.69	88.29	90.65
3D-CNN [9]	96.98	97.07	96.38	96.54	96.12	95.53	95.14	94.59	93.99
HybridSN [12]	99.85	99.59	99.52	99.93	99.03	99.81	99.22	98.56	99.12
Hybrid-RI [15]	95.33	94.89	95.01	95.31	95.02	94.99	90.57	90.21	90.16
AI-Net [16]	99.64	99.52	99.21	99.42	99.51	99.22	99.14	99.47	99.00
SyCNN [17]	99.52	99.41	98.86	99.91	98.95	98.86	97.06	96.13	95.6
Proposed Method	99.86	99.63	99.57	99.95	99.24	99.85	99.89	99.62	99.82

Table 3. Classification accuracies by reducing the amount of training samples

Dataset	5 % Training Data			10% Training Data		
	OA	AA	Kappa	OA	AA	Kappa
IP	95.31	95.02	94.73	99.00	98.52	98.68
PU	98.21	97.52	97.72	99.23	98.05	98.47
SA	98.62	98.01	98.34	99.06	98.41	98.21

Table 4. Classification accuracies (in %) on the AH1, and AH2 datasets

Model	AH1 Dataset			AH2 Dataset		
	OA	AA	Kappa	OA	AA	Kappa
3D-CNN [9]	80.99	82.13	78.17	70.06	69.30	67.93
HybridSN [12]	85.69	85.03	83.79	79.55	76.71	75.82
AI-Net [16]	85.23	84.72	83.51	79.83	77.53	75.79
SyCNN [17]	84.72	83.98	83.46	78.62	75.85	74.89
Proposed Method	86.25	86.98	84.94	80.30	77.62	76.99

For the HSI data, this is the Soft - max function model. Finally, the argmax (arguments of maximum) function. It identifies places in a function's region where the functional parameters are maximum. From n number of HSI class labels from $\Phi = \{1, 2, 3, \dots, n\}$, the class assignment can be done using Eq. (2).

$$Class (X_i) = argmax\{Smax(L)_i\} \tag{3}$$

4. Experimental study and analysis

4.1 Datasets description

To assess the proposed 3D-ICNN model performance, an experimental study was conducted

on HSI datasets, including Salinas (SA), Pavia University (PU), and Indian Pines (IP).

In addition, new datasets Ahmedabad1 (AH1), and Ahmedabad2 (AH2) have been used to check the robustness of the proposed model. These datasets are collected by the ISRO with AVIRIS-NG sensor, India [25]. The IIST dataset has 202 x 153 samples and 138 spectral bands, with 6 classes in the ground truth. The AH1 dataset has 351 spectral bands with a size of 300 x 200 pixels and the ground truth contains 5 classes. The AH2 dataset has 370 spectral bands with a size of 300 x 200 pixels and the ground truth contains 7 classes. The details of these data sets are described in Table 1.

