Inception SN: An Inception based Convolutional Neural Network for Hyperspectral Image Classification

Jay Kamat and Dr. Rajiv Gupta

***Abstract* - Hyperspectral Satellite Imagery provides information across multiple wavelength bands. As a result, it has extensive usage in material detection and pixel-wise classification. The advent of Convolutional Neural Networks has brought great development in this area, owing to automatic feature learning. This paper introduces a new Convolutional Neural Network, the Inception-SN, based on the famous Inception network. It makes use of 3D and 2D Inception neurons to extract both spectral as well as spatial information, thus providing state of the art accuracy on available datasets, by aggregation of several pooling and convolution filter results at each stage.**

***Index Terms* - Hyperspectral Image Classification, Convolutional Neural Networks, Inception SN, Pooling, Convolution.**

I. Introduction

With the advent of Hyperspectral Imaging satellite technology, there has been a lot of advancement in the field of Remote Sensing. Hyperspectral images provide a vast amount of information across multiple and continuous wavelength bands, thus revealing the spectral characteristics of the materials present in the scene. This property of Hyperspectral Images makes them useful for a variety of applications, such as Marine Pollution Monitoring[7], Crop Growth Monitoring[2], Ocean Color Monitoring[9] and so on. Hyperspectral Image Classification is the process of allotting a particular class to each pixel in the image, based on the spatial and spectral characteristics of the concerned pixel.

This paper was submitted for review on __________ . Our work was sponsored by ________.

Dr. Rajiv Gupta is a Senior Faculty at Department of Civil Engineering, Birla Institute of Technology and Science, Pilani, Rajasthan, India.

Jay Kamat is an Undergraduate Student at Department of Electronics and Instrumentation Engineering, Birla Institute of Technology and Science, Pilani, Rajasthan, India.

There are many methods to perform pixel-wise classification on Hyperspectral Images. Some of them include ISODATA classifier[10], Parallelopiped Classifier[11], Support Vector Machines [2], and Random Forest classifier[12]. However, these methods require great domain knowledge for obtaining good results out of them. Hyper parameter tuning of these algorithms to include all features is also a tricky affair, as there is a tendancy to miss out some features during the process of manual feature selection. Hence we require some algorithm which can learn features on it's own, so that no features are lost due to manual errors. Hence the idea of Convolutional Neural Networks, that learn features automatically by means of filters acting like weights, updating accordingly during each epoch of training. However, one disadvantage regarding the traditional convolutional neural network is that it employs only a single type of filter per convolution layer. Choosing the optimum size of such filters for each convolution layer is a difficult task. Secondly, deciding whether a particular layer should involve pooling or convoluting is another challenge, and often one may resort to hit and trial for obtaining results. One solution to this problem is through aggregation of many convolution and pooling filters at each stage, as was done to design GoogLeNet [12], by means of Inception Neurons.

In this paper, we have designed Inception SN, which has two 3D Inception layers, and one 2D Inception layer. Such an arrangement is inspired from the Hybrid SN, which relies on 3D Convolution for learning spectral features, as well as 2D Convolution for learning spatial features. We take advantage of aggregation of convolution and pooling filters of different sizes at each stage, so that the model can learn features quickly, without being too deep. Our network gives state of the art accuracy over six different Hyperspectral Datasets, as shown in Table 1. We shall use several metrics to judge the performance of our network

II. STRUCTURE OF THE INCEPTION SN

The InceptionSN is structured in a particular way to address the shortcomings of traditional deep Convolutional Neural Networks, which take a lot of time to finish training, as weight updation has to occur in a very deep network. Furthermore, deeper networks are very susceptible to overfitting. They also experience the problem of vanishing gradients, due to which the network stops learning anything once the gradient becomes too small.

Inception Modules make use of 1x1x1 convolution filters (1x1 in case of 2D convolution) to drastically reduce the number of parameters required to perform higher operations like a 3x3 convolution or 5x5 Max Pooling. This technique introduces more parameters for learning and allows the usage of multiple convolution and pooling filters in a single layer, allowing the network to use the best of them all. Hence, the depth of the network is reduced and the problem of overfitting is also avoided.

The Input layer of the Inception SN is provided by creating small 25x25 cubes of the original dataset. This is done because of the fact that there is high correlation between a given pixel and it's surrounding pixels [6]. The dimension 25x25 is chosen to give enough headroom, so that different sized filters can be used in the 2D Inception Layer. This layer is followed by two 3D Inception layers, which are responsible for automatic Spatial Feature extraction and learning. This layer is composed of several convolution and pooling filters of different sizes, whose proportion with respect to each other is chosen by following the original GoogLeNet structure.

