JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Enhanced Hyperspectral Image Classification Using ILDA and Lightweight CNN

[1]Ayush Kumar, Department of Electronics and Communication Engineering, Goel Institute of Technology and Management, Lucknow
[2]Dr. Priyanka Jaiswal, Associate Professor, Department of Electronics and Communication Engineering, Goel Institute of Technology and Management, Lucknow

Abstract : Hyperspectral image classification (HIC) is a technique to classify each pixel in an image to a semantic class based on the spectral signature of materials. Hyperspectral images provide a large spectrum of very detailed information but are very high dimensional and have a limited number of labeled samples which makes classification difficult. In this paper, a lightweight cascaded deep convolutional neural network (LC-DCNN) is proposed which is based on the integration of spectral feature reduction methods: PCA, LDA, and ILDA. It consists of a series of five convolutional layers with ReLU activations, fully connected layers and a Softmax classifier to extract the spatial-spectral features that are captured by selecting or transforming the spectral data, then by forming local image patches and finally by processing them through a series of convolutional layers and fully connected layers. The accuracy of PCA, LDA and ILDA on Indian Pines and Salinas datasets were 98.05%, 99.00% and 99.51%, on Indian Pines and 98.62%, 99.63% and 99.71% on Salinas, respectively. The LC-DCNN architecture features 105 634 parameters, a patch size of 25 and 30 preserved spectral bands that were determined as optimal. Results indicate that class-aware spectral selection paired with a compact CNN can effectively balance classification performance and model complexity, though independent reproduction and complete reporting of methodology are crucial for validation.

Keywords- Hyperspectral image classification; Linear Discriminant Analysis; Improved LDA; lightweight cascaded CNN; spectral band selection; Indian Pines; Salinas; spatial–spectral features

1. Introduction

Hyperspectral imaging records a scene across many narrow, contiguous wavelength bands. In contrast to conventional RGB imagery, a hyperspectral image can contain hundreds of spectral measurements for each pixel. The measurements are very useful in the hyperspectral image classification for agricultural monitoring, land-cover mapping, vegetation analysis, mineral exploration and environmental observation [1]. The classification task assigns each pixel to one of a set of known categories based on its spectral response (and spatial context if available). The high correlation between neighboring bands, presence of noise in certain bands and relatively small number of labeled pixels compared to the number of input bands make the rich hyperspectral data challenging from a computational perspective. If used directly in deep learning models, all the bands would require more memory and more time to train the model, and could lead to overfitting. Any good classifiers must be able to retain information without increasing the size of the model and the number of calculations [2]. Some of the solutions that are commonly used to solve these problems are dimensionality reduction and feature selection. PCA takes into account the directions with high variance and LDA cares about separating the classes by using the class label. An improved version, ILDA, uses the L2-norm based differences to make the discrimination calculations more stable, and to choose bands that are able to express the difference between classes. Convolutional neural networks (CNNs) can extract local patterns in the patch of hyperspectral image, considering both spatial and spectral features [3]. The problem of training and deployment of large networks, though, can be expensive. To address these concerns, a lightweight cascaded deep convolutional neural network (DCNN) of five-convolution-layer, fully connected layers, and softmax classifier is discussed. Further, the study is written in the same manner as a research paper with methodology, summary of reported results, implications and limitations, but without verifying the results of the studies.

1.1 Aim and contributions

The aim is to examine a compact HIC framework that combines spectral reduction with a cascaded CNN. The principal contributions described in the source material are: (i) a comparison of PCA, LDA and ILDA as spectral reduction approaches; (ii) a five-layer lightweight CNN for spatial–spectral feature learning; (iii) an evaluation on Indian Pines and Salinas; and (iv) an analysis of classification accuracy, class-wise measures, patch size, retained spectral-band count and recognition time.

2. Background and Related Concepts

2.1 Hyperspectral data and classification

A hyperspectral image can be represented as a three-dimensional data cube with spatial height, spatial width and spectral depth. In the supplied study, the working image dimension is described as 145 × 145 × 220. A spectral vector is represented by each pixel and nearby pixels give spatial context. Classification is difficult if the spectral signature of different classes are similar or if the variation within the classes are large [4,8]. An effective model should be able to learn features that capture spectral differences and local spatial structure that are relevant to class.

