Evaluating the Performance of the state-of-the-art HybridSN Deep Learning Algorithm for Airborne Hyperspectral Image Classification

M A A M Abidin¹, H Z M Shafri¹*, M M A Al-Habshi¹, N S N Shaharum²

¹Department of Civil Engineering and Geospatial Information Science Research Centre (GISRC), Faculty of Engineering, Universiti Putra Malaysia (UPM) 43400 Serdang, Malaysia
²School of Engineering, Monash University Malaysia, Jalan Lagoon Selatan, 47500, Bandar Sunway, Selangor Darul Ehsan, Malaysia

*helmi@upm.edu.my

Abstract. This study aims to evaluate the performance of state-of-the-art HybridSN deep learning algorithm versus standard machine learning (ML) and deep learning (DL) techniques using open-source Python libraries for producing hyperspectral land use and land cover (LULC) classification maps. Japanese Chikusei hyperspectral datasets captured by the airborne platform using Hyperspec-VNIR-C sensor were used in this study. Standard ML methods used in this study were support vector machine linear kernel (SVM-linear), support vector machine radial basis function kernel (SVM-RBF) and random forests (RFs) that were provided in Python’s Scikit-learn library. DL techniques used in this study were multilayer perceptron (MLP), two-dimensional convolutional neural network (2-D CNN) and hybrid spectral convolutional neural network (HybridSN), which integrates the 2-D and 3-D feature learning. These DL models were built based on the sequential model using Keras API. The results show that all the proposed methods obtained overall accuracies (OAs) above 95%. The HybridSN and 2-D CNN models gave the best score with 99.97% OAs for hyperspectral image classification using the Chikusei dataset.

1. Introduction

Land Use and Land Cover (LULC) information is an important data source for modeling environmental variables. Therefore, it is essential to develop high-quality LULC maps. Hyperspectral sensors provide high spectral detail with hundreds of continuous spectral bands – an appropriate option for LULC applications. Despite increased spectral detail, issues like the curse of dimensionality, huge volume of data and redundant information show that hyperspectral image (HSI) classification is a complex task. Therefore, it is essential to develop classification approaches that effectively deal with these issues.

Using open-source-based tools for HSI processing is a great option – it is 100% free to use, allows you to view, study, modify the code and make local specific adaptation (improve the overall efficiency of the underlying algorithms etc.). Python is currently among the top programming languages for data science and ML. It’s simple, easy to learn and user friendly. One huge advantage of python is the vast collection of libraries for performing a variety of analytical tasks.
In recent decades, rapid technological progression in remote sensing platform along with the optical scanners sees an increasing capacity of information due to improved spatial, spectral, and temporal resolutions. Notably, the better spectral information comprises of hyperspectral images allows the development of new application domains which in simultaneously, creates new computing technology challenges in data processing and analyses.

The HSI processing includes the application of algorithms and classification methods ranging from a basic mathematical formula or statistical approach to more advanced systems by adopting machine learning (ML) techniques [1]. In ML context, for many years, the remote sensing community are used to the well-established traditional methods such as support vector machine (SVM) and ensemble classifiers, e.g., random forest for HSI image classification and other analyses. It is due to the simple usage of algorithms and fast computing [2]. As for SVM, the ability to handle big dimensionality data with limited training samples and score high accuracy makes it very popular among other classifiers [3].

One of the prominent approaches studied by researchers lately is deep learning (DL) frameworks [4]. DL becomes a significant and accessible method for HSI classification because of its proven capabilities in other subjects such as natural language processing [5], speech recognition [6] and computer vision [7].

Multilayer Perceptron (MLP) is a conventional method of deep learning. The structure is characterized by three or more layers of nodes known as an input layer, a hidden layer, and an output layer. It uses a backpropagation technique for training the data. Roooposhti et al. (2019) investigated the uncertainty assessment to estimate the accuracy of the classification methods of MLP and Random Forests (RFs). The result shows that the MLP algorithm has a better estimate of classification accuracy compared to RFs.

Convolutional neural network (CNN) is a subclass of deep neural network. A basic structure of a supervised CNNs classifier is characterized by a pair of convolution and max-pooling layers followed by a fully connected layer and an output layer. It uses a backpropagation technique for training the data. Hu et al. (2015) proposed a basic one-dimensional (1xN) CNN consist of five layers (input, conv., pooling layer, fully conn. and output layers). The result shows that the proposed method outperformed the SVMs and conventional DL methods.

