Deep Learning-Based Man-made Object Detection from Hyperspectral Data

Konstantinos Makantasis[1], Konstantinos Karantzalos[2], Anastasios Doulamis[2],
Konstantinos Loupos[3]

[1]Technical University of Crete, University campus, Kounoupidiana, 73100, Chania, Greece
kmakantasis@isc.tuc.gr
[2]National Technical University of Athens, Zografou campus, 15780, Athens, Greece
karank@central.ntua.gr, adoulam@cs.ntua.gr
[3]Institute of Communication and Computer Systems, Athens, Greece,
kloupos@iccs.gr

Abstract. Hyperspectral sensing, due to its intrinsic ability to capture the spectral responses of depicted materials, provides unique capabilities towards object detection and identification. In this paper, we tackle the problem of man-made object detection from hyperspectral data through a deep learning classification framework. By the effective exploitation of a Convolutional Neural Network we encode pixels' spectral and spatial information and employ a Multi-Layer Perceptron to conduct the classification task. Experimental results and the performed quantitative validation on widely used hyperspectral datasets demonstrating the great potentials of the developed approach towards accurate and automated man-made object detection.

1 Introduction

Automatic extraction of man-made objects, such as buildings, building blocks or roads, is of major importance for supporting several government activities, such as urban planning, cadastre, monitoring of protected nature areas, environmental monitoring and various GIS applications like map generation and update [1–4]. To this end, the accurate extraction and recognition of man-made objects from remote sensing data has been an important topic in remote sensing, photogrammetry and computer vision for more than two decades [5, 6]. Today, man-made object extraction is, still, an active research field, with the focus shifting to object detailed representation, the use of data from multiple sensors and the design of novel, generic, spatially accurate algorithms. Recent quantitative results from the International Society for Photogrammetry and Remote Sensing (Working Group III/4) benchmark on urban object detection and 3D building reconstruction [7] indicated that, in 2D, buildings can be recognized and separated from the other terrain objects, however, there is room for improvement towards the detection of small building structures and the precise delineation of building boundaries.

Most research and development efforts do consider a classification approach at the core of their man-made detection framework [3], while the detection is based mainly on satellite multispectral datasets [8, 9]. For multi-temporal data their efficient registration is a prerequisite [10]. However, due to recent advances in photonics, optics and nanotechnology hyperspectral satellite and airborne data are becoming more (openly) available and cost-effective while new object detection algorithms are introduced and validated [11]. In particular, for remote sensing applications and the classification of hyperspectral data, deep learning algorithms have recently indicated their very promising potentials with relative low classification errors in comparison with other state-of-the-art methodologies [12, 13].

In contrast to the approaches that follow the conventional paradigm of pattern recognition, which consists of the construction of complex handcrafted features from the raw data input [14, 15], deep learning models [16–18] are a class of machines that can learn a hierarchy of features by building high-level features from low-level ones, thereby automating the process of feature construction for the problem at hand. Furthermore, human brains perform well in object recognition tasks because of its multiple stages of information processing from retina to cortex [19]. Similarly, machine learning architecture with multiple layers of information processing construct more abstract and invariant representations of data, and thus are believed to have the ability of yielding higher classification accuracy than swallow architectures [20]. Convolutional Neural Networks (CNN), Stacked Auto-Encoders, Deep Belief Networks, etc. consist examples of deep learning models.

We tackle the problem of man-made object detection through a deep learning classification framework using on hyperspectral data. In particular, through the exploitation of a CNN we encode pixels' spectral and spatial information and a Multi-Layer Perceptron (MLP) is employed to perform the classification task. Experimental results and quantitative validation on widely used datasets showcasing the potential of the developed approach for accurate man-made object detection.

The rest of this paper is organized as follows; section 2 presents our approach overview by describing the fundamentals of the deep learning architecture and its application on hyperspectral data classification. Section 3 describes the architecture of the proposed system, experimental results and comparisons are presented in section 4 and section 5 concludes this work.