2.2 Principal Component Analysis

PCA is an unsupervised linear transformation. It projects correlated spectral measurements onto orthogonal components ordered by the amount of variance they explain. Keeping only a subset of components can reduce the input dimensionality and computational burden. Since PCA does not directly use class labels, the components with the largest overall variance are not guaranteed to be the most discriminative for classification. In the supplied experiments, PCA provides the baseline feature-reduction strategy against which LDA and ILDA are compared [5].

2.3 Linear Discriminant Analysis

LDA is a supervised method, which requires labelled samples to obtain directions of the features that discriminate the classes. It's overall goal is to minimize dispersion in each class, and maximize dispersion between classes.

$$S_W = \sum_{c=1}^{C} \sum_{x_i \subset C_c} (x_i - \mu_c)(x_i - \mu_c)^T$$

$$S_B = \sum_{c=1}^{C} N_c(\mu_c - \mu)(\mu_c - \mu)^T$$

In the conventional formulation, the within class scatter matrix accounts for the scatter of the samples about their respective class mean and the between class scatter matrix accounts for the scatter of the class mean from the global mean [6]. A generalised eigenvalue problem of these matrices can be used to obtain a discriminant projection. Normally, the eigenvectors corresponding to higher values of discriminant eigenvalues are more useful for separating classes.

$$S_B w = \lambda S_W w$$

The spectral-band selection process by LDA is explained in six steps as follows: spectral-band samples identification; within-class disparity calculation; total/between-class disparity calculation; calculation of the eigenvalues and eigenvectors; determination of the contribution of each band; and sorting the bands according to their contribution [9].

$$Z = XW_d$$

where SW and SB denote the within-class and between-class scatter matrices, respectively; μc is the mean of class c, μ is the overall mean, and Wd contains the selected discriminant eigenvectors. These features can be used for subsequent hyperspectral image classification.

2.4 Improved LDA

The improved LDA strategy is described as replacing simple difference calculations with L2-norm-based calculations for within-class and between-class variability. The study states that this modification is intended to improve numerical stability and the accuracy of the discriminant solution [7, 10]. Then the same overall procedure is applied: calculate the class disparities with the L2 norm, build the disparity matrix, calculate the eigenvalues and eigenvectors, estimate the band contribution and rank the bands. The reported results indicate that the average accuracy of LC-DCNN with ILDA is higher than LC-DCNN with PCA and conventional LDA on both datasets. Because the full mathematical expression and implementation details cannot be found in the provided text, the details of the formulation of the ILDA should be included in a reproducible study.

3. Proposed Methodology

3.1 Overall framework

The proposed flow is data acquisition, data preprocessing, spectral reduction, patch generation, CNN feature extraction (Cascaded CNN), classification and evaluation. First, a labelled hyperspectral cube is given, along with reference labels. The spectral-reduction technique PCA, LDA or ILDA is used to get a smaller and/or more informative representation. The image is then segmented into local patches of size S × S. It is observed that usually an odd value of S is used to ensure that a distinct central pixel exists. The class of a patch is the reference class of the central pixel of the patch. Each patch is fed to the LC-DCNN that learns local and more abstract spatial–spectral representations before the class probabilities are produced using Softmax.

3.2 Spectral reduction and band selection

Spectral reduction is designed to reduce the input dimensions with redundancies or weak information before deep feature extraction. A comparison of PCA, LDA and ILDA is made and a configuration with 30 retained spectral bands is reported. The proposed sequence is not intended as an end unto itself, but rather as an input to a CNN that can take advantage of spatial neighbourhoods. The combined approach is significant since the dimension reduction of the spectrum can help reduce the number of inputs fed into the network, while the CNN can learn features from the remaining spectrum.

3.3 Patch construction

The image to be classified is split into overlapping local patches around pixels. It has been reported that the best configuration is a 25 × 25 spatial patch with 30 spectral bands which gives an input patch size of 25 × 25 × 30. Using a patch, the network can make use of neighbouring pixels as well, and not just one single pixel spectrum. Smaller patches could lack useful context, while larger patches would result in a greater amount of inputs and could even contain more than one class of pixels, which would make it harder to discriminate between them. Based on the evaluated settings, the studyreports that 25 × 25 offers a good balance.