Makantasis et al. (2015) proposed a two-dimensional (2-D) that combines spectral and spatial features into the CNNs model. They used randomized-PCA (R-PCA) to reduce the spectral dimensions. The model then constructs high-level spectral-spatial information simultaneously from the HSI input data followed by a classification operation using MLP. The CNN result gives better performance compared to the SVMs. Roy et al. (2020) proposed a hybrid spectral convolutional neural network (HybridSN), a 3-D – 2-D CNN feature hierarchy model for HSI classification. The work used a spectral-spatial 3-D CNN followed by spatial 2-D CNN convolutional layers. The result shows that HybridSN outperforms all the compared methods (SVMs, 2-D CNN, 3-D CNN) for Indian Pines, Pavia University and Salinas Scene remote sensing datasets. However, to further test the performance of the HyBridSN DL model, further experimentation is required.

Therefore in this study, by using open-source Python-based tools, the proposed standard ML techniques (SVM-linear, SVM-RBF and RFs) will be compared with DL frameworks namely multilayer perceptron (MLP) two-dimensional convolutional neural network (2-D CNN) and Hybrid Spectral Convolutional Neural Network (HybridSN) for HSI classification of Chikusei airborne hyperspectral dataset. The result is compared based on the accuracy assessment of the classifiers.

2. Materials and methods

2.1. Datasets
The Chikusei hyperspectral datasets were captured by airborne using Hyper-spc-VNIR-C (Headwall Photonics Inc.) sensor covering the agricultural and urban settlements over the area. The images were taken on July 29, 2014, around 9:56 to 10:53 UTC+9 times. These datasets were made freely available from Yokoya and Iwasaki (2016) [9] for scientific research purposes. The original size of the full image
is 2517 × 2335 pixels. The size of the scene was then reduced to 540 × 420 pixels due to limitations of computational capacity. The datasets were readily made and pre-processed with radiometric, geometric, atmospheric, and BRDF correction. The datasets are also enclosed with ground truth data containing 19 land cover classes with a total of 77592 pixels (Figure 1).

![Image](image_url)

**Figure 1.** (a) Chikusei HSI dataset, (b) Ground Truth dataset (gt).

2.2. Support Vector Machines (SVMs) and Random Forests (RFs) classifier

Scikit-learn is a Python package that was developed by David Cournapeau in 2007. It consists of a wide range of useful state-of-the-art ML algorithms that can easily be applied and tuned for medium-scaled supervised and unsupervised classification and other machine learning problems [10]. Scikit-learn library provides easy access to some of the common standard and ML classifiers. Among the classifiers, SVMs and RFs are the methods that were used in this study. These methods were chosen because of the top performance from the previous studies besides the ease of use [11].

SVMs is a set of the supervised non-parametric statistical learning algorithm, which means no assumption is being made on the distribution of input data. SVMs work by using a set of training data to draw an optimal decision line or surface called "hyperplane" that group clusters of data points into a discrete number of predefined classes in line with the labelled samples [3]. The hyperplane tries to maximize the margin between classes to increase the confidence in assigning the points into classes during the training task. The two SVMs kernel methods that were used in this study are linear (SVM-linear) and radial basis function (SVM-RBF). These methods were chosen due to the excellent interpretation of feature generalization as well as high accuracy achievement from track records of HSI classification results [12]. The SVM-linear has one parameter, which is the penalty, C while the SVM-RBF classifier has two parameters: gamma and C.

Gamma is a parameter of the RBF kernel that can be understood as the 'spread' of the kernel region is. As the gamma value increases, the 'curve' of the decision boundary is high. It means that as the value of gamma increases, the model gets overfits and vice versa. The C parameter is the penalty for misclassifying a data point. When C value is higher, the classifier is penalized for the misclassified data and bends over backwards. This is happening to avoid any misclassified data points (low bias, high variance). It means that as the value of C increases, the model gets overfits and vice versa. The input value for C and gamma parameters is crucial to the SVMs’ performance. Therefore, the experiment is usually conducted to determine the values using cross-validation methods [13]. In the Scikit-learn library, the "GridSearchCV" module was used to get the optimized value of the SVMs hyperparameters.

RFs, also known as random decision forest, is a popular method of ensemble learning-type classifiers. The algorithm works by creating a number of decision trees at a training period and generates the classification outputs with most votes over all the trees in the forest for a given input vector [14]. In RFs classification, a split is determined at each node based on the search across a random subset of variables. RFs require two parameters for the model training: the number of trees and the number of predictor
variables. The number of trees parameter value can be chosen by concerning the size and complexity of the training set. A possible method on how to select the optimal parameter is by analyzing the out-of-bag (OOB) error [14]. In this study, the number of trees parameter is set to 500.

