Classification Analysis on Open Hyperspectral Datasets

Abhijeet G

Dhirubhai Institute of Information and Communication Technology

October 3, 2018

Overview

1. Overview
2. Results and Discussion

Introduction

Hyperspectral data has both spatial and spectral components making a datacube which is very large to process. We have to adopt dimensionality reduction techniques and feature mapping techniques in order to train a model that can get accurate crop classification. A Hyperspectral sensor collects information which is a set of narrow wavelength range of the electromagnetic spectrum called as a spectral band. Since it is in the hyperspectral range, there will be many bands collected by the sensors . Some bands can be eliminated by doing atmospheric correction or selective band resampling. The Wavelength range of the hyperspectral sensor is 0.4 to 2.5 micrometers. These images are combined to form a three-dimensional (x, y,) hyperspectral data cube for processing and analysis.

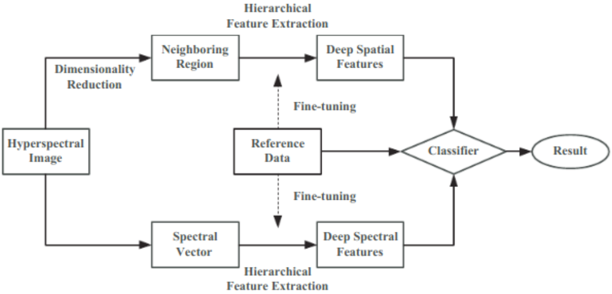


Fig. 1. General framework of HSI classification based on deep learning methods.

Credit : HSI journal paper

Datasets

Indian Pines Dataset

The following are the particulars of the dataset: Source: AVIRIS sensor
Region: Indian Pines test site over north-western Indiana
Number of spectral bands: 220
Size of image: 145x145 samples
Number of land-cover classes: 16

Pavia Dataset

Source: ROSIS sensor during a flight campaign over Pavia, nothern Italy.
Geometric Resolution: 1.3 meters
Number of spectral bands : 103
Size of Image 610x610 samples
Number of land-cover classes: 9

Datasets

Salinas dataset

Spatial Resolution of 3.7 meters
Size of Image 512x217 samples
Number of spectral bands : 224
Number of land-cover classes: 16 , consisting of vegetables, bare soils, and vineyard fields.

ISRO Dataset

Complete AVIRIS-NG Phase 1 over Agricultural University, Anand District , Gujarat (One Airborne flight scene)

Indian Pines Results

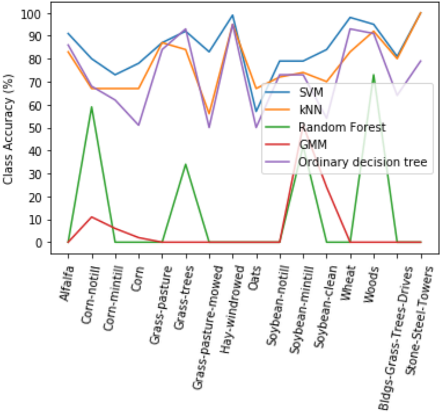


Figure: Classwise accuracy for different methods (Indian Pines)

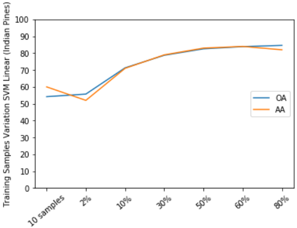


Figure: OA and AA variation as training size changes

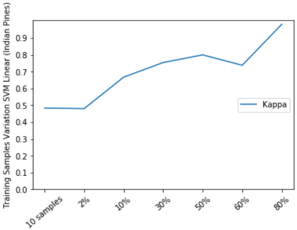


Figure: Kappa coefficient value variation as training size changes

Salinas Dataset Results

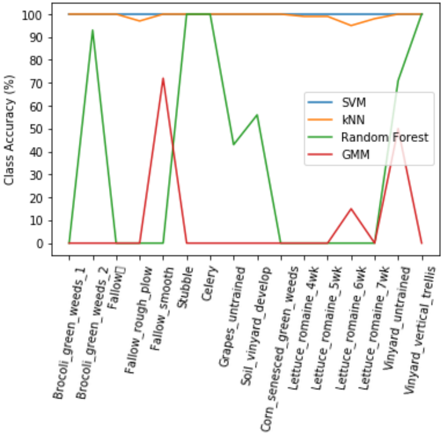


Figure: Classwise accuracy for different methods (Salinas)