3.4 LC-DCNN architecture

The LC-DCNN has five successive convolutional layers. Rectified Linear Unit (ReLU) activation is performed on the outputs of the convolutional layers. Convolutional filters learn patters locally and the number of feature maps is increased in a cascade. The output depths of the five convolutional stages are set in the configuration table in the source as 8, 16, 32, 64 and 128. The description of padding is “to maintain the spatial dimensions of the patch, stride 1”. The architecture is then composed of two full connected layers, and Softmax classifier. The activation function ReLU is defined by f(x) = max(0, x) which will set negative activation values to zero, adding non-linearity. The learned feature representation is then used in conjunction with fully connected layers to make the class decisions. The final class scores are then transformed into a probability distribution using Softmax and the class with the highest probability (most likely) is picked. In the source study, they have not specified the optimisation method, but it seems like mini-batch gradient descent. It also states a small positive parameter $\varepsilon = 10^{-8}$ to prevent division-by-zero errors and provides the value of the learning parameters as 0.001, 0.9 and 0.999, which are not clearly described in the text, and so should be checked against the original implementation before reproducing.

3.5 Network configuration

Stage	Reported output	Stride	Learnable parameters
Input	25 × 25 × 30	—	—
Convolution 1 + ReLU	25 × 25 × 8	1	2,168
Convolution 2 + ReLU	25 × 25 × 16	1	1,168
Convolution 3 + ReLU	25 × 25 × 32	1	4,640
Convolution 4 + ReLU	25 × 25 × 64	1	18,496
Convolution 5 + ReLU	25 × 25 × 128	1	73,856
Fully connected 1	10 × 1 × 1	—	5,130
Fully connected 2	16 × 1 × 1	—	176
Softmax classifier	16 classes	—	Not specified
Total reported	—	—	105,634

The source studyreports 105,634 total learnable parameters. This total is reproduced as reported; a replication should verify that the layer-by-layer counts and classifier accounting match the actual model implementation.

3.6 Training and evaluation

The source says it's implemented in MATLAB R2019b with an NVIDIA GPU with 512 Tensor Cores and 64 G of RAM. The accuracy and loss curves are used to measure the training behaviour. As iterations of training progress, accuracy is known to improve and loss goes down, as model parameters get better suited towards the training objective. But, training curves alone are not an evidence of generalisation. Training, validation and test partitions should be separately identified in the complete evaluation and the test data should not be used for band selection, tuning or model selection.

3.7 Evaluation metrics

This study presents accuracy, recall, precision and F1-score. Accuracy is number of correctly classified all samples evaluated. Precision is the percentage of correct predictions of a class, recall is the percentage of recovered samples of a class within the reference set. The F1 score is the harmonic mean of precision and recall. Precision for a class c can be computed as TPc/(TPc + FPc); recall can be computed as TPc/(TPc + FNc); F1 can be computed as 2 × precision × recall/(precision + recall), where TP, FP and FN are the number of true positives, false positives and false negatives in class c, respectively. Overall accuracy = number of correct predictions / number of predictions evaluated. Future reporting should differentiate between macro-averaged and class-weighted.

4. Experimental Results

4.1 Datasets

Indian Pines (IPs) and Salinas (SD) are reported in the evaluation. Indian Pines data are provided for 16 hyperspectral classes, which include agricultural classes of corn, soybean, grass and wheat, along with woods and built up classes. The Salinas results are for 16 classes: broccoli-green weeds, fallow fields, stubble, celery, grapes, vineyard and lettuce categories. The hyperspectral input described in the supplied study is of dimension 145 × 145 × 220. The text provided does not fully describe dataset-specific preprocessing and the number of train and test samples.

4.2 Overall accuracy comparison

Dataset	LC-DCNN + PCA	LC-DCNN + LDA	LC-DCNN + ILDA
Indian Pines	98.05%	99.00%	99.51%
Salinas	98.62%	99.63%	99.71%

In the case of Indian Pines, the overall accuracy is reported as 98.05% using PCA, 99.00% using LDA and 99.51% using ILDA. The ILDA result is 1.46 percentage points higher than PCA, and 0.51 percentage points higher than conventional LDA. In the case of Salinas, accuracy is improved from 98.62% with PCA to 99.63% with LDA and 99.71% with ILDA. The corresponding gains over ILDA with PCA, and over LDA, are 1.09 percentage points and 0.08 percentage points, respectively. The differences given are percentage-point differences computed from the figures presented in the study. The pattern is the same for both the datasets, class-aware LDA based methods perform better than PCA in the reported experiments and ILDA achieves the best overall accuracy. A possible explanation may be that the supervised methods maintain features that are beneficial for separating labelled classes, while the L2-norm modification in ILDA may benefit the band-ranking process. This interpretation is in line with the rationale of the method mentioned but the described comparison alone does not pinpoint the improvement's source.