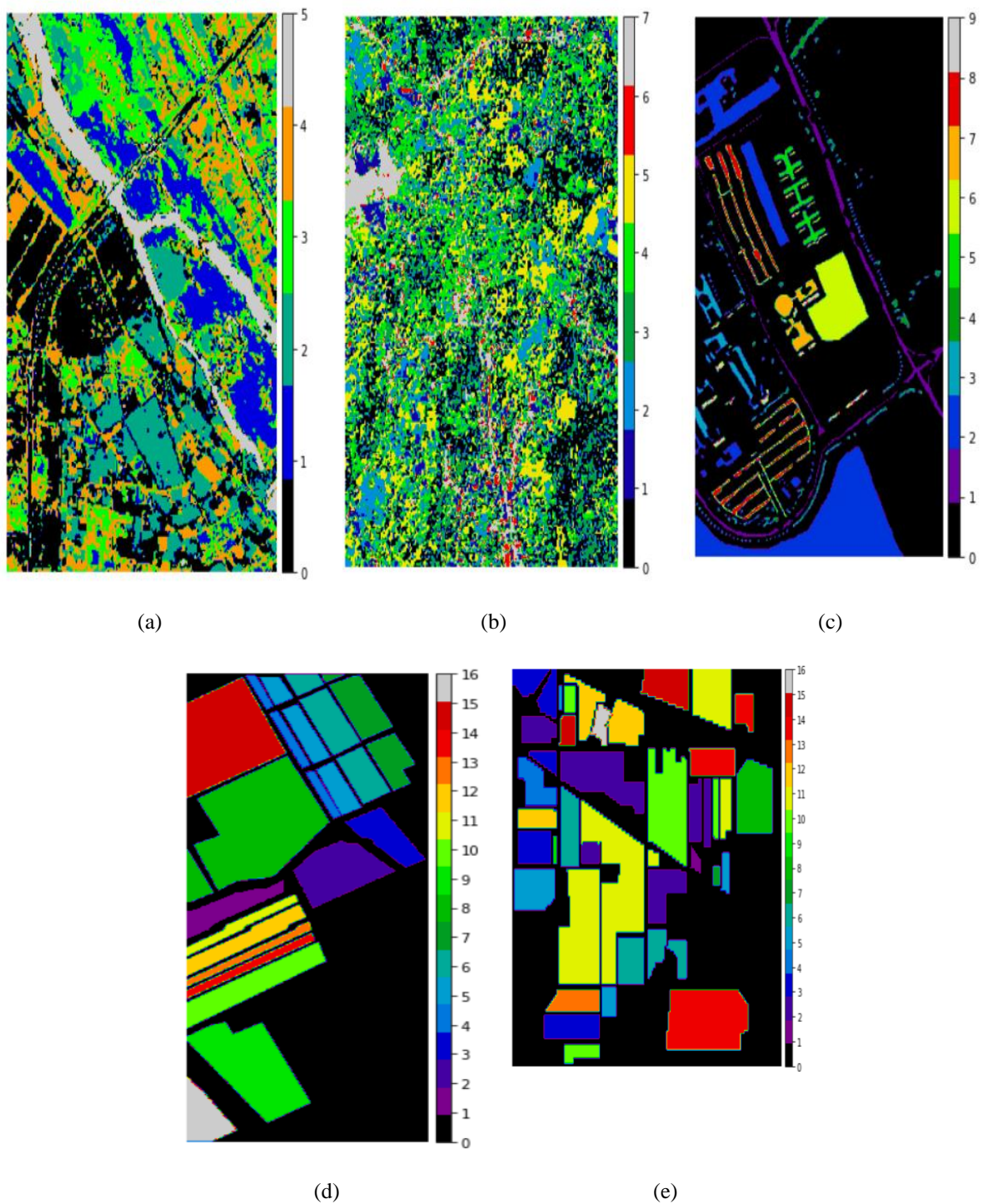


Figure. 4 Classification maps of HSI: (a) AH1, (b) AH2, (c) PU, (d) SA, and (e) IP

4.2 Experimental setup

The experiments are conducted on the google cloud with Graphical Processing Unit (GPU) having 25.51 GB RAM. To test the proposed method's generalization ability, randomly 20% selected as training set and 80% as test set from each dataset. Adam optimizer is used in the optimization process, along with a categorical cross-entropy with learning

rate 10^{-4} and decay 10^{-6} . The model has been trained for 100 epochs with batch normalization size 32. The experiments are repeated 10 times on each data set and the average results are presented.

4.3 Results and analysis

To test efficacy of the presented method, we employed conventional assessment measures: kappa

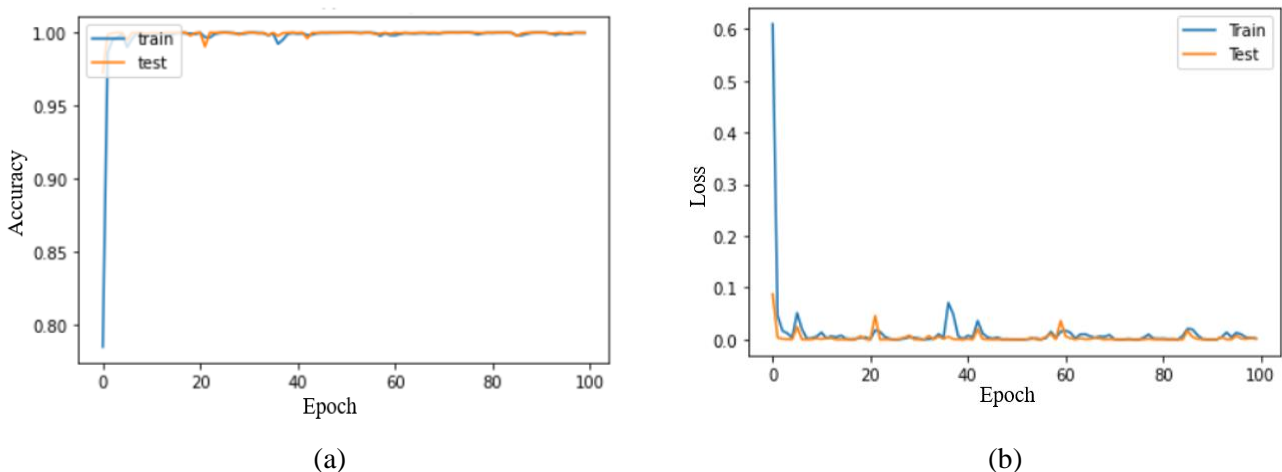


Figure. 5 Proposed model’s training and testing over the IP dataset: (a) Accuracy and (b) Loss