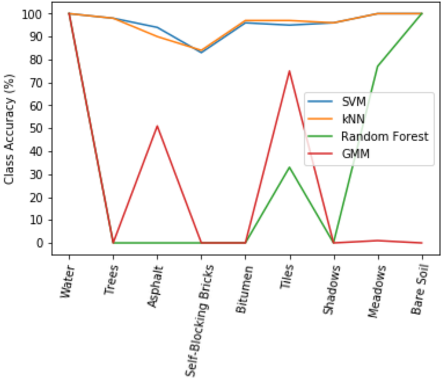
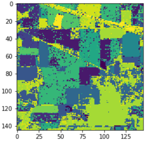


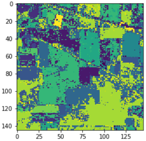
Figure: Classwise accuracy for different methods (Pavia)

Indian Pines Dataset

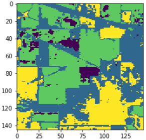
Figure: Classification results on Indian Pines dataset



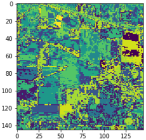
(a) SVM



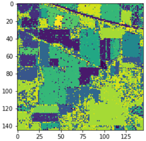
(b) kNN



(c) Random Forest



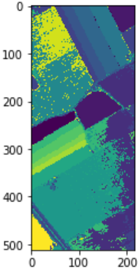
(d) GMM



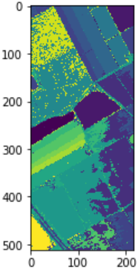
(e) Decision Tree

Salinas Dataset

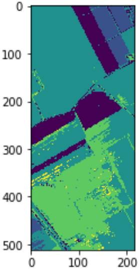
Figure: Classification results on Salinas dataset



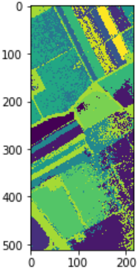
(a) SVM



(b) kNN



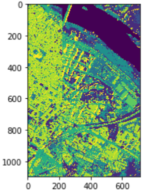
(c) Random Forest



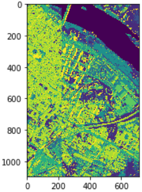
(d) GMM

Pavia Dataset

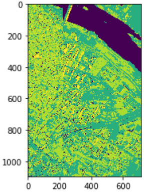
Figure: Classification results on Pavia dataset



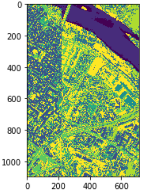
(a) SVM



(b) kNN



(c) Random Forest



(d) GMM

Thought Process while gradually converting to Deep Learning

- Start with Feature extraction techniques like PCA , ICA , LDA and see how the latent space representation distinguishes or decorrelates the sample points.
- Get familiar with inputting the data (class wise , index wise) to the model.
- Focussed on how to split the data keeping computation and size of data as the constraint.
- Hughes effect , on the training samples and minimum samples threshold , on classical and learning algorithm.
- To see how PCA whitening affects the spectral side (Contrast stretching), and improves data representation.

Thought Process while gradually converting to Deep Learning

- To start to understand complex decision boundaries , holding the notion of SVM in the picture.
- To observe hyperparameter tuning and all offline modelling techniques , in order to get more classification accuracy for the model.
- To see how Autoencoder performs as compared to the classical methods or combination of classical methods and (adding layers) building onto the autoencoder architecture.
- To see how various patch sizes capture the complex ground features , and to see if any filters can be applied to the input spatial data.

Next Tasks

Use the tensorflow library in Python to start making the neural architecture.

Complete class discrimination for a patch having many kinds of crops.

Conclusion Remarks

The classical methods have faster training and are more reliable , but they have a limitation when the complexity of the data increases.