2.3. Multi-layer Perceptron (MLP), 2-D convolutional neural network (2-D CNN) and Hybrid spectral convolutional neural network (HybridSN)

MLP is a supervised learning algorithm that is part of the feedforward artificial neural network (ANN). An MLP structure is characterized by three or more layers of nodes known as an input layer, a hidden layer, and an output layer.

A convolutional neural network (CNN) structure is originated from the biological visual cortex system and neuroscience. CNNs are basically regularized versions of MLP while MLP usually refers to fully connected networks between neurons of the different layers. A basic structure of a supervised CNNs classifier is characterized by a pair of convolution and max-pooling layers followed by a fully connected layer and an output layer [8].

Hybrid spectral convolutional neural network (HybridSN) is first proposed by Roy et al. (2020). It is a 3-D – 2-D CNN feature hierarchy model for HSI classification. The work utilized a spectral-spatial 3-D CNN followed by spatial 2-D CNN convolutional layers. The 3-D CNN enables the integration of spatial-spectral feature representation from the stacked spectral bands. The 2-D CNN layer which is on top of the 3-D CNN layer can learn a more generalized level of the spatial features. The architectures are shown in Figure 2.

![Figure 2](image_url)

**Figure 2.** The architectures of the MLP (a), 2-D CNN (b) and HybridSN (c) models.

3. Result and discussion

3.1. SVMs and RFs classifiers with Scikit-learn

To achieve high performance, different strategies and hyperparameters are used and tested for every model. In the SVM-linear method, the grid-search method was used for selecting the best C parameter.
SVM-RBF also applied the grid-search method with parameter the C and gamma parameter. However, the grid-search method does not cover all the value ranges where only the input ranges are cross-validated for the best parameter search. Therefore, the other experimental parameter values are still important to test the SVMs’s classification performance. RFs gives a high and competitive performance with other top models. However, the classification map seems to be overfitting even the number of trees hyperparameter was increased from 500 to 1000. The SVMs and RFs methods utilized all the bands without having computational issues. In Scikit-learn classifier module, the score of the trained model can be measured using the ‘score’ function. The function calculated the mean accuracy on the provided test data and ground truths.

3.2. MLP, 2-D CNN and HybridSN DL methods
For DL methods in this study, the datasets are rescaled or normalized first before fitting into a model. Having features on a standard scale can help the gradient descent converge more quickly towards the minima, thus improves the performance of the algorithm significantly. The technique used for rescaling is ‘standardization’ method which transforms the data to have zero mean and a standard deviation of 1.

The datasets are divided into patches for the input and target tensors so that compatible with neural network models that are used for image classification. The patch size is depending on the complexity of a neural network model. The DL models in this study are build based on the sequential model type provided by Keras API. It is organized using a layer-by-layer designation.

For the MLP model, the strategy is to learn by activation function from the input training data network using a back-propagation technique. Backpropagation is an algorithm that calculates the gradient of the loss function from top to bottom layer (backward) in relation to the network weights. Excluding the input nodes, each node represents by a neuron that applying a non-linear activation function. All the neurons of the layers are connected for the feature learning and creating the output.

The standard ML classifiers and MLP model only utilize the spectral information but not the rich spatial information. Therefore, the proposed 2-D CNN model exploits the spatial information to achieve a better performance in HSI classification. In 2-D CNN, the PCA is applied to reduce the bands. If the PCA is not applied, the redundant noise will affect the quality and the accuracy of the classification output. The strategy of 2-D CNN to utilize the spatial information is by applying two scales of filters: 32 and 64 with the same kernel size (3 x 3). The scales can be added further up to make the features more abstract or generalized. However, the more layers added will increase the computational complexity thus more powerful hardware is required.

A 2-D CNN only exploits the spatial context of the 2-D convolutional layer. However, the HybridSN method integrates the spectral-spatial 3D-CNN and spatial 2D-CNN. The strategy is to exploit spatial-spectral feature representation from stacked bands using 3D-CNN layer and then the 2-D CNN layer is put on top of the layer so that it can learn more abstract level of spatial representation. The downside of this method is that it increases the computational complexity that causing a longer training time.

