

# HYB-ICNN: HYBRID INCEPTION CONVOLUTIONAL NEURAL NETWORK FOR FEATURE EXTRACTION AND HYPERSPECTRAL IMAGE CLASSIFICATION

EASALA RAVI KONDAL<sup>1</sup>, SOUBHAGYA SANKAR BARPANDA<sup>2</sup>

<sup>1</sup>School of Computer Science and Engineering, VIT-AP University, Amaravati, India.

<sup>2</sup>School of Computer Science and Engineering, VIT-AP University, Amaravati, India.

E-mail: <sup>1</sup>easala.ravikondal@gmail.com, <sup>2</sup>soubhagya.barpanda@vitap.ac.in

## ABSTRACT

The remote sensing research community has devoted particular focus to the classification of hyperspectral images (HSI). To improve the classification accuracy of hyperspectral images (HSI), deep learning-based technology has been proposed. However, it remains a challenging obstacle to achieve satisfactory classification accuracy with insufficient data for training. Therefore, a more effective neural network design needs to be devised in order to boost the effectiveness of the HSI classification function. To address this concern, this letter provides a novel Hybrid-Inception CNN (Hyb-ICNN) framework for dynamically obtaining characteristics by laying inception components in the model which can acquire better accurate properties with smaller training samples by employing volatile spatial dimension convolutional filters and dynamic CNN framework. The outcomes of the experiments show that the presented model can boost classification performance by adaptively changing the network structure. The experiments are carried out on both the new data sets and the publicly accessible benchmark data sets to expose the efficiency and durability of the proposed model. The proposed Hybrid-Inception CNN model has achieved accuracies of 80.79% on the AH1 (Ahmedabad-1) dataset, 87.98% on the AH2 (Ahmedabad-2) dataset, 99.99% on the PU (Pavia University), 99.99% on the SA (Salinas), and 99.92% on the IP (Indian Pines) dataset. Empirically, it has been demonstrated that the presented model succeeds over the remaining state-of-the-art approaches in terms of classification accuracy.

**Keywords:** *Hyperspectral Image, Inception Network, Convolutional Neural Network, Feature Extraction, Classification*

## 1. INTRODUCTION

A hyperspectral image (HSI) is a 3D (3-dimensional) cube recorded with a spectrometer that incorporates spectral and image information to show the emission properties and spatial geometric connection of the object [1-2]. HSI is more spectral band-rich and has a greater spectral resolution than multispectral remote sensing images and conventional RGB images. HSI is widely employed in a variety of industries, including agriculture, geological exploration, environment, and ecology, due to its rich spatial and spectral information [3-7]. Land cover and land use classification are frequently required in these applications [8].

The spectral properties of HSI pixels are the primary criterion for categorization in the initial HSI classification techniques [9]. The spectral

characteristics of ground objects may change as a result of spectral fluctuations and distortion. As a result, depending solely on spectral data could lead to incorrect classification. A variety of spatial-spectral-based feature retrieval and classification approaches have been proposed by researchers to utilize full spatial data. These approaches typically outperform traditional spectral-based classification techniques in terms of reliability and precision. Nevertheless, such approaches are frequently used for particular intended features. It may result in poor classification efficiency if certain characteristics are not appropriate for particular application criteria.

Deep learning (DL) approaches are widely used for feature retrieval and classification of HSIs because of their ability to dynamically acquire suitable features for particular application challenges. However, a significant number of labeled samples are typically needed to train the

model to reach the comparable rate of classification. The typical method for obtaining sample labels for HSI is field research or straight interpretation by sight from images with a high resolution. The quantity of training samples is significantly reduced by the cost and time involved in conducting field studies, which are frequently utilized to generate more accurate labeling. Real-world scenarios frequently include a small-sample classification challenge, and it can be challenging to get enough training data to properly satisfy a DL network's sample needs. Researchers have developed a number of small-sample DL approaches for HSI classification to address this issue, including small-scale learning [10-11], lightweight networks [12-14], and data augmentation [15].

Approaches for data augmentation could be employed to boost the quantity of labeled samples. Adding random noise, flipping, translation, clipping, rotating are common data augmentation techniques that can improve the quantity and variety of samples. To increase the sample size, Li et al. [15] suggested a pixel-block pair (PBP) technique, in which each pixel is formed into a pixel block, and a PBP is applied for training. A random occlusion data augment strategy for training CNN was put out by Haut et al. [16]. It created training images with varying degrees of occlusion by randomly occluding pixels in various rectangular spatial locations. Methods for enhancing the data can increase the quantity of training samples, which enhances classification performance. Yet, these techniques are vulnerable to adding noisy data during the data augmentation process, which could result in unstable classification performance.

Various lightweight networks are suggested [12-14] to decrease reliance on the quantity of training data. The lightweight network employs light network layout principles including depth-separable convolution [14], clustered convolution [17], and other lightweight convolution techniques that can minimize the amount of computing required for convolution as well as the network parameters. Yet, the lightweight network's capacity for discrimination may also be degraded.

The objective of few-shot learning is to discover new classes from a small number of labelled examples [10]. For small-sample HSI classification, Liu et al. [11] introduced a deep few-shot learning (DFSL) approach that builds a deep 3D ResNet to gain a metric space where samples from the same class are near and those from distinct classes are separated. The fundamental drawback of these methods is that because the public HSI datasets are

so limited, it is impractical to build a deep method since more training instances are needed.