2 Approach overview

We consider the exploitation of a deep learning architecture for the detection of man-made objects using hyperspectral data, *i.e* the classification of each pixel to *man-made* or *non man-made* classes based on their spectral signatures and spatial properties. Spectral signatures are associated with the reflectance properties at every pixel for every spectral band, while spatial information is derived by taking into consideration its neighbors. The exploitation of spatial information is justified by the fact that, due to the nature of the problem, neighboring pixels is very probable to belong to the same class.

Towards this direction, high-level features that encode pixels' spectral and spatial information, are hierarchically constructed in an automated way using a CNN [16]. CNNs consist a type of deep models, which apply trainable filters and pooling operations on the raw input, resulting in a hierarchy of increasingly complex features.

2.1 Convolutional neural networks fundamentals

A CNN consists of a number of convolutional and sub-sampling layers. The input to a convolutional layer is a 3D tensor of dimensions $h \times w \times c$, where h, w and c correspond to input's height, width and channels respectively. The convolutional layer contains C trainable filters of dimensions $m \times m \times q$, where m is smaller than h and w and q is usually equal to c. Each filter is small spatially, but it extends through all channels of the input. By convolving each filter across the width and height of the input, 2D activation maps (feature maps) of that filter are produced.

Intuitively, the network learns filters that activate when they see some specific type of feature at some spatial position in the input. Stacking these activation maps for all filters along the depth dimension forms the full output volume, which is a 3D tensor of dimensions $h - m + 1 \times w - m + 1 \times C$ (convolution does not take into consideration the border of the input). The final output of a convolutional layer incorporates non-linearities, which are modeled through the application of non linear functions on the full output volume (*e.g.* sigmoid, tanh) and the addition of a bias term.

The output of the convolutional layers is fed to a sub-sampling layer, where each activation map is sub-sampled typically using the max polling operator over $k \times k$ contiguous regions. The sub-sampling layer incorporates scale and translation invariance to constructed activation maps. Again the output of the sub-sampling layer is a 3D tensor, whose dimensions are $\frac{h-m+1}{k} \times \frac{w-m+1}{k} \times C$.

A deep learning architecture may consist of many convolutional and sub-sampling layers. The last sub-sampling layer typically is sequentially connected with a fully-connected MLP, which is responsible for conducting the classification or regression task. The whole deep learning architecture is trained using the well known back propagation algorithm.

2.2 Convolutional neural networks for hyperspectral data

A hyperspectral image is represented as a 3D tensor of dimensions $h \times w \times c$, where h and w correspond to the height and width of the image and c to its channels (spectral bands). As mentioned before, CNNs produce global image features. However, man-made object detection can be seen a a pixel-based classification problem. In order to be able to exploit CNNs for man-made object detection, we have to decompose the captured hyperspectral image into *patches*, each one of which contains spectral and spatial information for a specific pixel.

More specifically, in order to classify a pixel $p_{x,y}$ at location (x, y) on image plane and successfully fuse spectral and spatial information, we use a square patch of size $s \times s$ centered at the same location. Let us denote as $l_{x,y}$ the class label of the pixel at location (x, y) and as $t_{x,y}$ the patch centered at pixel $p_{x,y}$. Then, we can form a dataset $D = \{(t_{x,y}, l_{x,y})\}$ for $x = 1, 2, \cdots, w$ and $y = 1, 2, \cdots, h$. Patch $t_{x,y}$ is also a 3D tensor with dimension $s \times s \times c$, which contains spectral and spatial information for the pixel located at (x, y) on image plane.