3.3. Result of classification maps
The classification maps are derived from the trained model’s prediction for the entire dataset including the training and test datasets, and the unlabeled pixels. In this study, all the standard ML methods using classifiers from Scikit-learn are capable to perform both the training and prediction of the whole image without hardware issues.

However, when it comes to the DL models, the hardware could not handle the huge volume of full-size Chikusei dataset for the training and image prediction process. Therefore, the subset image is used for the comparison of ML and DL methods for HSI classification. Figure 3 (a – f) show the results of the classification map for SVM-linear, SVM-RBF, RFs, MLP, 2-D CNN and original dataset in true color.

From the classification results, it seems that SVM-RBF provides the best visual interpretation of the feature classes. The generalization of each class is more consistent, and smoother compared to other models’ LULC map output. RF’s output also looks smooth as the SVMs, but it seems to contain mix
features within some classes. Same case with the SVM-linear where some classes seem to be mixed-up. This is due to the linear kernel used which is hard to predict the non-linear features of the dataset. MLP output looks slightly better than RFs in term of generalization but the quality of the visual image is slightly reduced. 2-D CNN result gives a good interpretation accuracy, but the feature classes are emphasized and look slightly overfit. While for the HybridSN, the model is unable to predict the whole image due to hardware limitation (insufficient memory). Thus, the model is used to predict the ground truth dataset as a partial map. The overall accuracy of the visual interpretation of the ground truth is very high, and the prediction map output is almost identical to the ground truth dataset (Figure 4).

**Figure 3.** The classification map for Chikusei dataset: (a) SVM-linear, (b) SVM-RBF, (c) RFs, (d) MLP, (e) 2-D CNN, (f) True color image
3.4. Classification accuracy assessment

This section provides the evaluation of classification accuracy for all the proposed algorithms. It gives an overview of how well the performance of a model in predicting the in-dependent samples or test samples in a classification task. The method that is used in this study is K-fold cross-validation (K-fold CV) method. K-Fold CV is where a given data set is divided into an equally sized K number of folds where each fold is used as a training and testing set at some point. With K-fold CV, all samples are utilized for both training and testing. In this study, the number of fold k=10 is used as conventionally recommended in general ML applications [15].

The accuracy results of the proposed classifiers are summarized in Table I. It is shown that all the proposed methods scored overall accuracy (OA) over 95%. From that, both HybridSN and 2-D CNN give the best performance result for HSI classification in this study with 99.97% OA followed by SVM-linear (99.83%), SVM-RBF (99.80%), RFs (99.62%), and MLP (99.03%). The DL methods are found to be more accurate than standard ML. However, the accuracy scores seem not to reflect the true visual interpretation for the DL models where SVM-RBF shows a better generalization of the classification map.

| Methods               | Standard ML           | DL               |
|-----------------------|-----------------------|------------------|
|                       | SVM-linear | SVM-RBF | RFs   | MLP       | 2-D CNN | Hybrid SN |
| OA (%)                | 99.83      | 99.80   | 99.62 | 99.03     | 99.97   | 99.97     |

4. Conclusion

In conclusion, this study demonstrates the use of standard ML and DL techniques for HSI classification of LULC map with freely available Python tools or modules. The standard ML classifiers used in this study are taken from the Scikit-learn open-source library. The classifiers used are SVM-linear, SVM-RBF and RFs. For DL models, the Keras API is used to build and tune the sequential model type for DL framework. The DL models proposed in this study are MLP, 2-D CNN, and HybridSN.

The classification result from the study shows that both HybridSN and 2-D CNN have the best overall accuracy with 99.97% OA followed by SVM-linear (99.83%), SVM-RBF (99.80%), RFs (99.62), and MLP (99.03%). Although HybridSN and 2-D CNN have the highest score, the accuracy does not reflect the truth of visual interpretation on the image where the SVM-RBF shows better generalization and smoothness of the feature classes.

In general, based on the results, it seems that the DL methods outperformed the standard ML methods in this study in term of OA. The advantage of a DL technique compared to standard ML method is that the number of layers can be added to improve the model and the arrangement of the structure is very flexible with some parameters can be added too. However, the downside of the DL method is the more layers added, the more computational resources are needed to handle the complex processes.
To achieve high performance, different strategies and methods are tested including reducing the redundant spectral dimensions. Although using PCA can reduce dimensionality and noisy bands, it is not always suitable for any model. For example, in this study, applying PCA in MLP framework will decrease the accuracy and the quality of the classification map.