We proposed a hybrid inception CNN (Hyb-ICNN) model for HSI classification and to resolve the issues mentioned earlier. The subsequent summarizes the contributions of the work:

- Proposed a hybrid inception CNN that can adapt the convolution kernel size to various locations in order to enhance the performance of HSIs classification.
- The proposed approach uses the CNN structure with inception components to address the overfitting issue.

Here is how the remainder of the paper is laid out: The review of the literature for the proposed work is mentioned in Section 2, and the model architecture is presented in Section 3. The experimental research, outcomes and conversations of the proposed framework are stated in Section 4. Finally, Section 5 presents the inferences.

## 2. RELATED WORK

It is widely acknowledged in the research community that classifying hyperspectral images is a significant problem in the field. However, earlier research mostly focused on traditional computational techniques. We quickly cover the most recent deep learning pixel-based models in this section.

A 2D-CNN pixel--based method was presented in [18] to retrieve the spectral-spatial data embedded in HSIs, where 2D-CNN has been used to examine the band selection outcomes. A variety of pixel--based techniques were put forth to combine specific 2D-CNN models for the classification of HSI based on the band selection outcomes. As an illustration, Liu et al. [19] used deep belief networks to retrieve deep spectral characteristics. A 2D-CNN adaptive HSI classification model has been presented by Zhu [20] and is composed of adaptable convolutions and down sampling which integrate the underlying information of every input sequence in an adaptive manner. Through the addition of three feature regions based on spectrum information to spatial regions, in [21], Han developed a 2D-CNN method to assess spectral-spatial characteristics. Zhao and Du [22] gave an example of a spectral-spatial feature-based categorization method that utilizes a 2DCNN to collect spatial data and a linear local

distinction encase to find spectral data. Although these methods might produce models with comparable performance, they required a huge training dataset, which is not feasible for the majority of real-world implementations.

There are techniques that account the spatial-spectral features of the HSI data. As CNN models are so flexible, a wide range of methods and networks can be created to do spectral-spatial analysis. Chen [23] presented a deep 3D-CNN model comprised of multiple 3D convolutional layers that extracted spectral-spatial characteristics for classification. Similar to this, Lee [24] proposed a deep network with such an emphasis on 3D data. The proposed model is capable of accurately capturing alterations in local impulses stated in spectral-spatial data. In order to evaluate a series of volumetric models of the HSI, Hamida [25] devised and assessed a set of 3D schemes that combine the conventional 3D convolution operations to enable a joint spatial-spectral information analysis. Kanthi [26] presented a 3D-CNN method that splits HSI into 3D regions and pulls deep spectral-spatial data for HSI classification.

However, certain hybrid approaches incorporated 2D-CNNs and 3D-CNNs. For example, Roy [27] proposed HybridSN model that merges spectral and spatial 3D-CNN with spatial 2D-CNN. Raviteja [28] established hierarchical image fusion model for HSI classification to merge specified spectral features into image groups. Wan [29] presented multiscale graph convolutional network for irregular image region convolution for HSI classification. Meng [30] employed connections with feed-forward shortcuts to access all convolutional layers' hierarchical input to create a dense multiscale hybrid network and a multi-scale HSI spatial and spectral HybridCNN was presented by Mohan [31] for Classification tasks. Kanthi [32] introduced a deep CNN model that uses three distinct multi-scale spatial-spectral regions to retrieve features from both the spatial and spectral bands. Inception and ResNet network designs were created by Bandar Alotaibi [33], integrating the fundamental idea of ResNet, the use of residual blocks and the input sequence of the Inception model that its topology is preferable to each one individually in terms of accuracy. Yet, certain data sets have very poor accuracy. An AI-Net that could apply deep learning to discover reflective characteristics was introduced by Haokui Zhang [34]. This AI-Net also included a transfer learning of data convergence model for more precise model initialization and faster training. However, the model improvement must use some techniques to get

beyond the data's HSI categorization imbalance. In feature learning, Yang [35] created a SyCNN model that incorporates 2D and 3D CNNs with a hybrid component that blends spectral-spatial HSI data with 3D attention mechanisms. Kanthi [36] presented 3D-ICNN that uses 3D-CNN and inception blocks to retrieve features with varying filters. However, optimal design of models is required to enhance the feature extraction procedure.

Along with to the specified samples, no more than 10% of the samples could be utilized for HSIs classification training. Nevertheless, deep learning models typically include an extensive amount of training variables. Deep models will likely overfit if there is a small enough data collection. Deep network structures have recently been proposed, and even with more than 100 layers, they are fairly profound. Examples are DensNet [37] and ResNet [38]. The networks, like Coco and Imagenet, operate in vast volumes of data. However, these networks prove too complicated for HSI imaging. These models have overfitting obstacles since the training data are minimal.

### 3. PROPOSED METHODOLOGY

This section focuses primarily on the CNNs history, Inception module, and proposed method.

#### 3.1 CNN

CNN employs a multiple-layer trained structure composed of stacking, pooling layers and uncertainty for learning properties including patterns and edges and advanced features that contain additional perceptive input [39-41]. The usual construction of a CNN is shown in figure 1.

In contrast to the conversion layer, that may consist of 2-dimensional size  $p \times q$  feature conversions in the convolution layer, each hidden layer component is related by shared weights to the local receptive field across the input rather than having completely attached to the input. The convolution layer is responsible for producing an intensity of the  $Z_i$  input feature maps using a  $F_i$  kernel with the dimensions  $f \times f \times n$  and a nonlinear unit activated function. This concentration is then applied to the  $Z_i$  output maps. Following the completion of the stacked layers comes the application of completely connected layers with Softmax so that classification labels can be predicted.