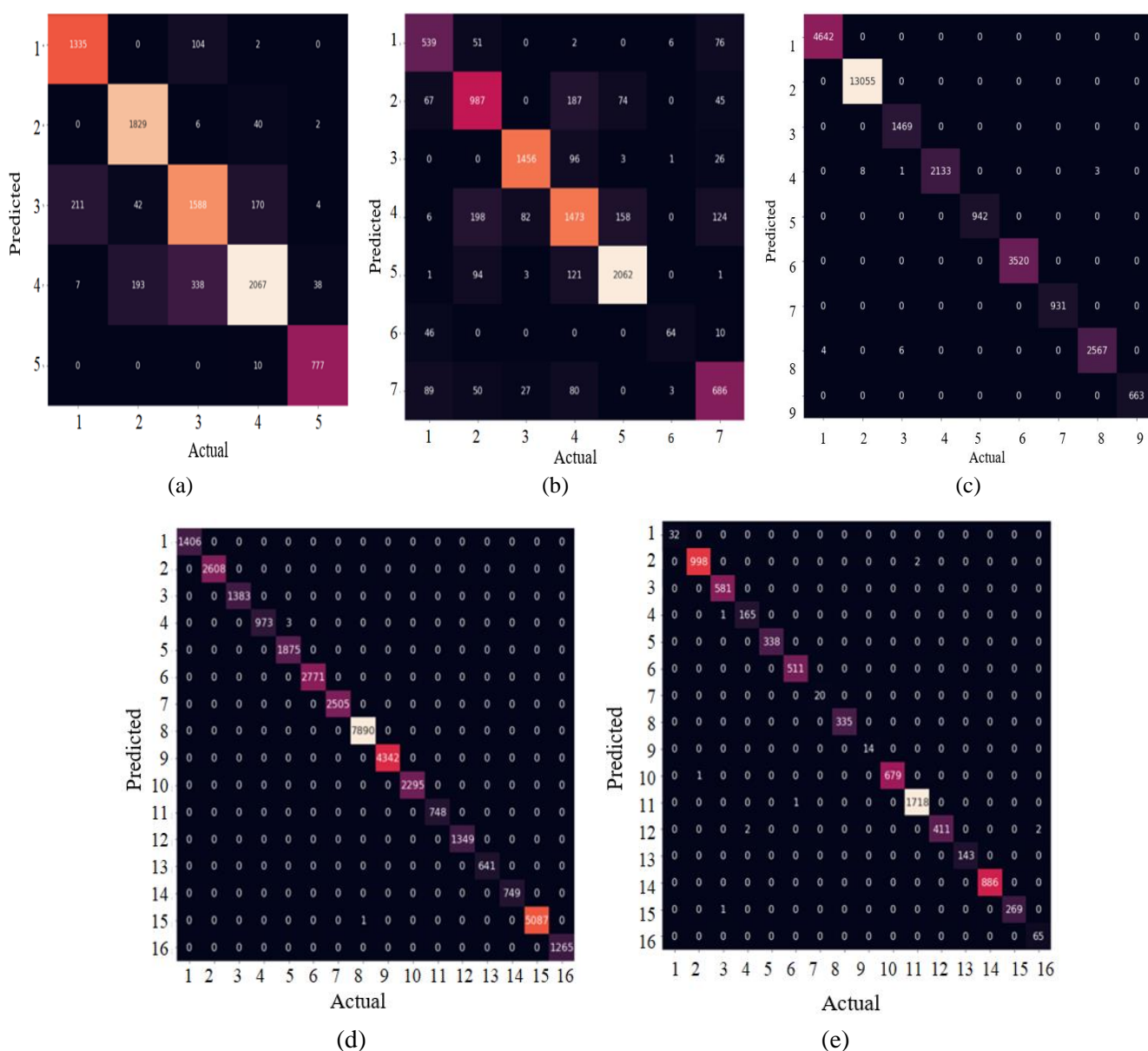


Figure. 6 Confusion matrices of HSI: (a) AH1, (b) AH2, (c) PU (d) SA, and (e) IP

(K) coefficient, average accuracy (AA), and overall accuracy (OA). Kappa Coefficient can be defined as the measurement of the agreement between the actual values and predicted values. AA is represented as the average of the class-wise accuracy. OA can be described as the ratio of correctly classified samples and total samples. The presented 3D-ICNN model's results are compared to those of other HSI classification models, such as the 2DCNN [8], 3DCNN [9], HybridSN [14], Hybrid-RI [15], AI-Net [16], and SyCNN [17]. Table 2 provides the classification accuracy obtained by all these methods. It is observed that the proposed method achieved 99.86% overall-accuracy, 99.63% average-accuracy on Salinas, 99.95% overall-accuracy, 99.24% average-accuracy on Pavia, and 99.89% overall-accuracy, 99.62% average-accuracy on IP datasets. Table 2 shows that the proposed model's classification accuracy is better than that of different approaches on the benchmark datasets in terms of evaluation metrics such as OA, AA, and Kappa. The accuracies for 2DCNN [8], 3DCNN [9], HybridSN [12], Hybrid-RI [15], AI-Net [16], and SyCNN [17] methods are taken from their respective papers, and the results are computed using publicly available code for the comparative methods. When compared to the state-of-the-art methods the proposed model's kappa, average-accuracy, and overall-accuracy values are significantly better. Compared with the HybridSN [12], the proposed method produces slightly better results in almost all cases. The HybridSN model used 30% samples for training. In the proposed method 20% samples used from each class to train the model.

Table 3 shows the efficacy of proposed approach according to size of the training data. With fewer training data, the presented model achieved a better accuracy of classification than state-of-the-art methods. Experiments on other new datasets, AH1, and AH2, are conducted to verify efficiency and robustness of the approach. We have compared our method with 3D-CNN [10], HybridSN [12], AI-Net [16], and SyCNN [17] models using their publicly available code. Other methods could not be compared as their code is not available. It is observed that the proposed method achieved 86.25% overall-accuracy, 86.98% average-accuracy on AH1, and 80.30% overall-accuracy, 77.62% average-accuracy on AH2 datasets. Table 4 shows that the presented model achieved 2% to 3% improvement on the new datasets.