Moreover, tensor $t_{x,y}$ is divided into c matrices of dimensions $s \times s$ which are fed as input into a CNN, which hierarchically builds high-level features that encode spectral and spatial characteristics of pixel $p_{x,y}$. These features are fed to a MLP, which

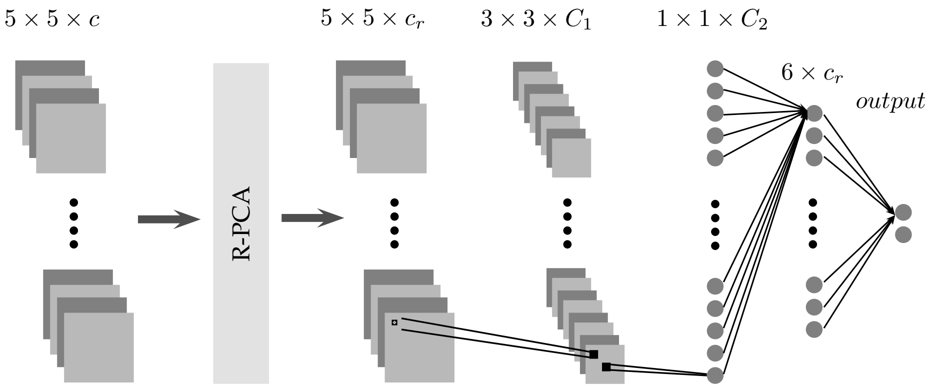

Fig. 1. Overall system architecture.

is responsible for the classification task. At this point, it has to be mentioned that after training the deep learning architecture is capable of classifying patches and not pixels. We assume that the label of the patch centered at location (x, y) on image plane must be the same with the pixel at the same location. Although, it is a strong assumption, for this specific problem at hand, it is valid for the vast majority of the pixels.

3 System architecture

In this section the developed system architecture is described. Firstly, the proposed approach for the dimensionality reduction of the input raw data is presented and then the structures of CNN and MLP are given.

3.1 Raw input data dimensionality reduction

CNNs are capable of hierarchically constructing high-level features in an automated way. Constructed features are the outcome of the convolution between the trainable filters and 2D network's input, which takes place during the training process. In contrast to RGB images that consist of three color channels, the *hundreds* of channels (network inputs) along the spectral dimension of a hyperspectral image may increase the computational cost of training and prediction phases to non acceptable levels.

However, the spectral signature of a material is very specific. Thus, pixels that depict the same material is expected to present very similar spectral responses [13]. Indeed, through a simple statistical analysis of pixels' spectral responses we observed two different things. Firstly, the variance of spectral responses of pixels that depict the same material is very small, and secondly, pixels that depict different materials, either they respond to different spectral bands or when they respond to the same spectral bands the values of their responses is very divergent. These two observations suggest that redundant information is present along the spectral dimension of the hyperspectral image. Therefore, a dimensionality reduction technique can be employed to reduce the dimensionality of the raw input data in order to speed up the training and prediction processes.

For dimensionality reduction, Randomized Principal Component Analysis (R-PCA) [21] is introduced along the spectral dimension to condense the whole image. Principal Component Analysis projects data to a lower dimensional space that preserves most of the variance by dropping components associated with lower eigenvalues. R-PCA limits the computation to an approximate estimate of principal components to perform data transformation. Thus, it is much more computationally efficient than PCA and suitable for large scale datasets, like hyperspectral images.

It should be noted that this step does cast away spectral information, but since R-PCA is applied along the spectral dimension, the spatial information remains intact. The number of principal components that are retained after the application of R-PCA, is appropriately set, in order to keep at least 99.9% of initial information. This is very important, since dimensionality reduction is conducted by taking into consideration the maximum allowed information loss and not a fixed number of principal components.

During the experimentation process on widely used hyperspectral datasets, 99.9% of initial information is preserved by using the first 10 to 20 principal components, reducing this way up to 20 times the dimensionality of the raw input.

3.2 Detection structure

After dimensionality reduction, each patch is a 3D tensor of dimensions $s \times s \times c_r$. Parameter c_r corresponds to the number of principal components that preserve at least 99.9% of initial information, while the parameter s determines the number of neighbors of each pixel that will be taken into consideration during classification task. The neighbors of a pixels are utilized to represent its spatial information.