#### 3.2 Inception Network

A standard CNN model consists of multiple layers of convolution preceding one or more fully connected (FC) layers. The FC layer comprises a

multilayer neural network in the usual sense. The outcome is saved in the bottommost FC layer.

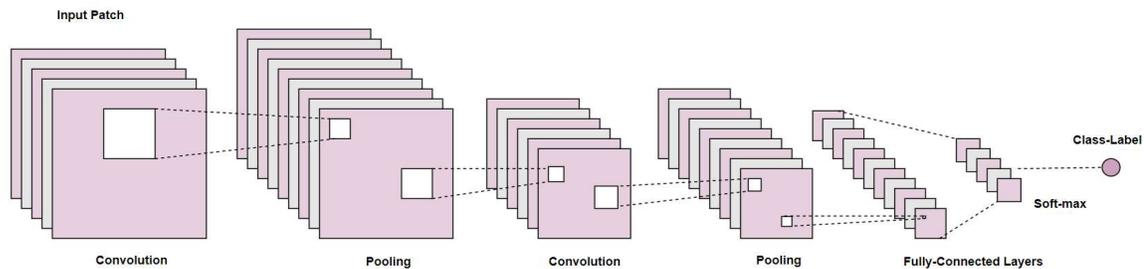


Figure 1: Illustration of A Basic CNN Structure.

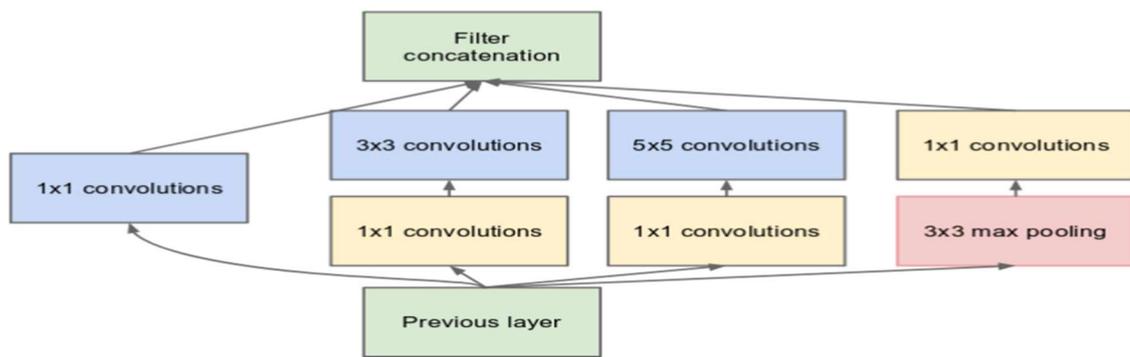


Figure 2: The Inception Model's Main Building Block.

The convolutional layer uses several filters to condense the input image, while the pooling layer down samples the data. The pooling layer often has max-pooling and average-pooling functions. The CNN converts the input image over numerous stacked layers, from the initial basic pixels to the final class values. Multiple semantic approaches to segmentation are built on top of CNN designs.

The powerful and sophisticated deep learning network Goog-LeNet [42] is adopted. As shown in Fig. 2, GoogleNet is a deep CNN design created by Google researchers and released to the public in 2014. This design made the top five ranking of the ILSVRC with a 93.3 percent accuracy rate. The GoogleNet is incredibly intricate with 22 layers and a special structural component known as the Inception model. This design makes use of a network in a pooling layer, a network layer, and big and tiny convolution layers that are computed concurrently rather than in the traditional sequence. After that, the dimensionality is decreased using a 1 x 1 convolution process. The number of parameters and operations has been significantly reduced as a result of the parallelism and dimensionality reduction incorporated in this design, leading to considerable memory and processing decreases [42].

### 3.3 Hyb-ICNN model

The proposed Hyb-InceptionCNN (Hyb-ICNN) model is explained in this section. The model uses 3D patches as input to produce deep spatial features from an HSI data cube for pixel classification, as shown in Figure 3. Consider an HSI  $X$ , which is a 3D cube with dimensions  $W \times H \times B$ , where  $W$  and  $H$  represent the image's spatial height and width, respectively, and  $B$  represents the spectral bands. Before processing, each CNN model needs an image that has been normalized. PCA is initially employed to the initial HSI to decrease spectral correlation and redundancy. In this, we employed two inception layers sequentially in the Hyb-ICNN model to extract deep spatial spectral features by supplying small (25 x 25) cubes of the initial dataset. It is due to the fact that each pixel and the pixels around it have a strong association.

For the Hyb-Inception layers, the (25 x 25) dimension was selected to allow for the use of different-sized filters. Each Hyb-Inception layer is made of a number of convolution and pooling kernels of varied sizes, the proportions of which are dictated by the original GoogLeNet structure. Inception modules make extensive use of (1 x 1 x 1)

convolution kernels to significantly decrease the number of parameters required to conduct more sophisticated operations such as (5 x 5) Max Pooling or (3 x 3) convolution. This strategy improves the number of learning parameters and permits the use of many convolutional and pooling kernels in a single layer, allowing the model to use the best of all available filters. As a result, the depth of the network is decreased, and the overfitting issue is avoided.