The classification maps of the datasets such as AH1, AH2, PU, SA, and IP are generated by the proposed model are depicted in Fig. 4. When comparing the classification maps produced by the proposed approach to those produced by state-of-the-

art methods, some regions in the proposed method's classification maps are less noisy. The proposed method's accuracy and loss convergence over 100 epochs of training and validation sets are shown in Fig. 5. It can be observed that the convergence occurs in around 50 epochs, demonstrating the method's rapid convergence, and the confusion matrices are presented in Fig. 6, which summarize the number of accurate and incorrect predictions with count values and break them down by each class.

5. Conclusion

In this work, it is recommended that a unique 3D-Inception CNN model for HSI classification be employed, which has the potential to perform very well even with limited training data. This model takes advantage of a variety of filter types in each layer and efficiently learns features. It is not very deep, and as a result, it is not prone to becoming overfit. The results obtained from experiments conducted on the benchmark datasets demonstrate that the proposed method outperforms current approaches in terms of accuracy and precision. Further, the presented model efficiency is also evaluated on new datasets and it is significantly superior to the HybridSN and AI-Net models. It is the goal of this study in the future to reduce the training time of each inception module, which will in turn reduce the overall training time of the model.

Conflicts of Interest

“The authors declare no conflict of interest.”

Author Contributions

The paper Conceptualization, Murali and T. Hitendra; methodology, Murali; software, T. Hitendra; validation, Murali, T. Hitendra, and C. Shobha; formal analysis, T. Hitendra; investigation, C. Shobha; resources, T. Hitendra; writing—original draft preparation, Murali; writing—review and editing, Murali; visualization, Murali; supervision, T. Hitendra; project administration, T. Hitendra;

Acknowledgments

The authors would like to thanks to Dr. N. Rama Rao, Professor, Indian Institute of Space Science and Technology (IIST), Trivendrum for providing Ahmadabad City-1 (AH1) and Ahmadabad City-2 (AH2) datasets.