The second 3D Inception layer has higher number of filters as compared to the first layer. This was done so that the second stage can extract more finer and detailed spectral features. After two 3D Inception layers, one 2D Inception Layer is placed, which extracts and learns spatial features. The proportion of each kind of filter involved in this layer was kept the same as that of the 3D layer. After two 3D Inception layers, one 2D Inception Layer is placed, which extracts and learns spatial features. The proportion of each kind of filter involved in this layer was kept the same as that of the 3D layer. Thus, after having performed automatic feature selection and learning, The last two fully connected layers are responsible for classification of pixels. While doing so, each pixel in the 25x25 window is allotted the same class as the central pixel. The Dropout Regularization technique is used to prevent overfitting of the model. In order to make the algorithm computationally efficient, as well as to use images with maximum information content, Principal Component Analysis is performed on each dataset before it is broken down into cubes and fed to the Input Layer.

III. DATASETS USED

We have used six different datasets to judge the performance of our classifier. For each dataset, the number of bands to be selected after applying PCA was decided based upon the hardware limitations of RAM and GPU available on Google Colab.

Each of these datasets used were atmospherically corrected datasets. As evident from Table 1., most of the datasets are captured either by AVIRIS or EO-1 Hyperion satellite. For the Indian Pines Dataset, water absorption bands (104-108), (150-163) and 220 were removed before analysis. Similarly, water absorption as well as low SNR bands were removed for Salinas scene, Kennedy Space Centre, and Botswana dataset, leaving 204, 176 and 145 bands respectively for analysis. Some portions of Pavia University and Pavia Centre dataset were removed as they contained lack of information.