As illustrated in Figure 3, the 3D-Inception module comprises of an input layer, a (1 x 1 x 1) convolution layer, a (3 x 3 x 3) convolution layer, a (3 x 3 x 5) convolution layer, a max-pooling layer, and a concatenation layer for each inception layer. Initially, a three-dimensional image path with a size of (25 x 25 x 30) is received as input and fed to three (1 x 1 x 1) convolution layers ( $C_{11}, C_{21},$  and  $C_{31}$ ) provided with three sets of filters  $K_{11} = 32, K_{21} = 48,$  and  $K_{31} = 8,$  respectively, and to a max pooling layer ( $P$ ) provided with strides of (1 x 1 x 1). The features retrieved from the layers  $C_{11}, C_{21}, C_{31},$  and  $P_1$  following the (1 x 1 x 1) convolution operation is sent to the (1 x 1), (3 x 3 x 3), (3 x 3 x 5), and (3 x 3 x 3) convolution layers, respectively, to obtain additional characteristics from the  $C_{12}, C_{22}, C_{32},$  and  $C_p$  convolution layers using filters  $K_{12} = 32, K_{22} = 48, K_{32} = 64,$  and  $K_p = 16.$  The features retrieved from the layers  $C_{22}, C_{32},$  and  $C_{42}$  are sent to the (3 x 3 x 3), (3 x 3 x 5), and (1 x 1) convolution layers, respectively, to obtain additional characteristics from the  $C_{22}, C_{32},$  and  $C_{42}$  convolution layers using filters  $K_{22} = 64, K_{32} = 16,$  and  $K_p = 16.$  The characteristics retrieved from layers  $C_{21}, C_{32}, C_{33},$  and  $C_{43}$  are combined in a concatenation layer (Conc<sub>1</sub>) before being supplied to the second Hyb-Inception module. The second Hyb-Inception module takes the

generated (Conc<sub>1</sub>) feature map as input and feeds it through three (Conc<sub>1</sub>) convolution layers using the same filters and max pooling layer as in the first Hyb-Inception module. In the second Hyb-Inception module, the same process is repeated with the (Conc<sub>1</sub>) feature map to build the (Conc<sub>2</sub>) feature map for extracting more discriminative context characteristics. Convolution with a variety of spatial context kernel sizes is used to extract features with a variety of characteristics. The extracted feature map (Conc<sub>2</sub>) is flattened and sent to the fully connected layers  $fcl_1, fcl_2,$  and  $fcl_3$  for classification. Each neuron's activation function in a fully connected layer is calculated using Eq. (1).

$$Act_x(fcl) = g(w_x(fcl) * act_{x-1}(fcl) + b_x) \quad (1)$$

Where,  $w_x(fcl)$  is the weighted total of all the inputs to the precedent layer and  $b_x$  is the bias.  $g(.)$  is a representation of the ReLU activation.

Finally, the data is classified using a soft-max probabilistic model.  $F = [F_x]x,$  where  $x$  is a positive integer between 1 and  $n,$  represents the feature representations after the entire model has been implemented, as in Eq. (2).

$$Smax(F)_x = \frac{e^{F_x}}{\sum_{y=1}^k e^{F_y}} \text{ for } x = 1, 2, 3, \dots, n \quad (2)$$

At last, there is the argmax function (maximum arguments). It establishes the location in the region of a function where the functional parameters are at their highest. Equation (2) can be used to allocate classes to a  $m$  number of hyperspectral image class labels ranging from  $Q = \{1, 2, 3, 4, \dots, m\}.$

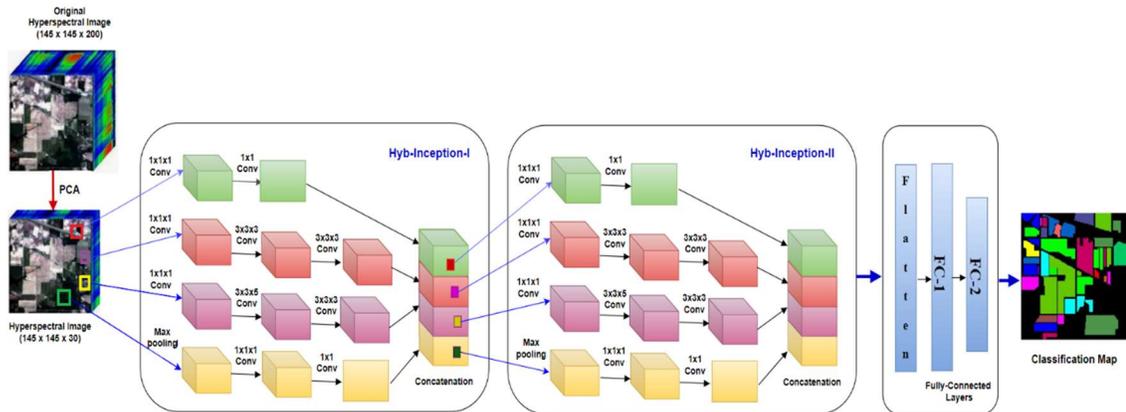


Figure. 3: Overview of the Presented Hybrid Inception CNN (Hyb-ICNN) Network.

Table 1. Description-Benchmark and Indian HSI datasets.

Parameters	SA	PU	IP	AH-1	AH-2
Sensor	AVIRIS	ROSIS	AVIRIS	AVIRIS-NG	AVIRIS-NG
Wavelength Range	360–2500 $\mu\text{m}$	0.43–0.86 $\mu\text{m}$	0.4–2.5 $\mu\text{m}$	0.37–2.5 $\mu\text{m}$	0.37–2.48 $\mu\text{m}$
No. of Classes	16	9	16	5	7
No. of Spectral Bands	200	115	200	351	370
Spatial Dimension	512 $\times$ 217	610 $\times$ 340	145 $\times$ 145	300 $\times$ 200	300 $\times$ 200

Table 2. Proposed Model Classification Accuracies (In %) on Benchmark Datasets.