It is noted that all the proposed methods’ performance in this study is very close and very competitive with each other. Therefore, any of the above methods are suitable to be used for producing a decent and accurate LULC map. However, it is well-known that the HSI classification performance for ML and DL methods are highly dependent on the proper hyperparameter value. Thus, proper methods for tuning the hyperparameters is essential. Users also need to have appropriate knowledge of the underlying algorithms to build an effective and efficient model. Future work could also investigate more recently developed algorithms such as the SSNET [16] and SyCNN [17] and applying state of the art algorithms to wider and larger coverage hyperspectral datasets.

References
[1] H. Z. M. Shafri, “Machine Learning in Hyperspectral and Multispectral Remote Sensing Data Analysis,” pp. 3–9, 2017, doi: 10.1142/9789813206823_0001.
[2] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. A. Johnson, “Deep learning in remote sensing applications: A meta-analysis and review,” ISPRS J. Photogramm. Remote Sens., vol. 152, no. April, pp. 166–177, 2019, doi: 10.1016/j.isprsjprs.2019.04.015.
[3] G. Mountrakis, J. Im, and C. Ogole, “Support vector machines in remote sensing: A review,” ISPRS J. Photogramm. Remote Sens., vol. 66, no. 3, pp. 247–259, 2011, doi: 10.1016/j.isprsjprs.2010.11.001.
[4] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, “Deep learning classifiers for hyperspectral imaging: A review,” ISPRS J. Photogramm. Remote Sens., vol. 158, no. November 2018, pp. 279–317, 2019, doi: 10.1016/j.isprsjprs.2019.09.006.
[5] A. Torfi, R. A. Shirvani, Y. Keneshloo, N. Tavaf, and E. A. Fox, “Natural Language Processing Advancements By Deep Learning: A Survey,” pp. 1–21, 2020, [Online]. Available: http://arxiv.org/abs/2003.01200.
[6] A. B. Nassif, I. Shahin, I. Attili, M. Azzez, and K. Shaalan, “Speech Recognition Using Deep Neural Networks: A Systematic Review,” IEEE Access, vol. 7, no. February, pp. 19143–19165, 2019, doi: 10.1109/ACCESS.2019.2896880.
[7] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.
[8] X. X. Zhu et al., “Deep learning in remote sensing: a review,” no. december, 2017, doi: 10.1109/MGRS.2017.2762307.
[9] N. Yokoya and A. Iwasaki, "Airborne hyperspectral data over Chikusei," Space Appl. Lab., Univ. Tokyo, Japan, Tech. Rep. SAL-2016-05-27, May 2016.
[10] G. Varoquaux, L. Buitinck, G. Louppe, O. Grisel, F. Pedregosa, and A. Mueller, “Scikit-learn,” GetMobile Mob. Comput. Commun., vol. 19, no. 1, pp. 29–33, 2015, doi: 10.1145/2786984.2786995.
[11] Y. Chen, J. U. N. Li, and A. Plaza, “Advanced Spectral Classifiers for Hyperspectral Images,” no. March, 2017.
[12] H. Hasan, H. Z. M. Shafri, and M. Habshi, “A Comparison Between Support Vector Machine (SVM) and Convolutional Neural Network (CNN) Models For Hyperspectral Image Classification”, IOP Conference Series: Earth and Environmental Science 2019, doi: 10.1088/1755-1315/357/1/012035.
[13] V. Kecman, “Support Vector Machines – An Introduction Support Vector Machines – An Introduction,” no. May, 2014, doi: 10.1007/10984697.
[14] L. Breiman, “Random forests,” Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.
[15] P. Refaeilzadeh, L. Tang, and H. Liu, “Cross-Validation BT - Encyclopedia of Database Systems,” L. LIU and M. T. ÖZSU, Eds. Boston, MA: Springer US, 2009, pp. 532–538.

[16] A. Paul, S. Bhoumik, and N. Chaki, “SSNET: an improved deep hybrid network for hyperspectral image classification,” Neural Comput. Appl., 2020, doi: 10.1007/s00521-020-05069-1.

[17] X. Yang et al., “Synergistic 2D/3D convolutional neural network for hyperspectral image classification,” Remote Sens., vol. 12, no. 12, pp. 1–19, 2020, doi: 10.3390/rs12122033.

Acknowledgment
The authors gratefully acknowledge Space Application Laboratory, Department of Advanced Interdisciplinary Studies, the University of Tokyo for providing the hyperspectral data. Comments from anonymous reviewers towards improving this paper are also acknowledged.