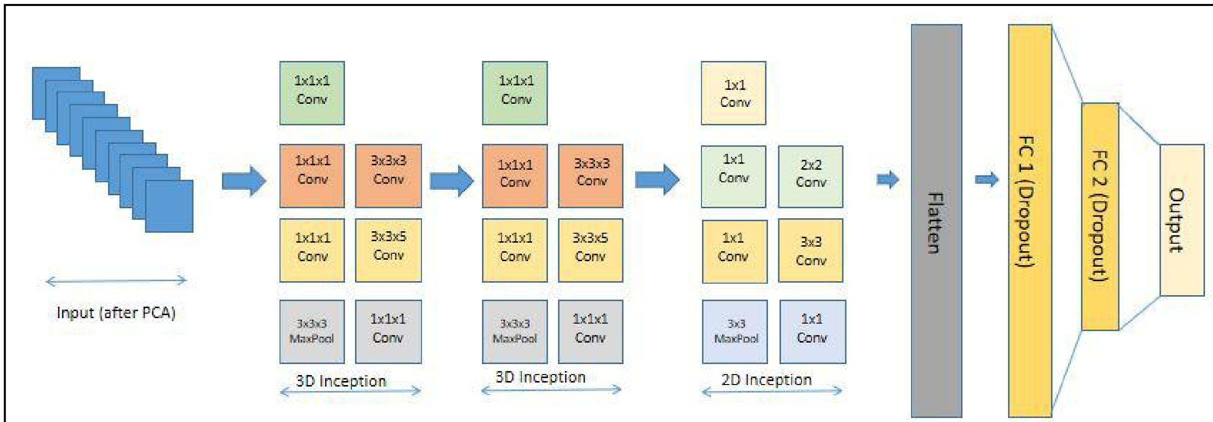


Figure I. Structure of the Inception SN

Table I. Information regarding datasets used

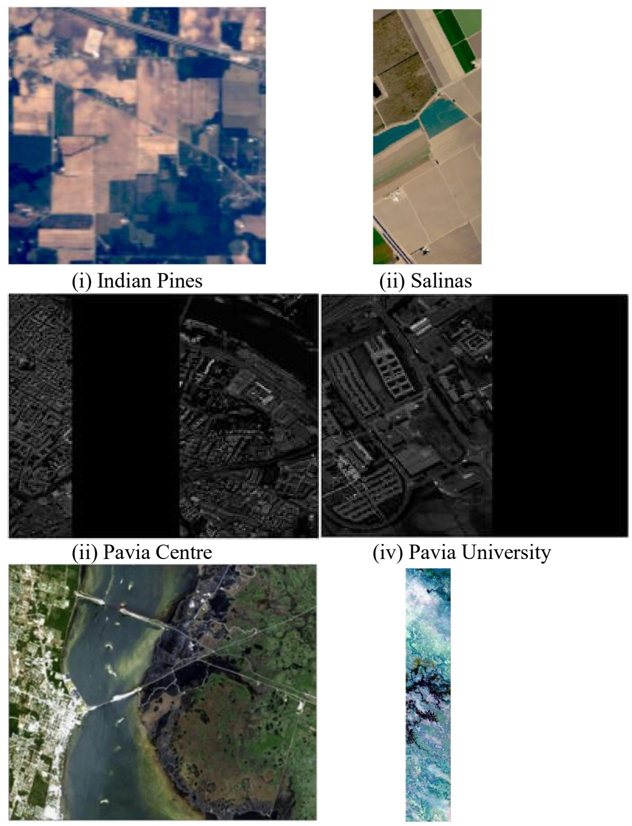
Sr. No.	Name of Dataset	Location	Sensor	Bands	Classes
1	Pavia University	Pavia, Northern Italy	ROSIS	103	9
2	Pavia Centre	Pavia, Northern Italy	ROSIS	102	9
3	Indian Pines	North-West Indiana	AVIRIS	224	16
4	Salinas	Salinas Valley, California	AVIRIS	224	16
5	Kennedy Space Centre	Florida	AVIRIS	224	13
6	Botswana	Okavango Delta, Botswana	Hyperion	242	14

IV. RESULTS

The Inception SN was trained by random division of the dataset into training and testing data in the ratio 3/7. Along with several accuracy metrics, Kappa coefficient was also chosen as a performance metric. This was to verify the credibility of the data used, and to show how much accuracy was caused due to the information contained in the dataset, instead of that caused by random selection. Table 2. shows several performance metrics corresponding to each dataset, while Table 3. compares the Inception SN's performance with other networks on the same data. In order to have a fair comparison, the same train:test ratio was used to train the Inception SN as that of the other networks.

Our experiments also showed that the InceptionSN performs well on the test set even when provided with little training data . Table 4 compare the performance of Inception SN with other networks for a train : test ratio of 1/9. The models with which our Inception SN's performance was compared are the Support Vector Machines [2], 2D CNN [3], 3D CNN [4], the Spatial-Spectral Residual Network [5], the Multi-Scale 3D CNN [6], and the Hybrid SN [1]. The performance results of all these networks are taken from [1], and the Inception SN has been trained in the same scheme as of [1], to compare results.

Fig II. Scene Images of Data



(i) Indian Pines (ii) Salinas

(ii) Pavia Centre (iv) Pavia University

(v) Kennedy Space Centre (vi) Botswana

Table II. Result of Inception SN (test set result)

Sr. No.	Name of Dataset	Overall Accuracy %	Average Accuracy %	Kappa Coefficient	Test Accuracy %
1	Pavia University	100	100	100	100
2	Pavia Centre	99.9807	99.9369	99.9727	99.9807
3	Indian Pines	99.5819	99.7757	99.5234	99.5818
4	Salinas	99.9921	99.9836	99.9912	99.9921
5	Kennedy Space Centre	99.6985	99.5382	99.6643	99.6985
6	Botswana	99.8242	99.8482	99.8094	99.8241

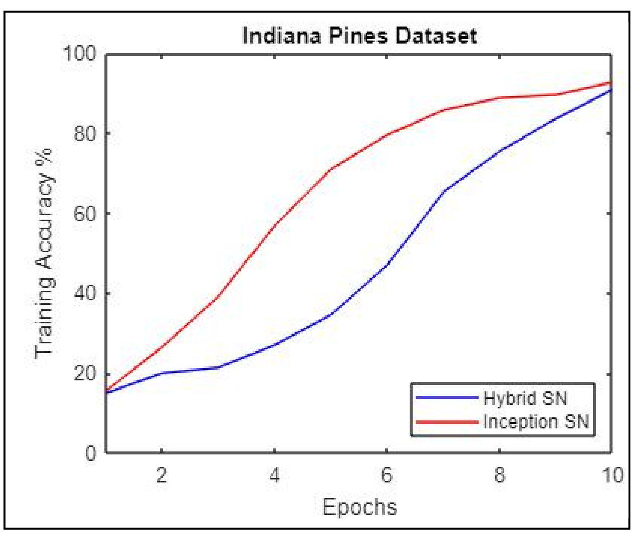
Table III. Comparison with other datasets for 30% training data (test set result)