Model	SA			PU			IP		
	AA	OA	Kappa	AA	OA	Kappa	AA	OA	Kappa
2D-CNN	95.95	95.64	95.25	92.29	92.32	92.12	90.10	88.31	90.61
3D-CNN	96.98	97.98	97.02	97.02	97.32	96.42	96.24	95.36	94.85
Hybrid-RI	95.46	95.21	95.53	95.42	95.31	94.98	91.12	91.34	91.31
SyCNN	99.75	99.52	98.98	99.85	99.02	98.92	97.52	96.75	96.04
AI-Net	99.71	99.68	99.34	99.28	99.71	99.32	99.51	99.64	99.15
HybridSN	99.85	99.59	99.52	99.93	99.03	99.81	99.22	98.56	99.12
3D-ICNN	99.86	99.63	99.57	99.95	99.24	99.85	99.89	99.62	99.82
Proposed Method	99.99	99.99	99.98	99.98	99.99	99.99	99.95	99.98	99.92

Table 3. Proposed Model Classification Accuracies (In %) with Few Training Data.

Dataset	5 % Training Samples			10% Training Samples		
	AA	OA	Kappa	AA	OA	Kappa
IP	96.87	95.37	95.43	98.82	99.34	98.76
PU	97.98	98.52	97.93	98.84	99.92	98.91
SA	98.85	98.89	98.62	98.85	99.35	98.81
AH-1	80.46	80.85	78.57	82.75	82.83	81.62
AH-2	73.05	72.92	71.84	75.72	75.99	73.92

Table 4. Proposed Model Classification Accuracies (In %) on Indian Datasets.

Model	AH-1			AH-2		
	AA	OA	Kappa	AA	OA	Kappa
3D-CNN	80.99	82.13	78.17	70.06	69.30	67.93
HybridSN	85.69	85.03	83.79	79.55	76.71	75.82
AI-Net	85.23	84.72	83.51	79.83	77.53	75.79
SyCNN	84.72	83.98	83.46	78.62	75.85	74.89
3D-ICNN	86.25	86.98	84.94	80.30	77.62	76.99
Proposed Method	87.05	87.86	86.96	80.57	80.24	78.43

#### 4. EXPERIMENTAL RESULTS

This section provides the datasets descriptions, experimental setups, and experimental evaluation of Hyb-ICNN model. The details are explained in the following subsections.

##### 4.1 Datasets and setup

An experimental investigation was carried out using three publicly available HSI datasets: PU (Pavia-University), SA (Salinas), and IP (Indian-Pines). Furthermore, two new Indian datasets, AH1 (Ahmedabad-1) and AH2 (Ahmedabad-2), had been employed to assess the potency of the presented method's performance. These datasets were gathered by the ISRO using AVIRIS-NG sensor [43]. The details are provided in Table 1.

The experiments are carried out using a GPU with 25-GB RAM on the Google Cloud. The established network Hyb-ICNN is evaluated by randomly picking 20% of samples as train and 80% as test set from every dataset. In the optimization process, the Adagrad optimizer is employed, as well as a categorical cross-entropy with decay (1e-06) and learning rate (0.001). The approach was trained for 50 epochs using batch size of 32. On each data set, the experiments are recited 8 times and the average outcomes were reported.

##### 4.2 Classification Results and Analysis

The kappa (K), AA (Average-Accuracy), and OA (Overall-Accuracy) had been incorporated to assess the efficacy of the provided network. Contemporary HSI classification models, like 3D-CNN, 2D-CNN, HybridSN, Hybrid-RI, AI-Net, SyCNN, MS-3DCNN, 3D-ICNN, are compared to

the results of the presented Hyb-ICNN model. The classification accuracy attained by all these approaches is displayed in Table 2 and it demonstrates that the presented method's classification efficiency is superior to that of alternative models on the contrast datasets. Table 3 indicates the efficiency of the presented approach based on the amount of the training data. The given model outperformed state-of-the-art approaches in classification efficacy with less training samples.

The studies on two new Indian datasets, AH1 and AH2, are being carried out to ensure that the method is efficient and resilient. We used publicly available code to compare our method against models such as 3D-CNN, HybridSN, HybridCNN, MS-3DCNN, and 3D-ICNN. In order to compare other methods, their code was unavailable. The proposed method achieved better accuracy on new dataset and the provided model improved by 2% to 3%, as shown in Table 4.

The comparison of various accuracies of proposed model with current methods for bench mark datasets and Indian datasets are depicted in Figure 4. Figure 5 shows the classification maps created by the proposed model for all datasets. It's easier to compare the proposed method's classification maps to those made by other methods. Some parts of the presented model's maps are lesser noisy. The accuracy and loss progress of the proposed technique throughout 100 epochs of validation and training samples are presented in Figure 6. The method's rapid convergence could be identified in fact that the progress occurs in roughly 50 epochs.