4.3 Class-wise performance

Most categories in the Indian Pines table also contain data of high performance for each class. The reported accuracy is high for ILDA for each class, with 92.86% for Alfalfa, and 99.77% for Corn-notill and 99.60% for Corn-mintill, and 100% for several other classes, such as Corn, Grass-pasture, Grass-trees, Grass-pasture-mowed, Hay-windrowed, Oats, Soybean-notill, Soybean-mintill, Soybean-clean, Wheat, Woods, and Buildings-Grass-Trees-Drives and Stone-Steel-Towers. Table's average row shows that LC-DCNN + ILDA achieved 0.995 recall, 0.999 precision and 0.997 F1-score. The lower reported Alfalfa accuracy may be because more difficult categories may be included in the overall average. Class-specific errors may be due to class imbalance, number of labelled examples, and/or spectral similarity or mixed pixels, but these cannot be determined from the given tables. The Salinas' supplied table had good results for most classes, with 99.71% accuracy for the class ILDA. The results from the aggregates seem more obvious than some of the individual entries in the source table; some source table entries have line breaks, and some decimal values are truncated in the extracted text. When Salinas class-wise values are to be quoted in a final publication, the original formatted table should be checked prior to quoting any detailed values.

4.4 Effect of the number of spectral bands

The study demonstrates the results of a varying number of retained spectral-bands and claims that 30 bands provide the best performance for a 25 × 25 patch. With 30 bands, Indian Pines accuracy is reported as 98.05%, 99.00% and 99.51% for PCA, LDA and ILDA. For Salinas, the respective accuracies are 98.62%, 99.63% and 99.71%. Keeping too few bands can result in insufficient spectral representation, while keeping more bands can result in too much variability and computation. Such observations suggest that it would be more sensible not to take the view that either the largest or smallest set of features is best, but rather to select this number empirically.

4.5 Effect of patch size

Each central pixel will have spatial context, depending on the size of the patch. A 25 × 25 patch is better than the smaller and larger patches tested with 30 bands, according to the source. The accuracy for LC-DCNN + ILDA is claimed to be 99.51% for the Indian Pines and 99.71% for the Salinas data set. A medium-size patch can be adequate to capture neighbouring information to identify local patterns but without too much information from unrelated classes. This finding only applies to the datasets reported and the actual implementation; other sensors, spatial resolutions, and class layouts may need other patch sizes.

4.6 Recognition time and comparison with earlier methods

Method	Indian Pines time (s)	Salinas time (s)	Reported accuracy context
LC-DCNN + PCA	36	55	98.05% / 98.62%
LC-DCNN + LDA	38	62	99.00% / 99.63%
LC-DCNN + ILDA	38	60	99.51% / 99.71%
Deep CNN ensemble (Chen et al., 2019)	38	183	91.20% / 96.88%
C-CNN (Chang et al., 2022)	39	111	83.68% / 99.43%
SP-CNN-8 (Xie et al.,	880	—	93.57% / 96.12%

2021)			
SFE-SCNN (Gao et al., 2021)	—	—	98.72% / 99.56%
DAGCNN (Bai et al., 2022)	—	—	98.61% / —
DCNN (Shinde & Patidar, 2022)	36	—	98.20% / —

The study mentions 36, 38 and 38 seconds for PCA, LDA and ILDA respectively for Indian Pines. The times reported for Salinas are 55, 62 and 60 seconds. Hence, ILDA achieves the best reported accuracy, while having a recognition time near LDA and PCA. Several earlier methods are also included in the comparison table, some of which do not show a time. These comparisons should be taken with a pinch of salt as the hardware, batch size, data splits, preprocessing, timing scope, and software implementations may vary. The term 'recognition time' should also be clearly specified, such as including pre-processing and model loading, and whether this is measured per image, per test set or per sample.