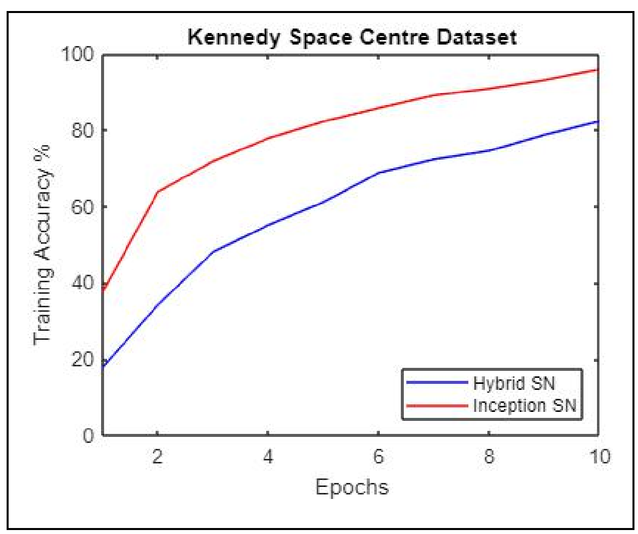
Sr. No.	Network	Indian Pines			Salinas			Pavia University		
		OA	AA	K	OA	AA	K	OA	AA	K
1 [1]	SVM	85.30	79.03	83.10	92.95	94.60	92.11	94.34	92.98	92.50
2 [1]	2D-CNN	89.48	86.14	87.96	97.38	98.84	97.08	97.38	96.55	97.16
3 [1]	3D-CNN	91.10	91.58	89.98	93.96	97.01	93.32	93.96	97.57	95.51
4 [1]	M3D-CNN	95.32	96.41	94.70	94.79	96.25	94.20	94.79	95.08	94.50
5 [1]	SSRN	99.19	98.93	99.07	99.98	99.97	99.97	99.9	99.91	99.87
6 [1]	Hybrid SN	99.75	99.63	99.71	100	100	100	99.98	99.97	99.98
7	**Inception SN**	**99.58**	**99.78**	**99.52**	**99.99**	**99.98**	**99.99**	**100**	**100**	**100**

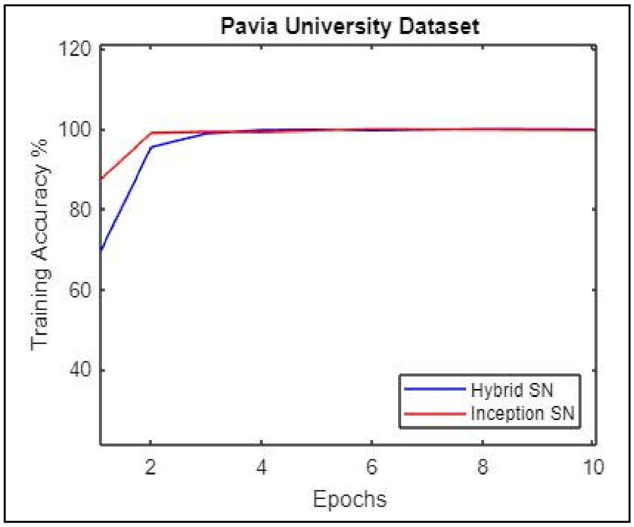
Table IV. Comparison with other datasets for 10% training data (test set result)

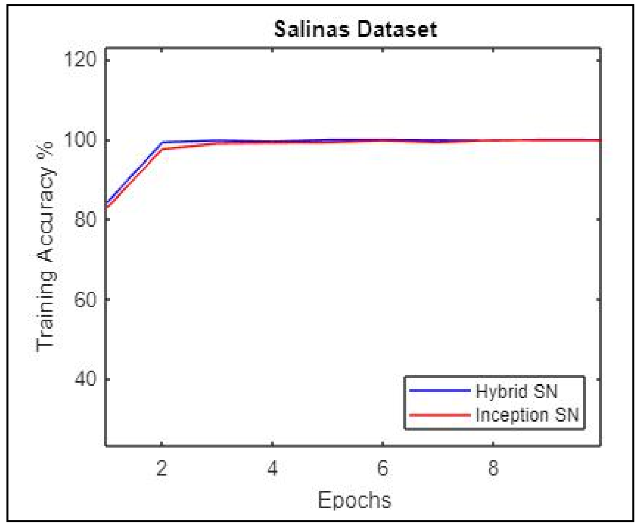
Sr. No.	Network	Indian Pines			Salinas			Pavia University		
		OA	AA	K	OA	AA	K	OA	AA	K
1 [1]	2D-CNN	80.27	68.32	78.26	96.34	94.36	95.93	96.63	94.84	95.53
2 [1]	3D-CNN	82.62	76.51	79.25	85	89.63	94.90	96.34	97.03	94.90
3 [1]	M3D-CNN	81.39	75.22	81.20	94.20	96.66	93.40	95.95	97.52	93.40
4 [1]	SSRN	98.45	86.19	98.23	99.64	99.76	99.50	99.62	99.49	99.50
5 [1]	Hybrid SN	98.39	98.01	98.16	99.98	99.98	99.98	99.72	99.20	99.64
6	**Inception SN**	**97.77**	**96.63**	**97.45**	**99.85**	**99.80**	**99.83**	**99.77**	**99.36**	**99.69**