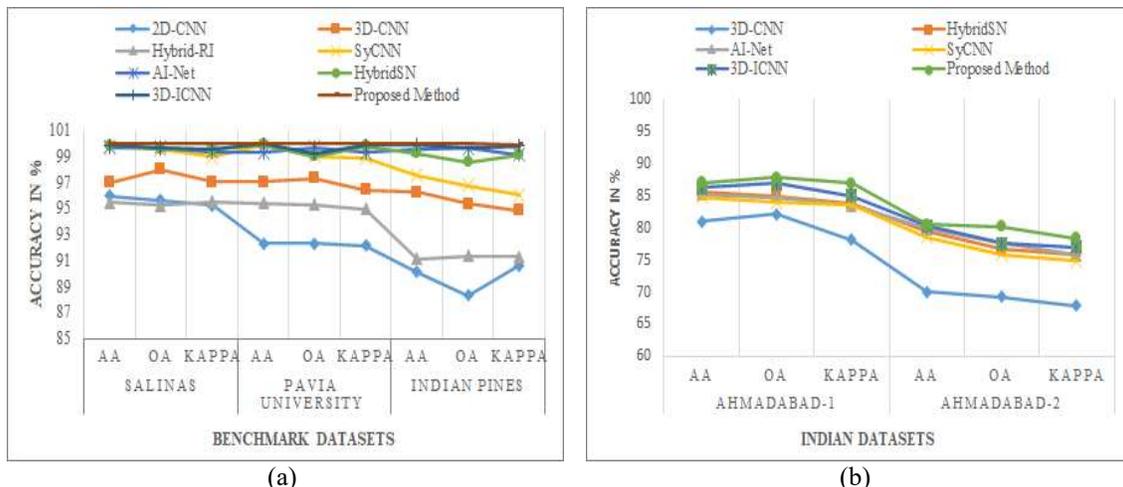


Figure 4. Comparison of various accuracies of presented model with state-of-art approaches: (a) Benchmark datasets (b) Indian datasets.

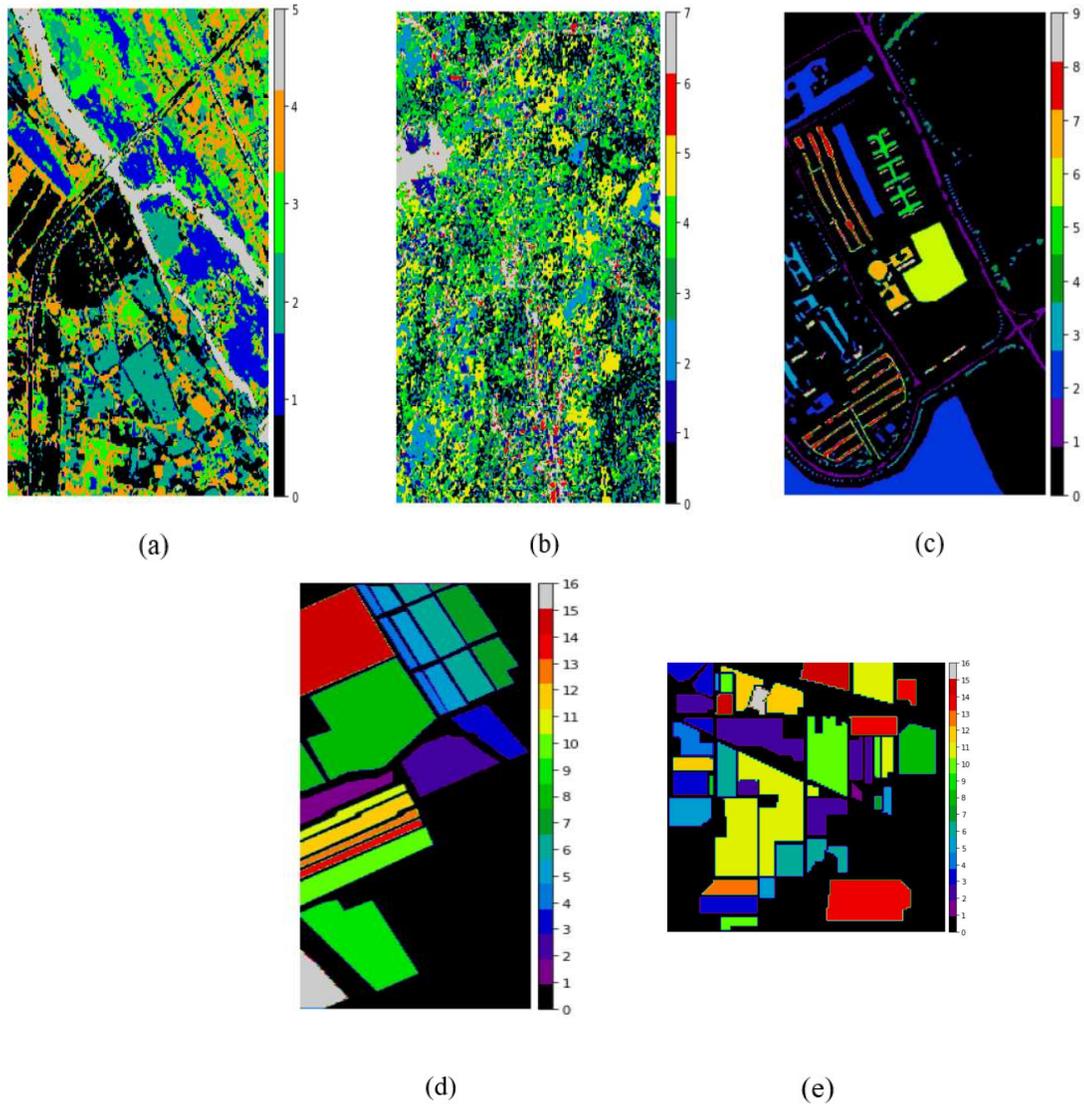


Figure 5. HSI Classification images: (a) AH-1, (b) AH-2, (c) PU, (d) SA, and (e) IP.