5. Discussion

The main finding is that the three feature-reduction methods (ILDA, LDA, PCA) are consistently ranked according to their overall accuracy as reported on both datasets. This is in line with the idea that, for classification purposes, supervised, class-aware spectral selection can be beneficial, as opposed to general-purpose variance preservation. The proposed ILDA modification has a goal to take care of within-class and between-class differences using L2 norms, and the results support this goal. The network is lightweight with only 105,634 learnable parameters in 5 convolutional stages with a growth of feature depth from 8 to 128 channels, which may help in experimentation and reduce the memory consumption. But not only the number of parameters are relevant to the computation cost, but also the convolutional operations, the input size, and hardware details. Combining spatial and spectral information enables learning local spatial-spectral patterns, and a good context is obtained with 25 × 25 patches. Evaluation leakage, which may occur in hyperspectral imagery because of spatial overlap between the tests, needs to be taken care, especially in case of hyperspectral imagery. Moreover, the reporting should be improved for better reproducibility by adding all mathematical expressions, number of classes included in the sample, and specifying hyperparameters. Future studies should explicitly state the methodology, and make sure that the test-label is not leaked by the methodology used for feature selection. Previous methods are compared contextually: It is necessary to test them consistently on the same data partitions and methods in order to obtain a valid comparison, especially in the case of extremely high accuracy.

7. Conclusion and Future Work

In this paper, a compact hyperspectral image classification method by combining the spectral feature reduction with the lightweight cascaded deep convolutional neural network (DCNN) is proposed. The framework includes an evaluation of PCA, LDA, and an improved version of LDA technique, and finally, a five-stage DCNN over local image patches is used. LC-DCNN with improved LDA (ILDA) is obtained an overall accuracy of 99.51% on the Indian Pines data set and 99.71% on the Salinas data set outperforms the traditional PCA and LDA methods. The best one is 30 retained spectral bands with a patch size of 25×25 and a network of 105,634 learnable parameters. Future studies should include detailed equations and pseudocode for reproducibility, use different training, validation, and test data to avoid spatial leakage, and present a comprehensive performance measure. Further testing is recommended to evaluate the robustness and efficiency of the method with other hyperspectral data.

References

1. Paoletti, M. E., Haut, J. M., Plaza, J., & Plaza, A. (2019). Deep learning classifiers for hyperspectral imaging: A review. ISPRS Journal of Photogrammetry and Remote Sensing, 158, 279–317.
2. Fang, C. Wang, S. Li, J.A. Benediktsson, Hyperspectral image classification via multiple-feature-based adaptive sparse representation, IEEE Transactions on Instrumentation and Measurement 66(7) (2017) 1646-1657.
3. Melgani, F., & Bruzzone, L. (2004). Classification of hyperspectral remote sensing images with support vector machines. IEEE Transactions on Geoscience and Remote Sensing, 42(8), 1778–1790.
4. Guerri, M. F., Distante, C., Spagnolo, P., Bougourzi, F., & Taleb-Ahmed, A. (2024). Deep learning techniques for hyperspectral image analysis in agriculture: A review. ISPRS Open Journal of Photogrammetry and Remote Sensing, 12, Article 100062.
5. Kumar, D., & Kumar, D. (2021). Hyperspectral image classification using deep learning models: A review. Journal of Physics: Conference Series, 1950(1), Article 012087
6. Jia, S., Jiang, S., Lin, Z., Li, N., Xu, M., & Yu, S. (2021). A survey: Deep learning for hyperspectral image classification with few labeled samples. Neurocomputing, 448, 179–204.

7. Zhong, Z., Li, J., Luo, Z., & Chapman, M. (2018). Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework. IEEE Transactions on Geoscience and Remote Sensing, 56(2), 847–858.
8. Chen, Y., Jiang, H., Li, C., Jia, X., & Ghamisi, P. (2016). Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. IEEE Transactions on Geoscience and Remote Sensing, 54(10), 6232–6251.
9. H. Lee, H. Kwon, Going deeper with contextual CNN for hyperspectral image classification, IEEE Transactions on Image Processing 26(10) (2017) 4843-4855.
10. X. Zhang, Y. Liang, C. Li, N. Huyan, L. Jiao, H. Zhou, Recursive autoencoders-based unsupervised feature learning for hyperspectral image classification, IEEE Geoscience and Remote Sensing Letters 14(11) (2017) 1928-1932.