Figure III. Training accuracy comparison for 30% training data









From Figure III, it can be concluded that the Inception SN is faster than Hybrid SN in learning different features of the Hyperspectral Datasets available, and maintaining close, and sometimes even better performance. While there is drastic improvement over the Kennedy Space Centre and Indian Pines dataset, the performance is marginally better on Pavia University, and almost similar on Salinas dataset. One reason for the following could be the complexity of features involved. While the Hybrid SN has roughly 5 million trainable parameters, the Inception SN has roughly 39 million trainable parameters. This helps the Inception SN perform better while learning complicated features, and yields similar performance on simpler features

V. Conclusion

We have seen in this paper that the Inception SN can perform very well, even with reduced training data. It takes advantage of multiple filter types in each layer, and learns features effectively. It is not too deep, hence it is not prone to overfitting. It is also quicker to learn features. One limitation of the field of Hyperspectral Image classification is the lack of publicly available labelled hyperspectral data. We were able to obtain only six such datasets (which were accompanied by ground truth). We hope to obtain more data in the future so that the Inception SN can be tested more thoroughly.

VI. References

[1] 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 Geosci. Remote Sens. Lett., vol. 7, no. 2, pp 277, Feb. 2020.

[2] F. Melgani and L. Bruzzone, “Classification of hyperspectral remote sensing images with support vector machines”, IEEE Geosci. Remote Sens. Lett., vol. 42, pp. 1778-1790, Aug. 2004

[3] K. Makantasis, K. Karantzalos, A. Doulamis, and N. Doulamis, “Deep supervised learning for hyperspectral data classification through convolutional neural networks”, Proceedings IEEE Int. Geosci. Remote Sens. Symposium, July 2015, pp. 4959-4962.

[4] A.B Hamida, C.B. Amar, P. Lambert and A. Benoit, “3D Deep learning approach for remote sensing image classification”, IEEE Geosci. Remote Sens. Lett., vol. 56, no. 8, pp. 4420-4434, Aug. 2018.

[5] H. Chen, B. Li, and M. He, “Multi-scale 3D deep convolutional neural network for hyperspectral image classification”, in Proceedings IEEE Intl. Conference on Image Processing (ICIP), Sept. 2017, pp. 3904-3908.

[6] Leng J, Li T, Bai G, “Cube-CNN-SVM: A Novel Hyperspectral Image Classification Method”, IEEE, International Conference on TOOLS with Artificial Intelligence. IEEE, 2017:1027-1034

[7] S. Hafeez, M.S. Wong, S. Abbas, C,Y,T Kwok, J. Nichol, K.H. Lee, D. Tang, and L. Pun, “Detection and monitoring of marine pollution using remote sensing technologies”, Monitoring of Marine Pollution, InTech Open.

[8] A. Kumar, V. Bharti, V. Kumar, U. Kumar and P.D. Meena, “Hyperspectral imaging: A potential tool for monitoring crop infestation, crop yield and macronutrient analysis, with special emphasis to Oilseed Brassica”, Journal of Oilseed Brassica, 7 (2) : 113-125, July 2016.

[9] J.K Joliff, S. Ladner, D. Lewis, E. Jarosz, A. Lawson, T. Smith, A. Penko, S. McCarthy, “Hyperspectral determination of ocean color as an ocean monitoring tool: example applications in the Gulf of Mexico”, Proceedings to SPIE Defense+Security, 2018. Ocean Sensing and Monitoring, vol. 10631.

[10] Q. Wang, Q. Li, H. Liu, Y. Wang, J. Zhu, "An improved ISODATA algorithm for hyperspectral image classification", IEEE Intl. Conference on Signal and Image Processing, Oct. 2014.

[11] Sahar A., El Rahman, "Performance of Spectral Angle Mapper and Parallelepiped Classifiers in Agriculture Hyperspectral Image", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 5, 2016.

[12] S.R. Joelsson, J.A Benediktsson, J.R Sveinsson, "Random forest classifiers for hyperspectral data", Geoscience and Remote Sensing Symposium, 2005. IGARSS '05. Proceedings. 2005 IEEE International, Volume: 1.

[13] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, " Going deeper with convolutions", CVPR 2015.