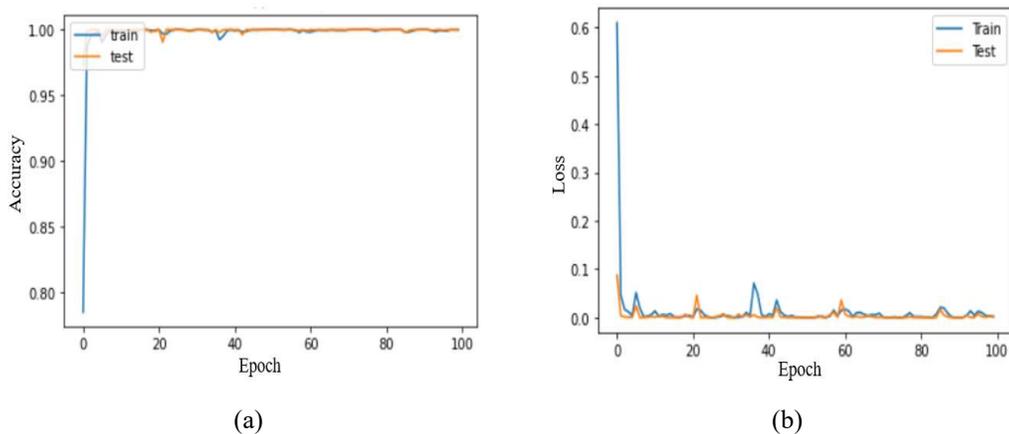


Figure 6. Presented method's testing and training on Indian Pines: (a) Accuracy, (b) Loss.

## 5. CONCLUSION

A hybrid inception CNN (Hyb-ICNN) model for HSI classification is recommended in this paper, since it has the ability to perform well even with limited training data. The Hyb-ICNN model dynamically obtains the features by laying inception components in the model which can acquire better accurate properties with smaller training samples by employing volatile spatial dimension convolutional filters and dynamic CNN framework. This model learns features quickly by incorporating a range of filter types into each layer. It isn't particularly deep; therefore, it isn't susceptible to experiencing overfit. Experiments on benchmark datasets show that the proposed model improves existing methods in terms of precision and accuracy. The efficiency of the presented model is further evaluated on new datasets, and it outperforms the HybridSN, AI-Net, and 3D-ICNN models by a significant margin. The proposed technique exhibited 2%-3% improvement in overall accuracy on benchmark and Indian datasets. The purpose of this study is to decrease the training time of each inception module in the future, resulting in a reduction in the model's overall training time.

## REFERENCES:

- [1] JM. Bioucas-Dias A. Plaza, G. Camps-Valls, P. Scheunders, NM. Nasrabadi, J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges" *IEEE Geosci Remote Sens Mag.* 2013;1(2):6–36.
- [2] A. Plaza, JA. Benediktsson, JW. Boardman, J. Brazile, L. Bruzzone, G. Camps-Valls, J. Chanussot, M. Fauvel, P. Gamba, A. Gualtieri, "Recent advances in techniques for hyperspectral image processing". *Remote Sens Environ.* 2009;113: S110–S122.
- [3] I. B. Strachan, E. Pattey, and J. B. Boisvert, "Impact of nitrogen and environmental conditions on corn as detected by hyperspectral reflectance, "Remote Sensing of environment, vol. 80, no. 2, 2002, pp. 213–224.
- [4] S. Jay and M. Guillaume, "A novel maximum likelihood-based method for mapping depth and water quality from hyperspectral remote sensing data," *Remote Sensing of Environment*, vol. 147, 2014, pp. 121–132.
- [5] C. J'anicke, A. Okujeni, S. Cooper, M. Clark, P. Hostert, and S. van derLinden, "Brightness gradient-corrected hyperspectral image mosaics for fractional vegetation cover mapping in northern california," *Remote Sensing Letters*, vol. 11, no. 1, 2020, pp. 1–10.
- [6] X. Shang and L. A. Chisholm, "Classification of australian native forest species using hyperspectral remote sensing and machine learning classification algorithms," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 6, 2013, pp. 2481–2489.
- [7] M. Kanthi1, TH. Sarma, CS. Bindu. "A Survey: Deep learning classifiers for hyperspectral image classification". *Journal of Theoretical and Applied Information Technology*, vol 99, no. 24, 2021, pp. 6042-6053.
- [8] J. Peng, W. Sun, HC. Li, W. Li, X. Meng, C. Ge, Q. Du, "Low-rank and sparse representation for hyperspectral image processing: A review". *IEEE Geosci Remote Sens Mag*, vol. 10(1), 2022, pp. 10–43.
- [9] W. Li, EW. Tramel, S. Prasad, JE. Fowler, "Nearest regularized subspace for hyperspectral classification". *IEEE Trans Geosci Remote Sens*, vol. 52(1), 2014, pp. 477–489.
- [10] Li, Xiaoxu, Z. Sun, JH. Xue, and Z. Ma. "A concise review of recent few-shot meta-learning methods.", *Neurocomputing*, vol. 456, 2021, pp. 463-468.
- [11] Liu, Bing, X. Yu, A. Yu, P. Zhang, G. Wan, R. Wang. "Deep few-shot learning for hyperspectral image classification." *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57(4), 2018, pp. 2290-2304.
- [12] Gao, Hongmin, Y. Yang, C. Li, L. Gao, B. Zhang. "Multiscale residual network with mixed depthwise convolution for hyperspectral image classification.", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59(4), 2020, pp. 3396-3408.
- [13] Jia, Sen, Z. Lin, M. Xu, Q. Huang, J. Zhou, X. Jia, Q. Li. "A lightweight convolutional neural network for hyperspectral image classification." *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 5, 2020, pp. 4150-4163.
- [14] Cui, Benlei, XM. Dong, Q. Zhan, J. Peng, W. Sun. "LiteDepthwiseNet: A lightweight network for hyperspectral image classification." *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, 2021, pp. 1-15.
- [15] Li, Wei, C. Chen, M. Zhang, H. Li, Q. Du. "Data augmentation for hyperspectral image classification with deep CNN.", *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 4, 2018, pp. 593-597.