References

- [1] L. Mou, P. Ghamisi, and X. Zhu, “Deep recurrent neural networks for hyperspectral image classification”, *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 55, No. 7, pp. 3639-3655, 2017.
- [2] L. Zhang, L. Zhang, and B. Du, “Deep learning for remote sensing data: A technical tutorial on the state of the art”, *IEEE Geoscience and Remote Sensing Magazine*, Vol. 4, No. 2, pp. 22-40, 2016.
- [3] J. Ma, Y. Ma, and C. Li, “Infrared and visible image fusion methods and applications: A survey”, *Information Fusion*, Vol. 45, pp. 153-178, 2019.
- [4] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, “Deep learning classifiers for hyperspectral imaging: A review”, *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 158, pp. 279-317, 2019.
- [5] M. Hamouda, K. S. Ettabaa, and M. S. Bouhleb, “Smart feature extraction and classification of hyperspectral images based on convolutional neural networks”, *IET Image Processing*, Vol. 14, No. 10, pp. 1999-2005, 2020.
- [6] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, “Deep convolutional neural networks for hyperspectral image classification”, *Journal of Sensors*, Vol. 2015, 2015.
- [7] H. Wu and S. Prasad, “Convolutional recurrent neural networks for hyperspectral data classification”, *Remote Sensing*, Vol. 9, No. 3, p. 298, 2017.
- [8] J. Yang, Y. Q. Zhao, and J. C. Chan, “Learning and transferring deep joint spectral-spatial features for hyperspectral classification”, *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 55, No. 8, pp. 4729-4742, 2017.
- [9] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, “Deep feature extraction and classification of hyperspectral images based on convolutional neural networks”, *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 54, No. 10, pp. 6232-6251, 2016.
- [10] X. Cao, F. Zhou, L. Xu, D. Meng, Z. Xu, and J. Paisley, “Hyperspectral image classification with Markov random fields and a convolutional neural network”, *IEEE Transactions on Image Processing*, Vol. 27, No. 5, pp. 2354-2367, 2018.
- [11] Y. Xu, L. Zhang, B. Du, and F. Zhang, “Spectral-spatial unified networks for hyperspectral image classification”, *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 56, No. 10, pp. 5893-5909, 2018.
- [12] Q. Wang, J. Wan, and Y. Yuan, “Deep metric learning for crowdedness regression”, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 28, No. 10, pp. 2633-2643, 2017.
- [13] Q. Wang, J. Gao, and Y. Yuan, “Embedding structured contour and location prior in siamese fully convolutional networks for road detection”, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 19, No. 1, pp. 230-241, 2017.
- [14] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, “HybridSN: Exploring 3-D-2-D CNN feature hierarchy for hyperspectral image classification”, *IEEE Geoscience and Remote Sensing Letters*, Vol. 17, No. 2, pp. 277-281, 2019.
- [15] B. Alotaibi and M. Alotaibi, “A Hybrid Deep ResNet and Inception Model for Hyperspectral Image Classification”, *PFJ—Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, Vol. 88, No. 6, pp. 463-476, 2020.
- [16] H. Zhang, Y. Liu, B. Fang, Y. Li, L. Liu, and I. Reid, “Hyperspectral Classification Based on 3D Asymmetric Inception Network with Data Fusion Transfer Learning”, *ArXiv Preprint ArXiv:2002.04227*, 2020.
- [17] X. Yang, X. Zhang, Y. Ye, R. Y. K. Lau, S. Lu, X. Li, and X. Huang, “Synergistic 2D/3D convolutional neural network for hyperspectral image classification”, *Remote Sensing*, Vol. 12, No. 12, p. 2033, 2020.
- [18] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition”, In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.
- [19] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, “Densely connected convolutional networks”, In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700-4708, 2017.
- [20] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions”, In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-9, 2015.
- [21] A. G. Garcia, S. O. Escolano, S. Oprea, V. V. Martinez, and J. G. Rodriguez, “A review on deep learning techniques applied to semantic segmentation”, *ArXiv Preprint ArXiv:1704.06857*, 2017.
- [22] T. Tian, C. Li, J. Xu, and J. Ma, “Urban area detection in very high-resolution remote sensing

- images using deep convolutional neural networks”, *Sensors*, Vol. 18, No. 3, p. 904, 2018.
- [23] Y. Li, Y. Zhang, X. Huang, and J. Ma, “Learning source-invariant deep hashing convolutional neural networks for cross-source remote sensing image retrieval”, *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 56, No. 11, pp. 6521-6536, 2018.
- [24] J. Ma and J. Zhao, “Robust topological navigation via convolutional neural network feature and sharpness measure”, *IEEE Access*, Vol. 5, pp. 20707-20715, 2017.
- [25] M. K. Tripathi and H. Govil, “Evaluation of AVIRIS-NG hyperspectral images for mineral identification and mapping”, *Heliyon*, Vol. 5, No. 11, p. e02931, 2019.