- [16] Haut, J. Mario, M. E. Paoletti, J. Plaza, A. Plaza, J. Li. "Hyperspectral image classification using random occlusion data augmentation.", *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 11, 2019, pp. 1751-1755.
- [17] X. Li, M. Ding, A. Pizurica, "Group convolutional neural networks for hyperspectral image classification". Paper presented at: *ICIP 2019. Proceedings of the IEEE International Conference on Image Processing (ICIP)*; 2019 Sep 22–25; Taipei, Taiwan; p. 639–643.
- [18] Sarma, Vivek, A. Diba, T. Tuytelaars, and L. V. Gool, "Hyperspectral CNN for image classification & band selection, with application to face recognition", *Technical report KUL/ESAT/PSI/1604, KU Leuven, ESAT, Leuven, Belgium*, 2016.
- [19] Liu, Peng, H. Zhang, and K. B. Eom, "Active deep learning for classification of hyperspectral images.", *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 2, 2016, pp. 712-724.
- [20] Zhu, Jian, L. Fang, and P. Ghamisi, "Deformable convolutional neural networks for hyperspectral image classification.", *IEEE Geoscience and Remote Sensing Letters*, Vol. 15, No. 8, pp. 1254-1258, 2018.
- [21] Yue, Jun, W. Zhao, S. Mao, and H. Liu, "Spectral-spatial classification of hyperspectral images using deep convolutional neural networks.", *Remote Sensing Letters*, Vol. 6, No. 6, pp. 468-477, 2015.
- [22] Zhao, Wenzhi, and S. Du. "Spectral-spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach.", *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 54, No. 8, pp. 4544-4554, 2016.
- [23] Chen, Yushi, 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.
- [24] Lee, Hyungtae, and H. Kwon, "Contextual deep CNN based hyperspectral classification.", *In 2016 IEEE international geoscience and remote sensing symposium (IGARSS)*, IEEE, pp. 3322-3325, 2016.
- [25] Hamida, A. Ben, A. Benoit, P. Lambert, and C. B. Amar, "3-D deep learning approach for remote sensing image classification", *IEEE Transactions on geoscience and remote sensing*, Vol. 56, No. 8, pp. 4420-4434, 2018.
- [26] K. Murali, T. H. Sarma, and C. S. Bindu, "A 3D-deep CNN based feature extraction and hyperspectral image classification.", *In 2020 IEEE India Geoscience and Remote Sensing Symposium (InGARSS)*, IEEE, pp. 229-232, 2020.
- [27] Roy, S. Kumar, G. Krishna, S. R. Dubey, and 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.
- [28] B. Raviteja, M. S. P. Babu, K. V. Rao, and J. Harikiran, "A New Methodology of Hierarchical Image Fusion in Framework for Hyperspectral Image Segmentation," *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 6, No. 1, pp. 58-65, 2017.
- [29] S. Wan, C. Gong, P. Zhong, B. Du, L. Zhang, and J. Yang, "Multiscale Dynamic Graph Convolutional Network for Hyperspectral Image Classification.", *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 58, No. 5, pp. 3162-3177, 2020.
- [30] Z. Meng, L. Li, L. Jiao, Z. Feng, X. Tang, and M. Liang, "Fully dense multiscale fusion network for hyperspectral image classification.", *Remote Sensing*, Vol. 11, No. 22, 2019.
- [31] Mohan and M. Venkatesan, "HybridCNN based hyperspectral image classification using multiscale spatio-spectral features", *Infrared Physics & Technology*, Vol. 108, 2020.
- [32] M. Kanthi, T. H. Sarma, and C. S. Bindu, "Multi-scale 3D-convolutional neural network for hyperspectral image classification.", *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 25, No. 1, pp. 307-316, 2022.
- [33] B. Alotaibi, and M. Alotaibi, "A Hybrid Deep ResNet and Inception Model for Hyperspectral Image Classification", *PFG-Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, Vol. 88, No. 6, pp. 463-476, 2020.
- [34] 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.
- [35] 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, pp. 2033, 2020.

- [36] Kanthi, Murali, T. Hitendra Sarma, and C. Shoba Bindu. "A 3D-Inception CNN for Hyperspectral Image Classification.", *International Journal of Intelligent Engineering and Systems*, vol.15, no.1, 2022.
- [37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016.
- [38] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks", Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4700-4708, 2017.
- [39] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions", Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1-9, 2015.
- [40] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, "A review on deep learning techniques applied to semantic segmentation", arXiv preprint arXiv:1704.06857, 2017.
- [41] 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, pp. 904, 2018.
- [42] 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.
- [43] J. Ma, and J. Zhao, "Robust topological navigation via convolutional neural network feature and sharpness measure", *IEEE Access*, Vol. 5, pp. 20707-20715, 2017.