During experimentation process we set the parameter s to be equal to 5, in order to take into consideration the closest 24 neighbors of each pixel. By increasing the value of s, the number of neighbors that are taken into consideration is increased and thus the computational cost of classification is increased, also. However, setting the parameter s to a value larger than 5, no further improvement on classification accuracy was reported in all experiments. On the contrary, increasing the value of s over 13, deteriorates classification accuracy. This is justified by the fact that our previous assumption, which states that the label of the patch centered at location (x, y) on image plane must be the same with the label of the pixel at the same location, is not valid for large s.

Having estimate the values of the parameters s and c_r, we can proceed with the CNN structure design. The first layer of the proposed CNN is a convolutional layer with $C_1 = 3 \times c_r$ trainable filters of dimension 3×3. This layer delivers C_1 matrices of dimensions 3×3 (during convolution we don't take into consideration the border of the patch). In contrast to conventional CNNs, we do not use a sub-sampling layer after the convolution layer, since we don't take into account any translation and scale invariance. For this reason the first convolutional layer is followed by a second convolutional layer with $C_2 = 3 \times C_1$ trainable filters. Again, the filters are 3×3 matrices.

The second convolutional layer delivers a vector with C_2 elements, which is fed as input to the MLP classifier. The number of MLP hidden units is smaller than the dimensionality of its input. In particular, we set the number of hidden units to equal

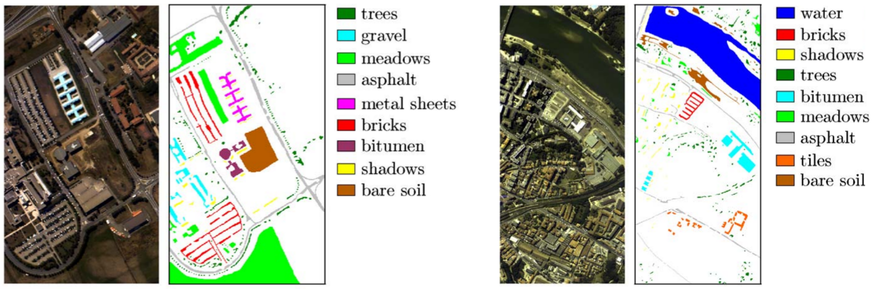

Fig. 2. Pavia University (left) and Pavia Centre (right) datasets along with their ground truths and labels of depicted materials (figures taken from [22]).

$6 \times c_r$. For training the deep learning architecture the standard back propagation algorithm was employed, in order to learn the optimal model parameters, *i.e.* minimize the negative log-likelihood of the data sets under the model parameterized by MLP weights and filters elements. The overall system architecture is presented in Fig.1.

4 Experimental results and validation

In this section we present the experimentation framework. Specifically, we present the dataset description, comparisons with other techniques and prediction capabilities of the proposed method.

4.1 Datasets for validating the proposed method

In our study we experimented and validated the developed framework with widely known and publicly available ROSIS hyperspectral datasets. In particular, we utilized i) the Pavia Centre and ii) Pavia University datasets, whose number of spectral bands are 102 and 103 respectively. Pavia Centre is a 1096×1096 pixels image, and Pavia University is 610×610 pixels. Some of the samples in both images contain no information and discarded before the analysis. The geometric resolution is 1.3 meters. Ground truths for both images contain 9 classes. Pixels of ground truth images that are depicted in white color are not annotated. Both datasets along with their ground truths and the labels of depicted materials are presented in Fig.2.

In this paper we focus on the detection of man-made objects. Thus, we grouped together pixels that depict man-made objects and discriminated them than the rest of the pixels. Pixels that are not annotated were not taken into consideration for classification purposes. For the Pavia University dataset, pixels that depict man-made objects are labeled as *asphalt*, *metal sheets*, *bricks* and *bitumen*, while for the Pavia Centre dataset pixels that depict man-made objects are labeled as *asphalt*, *tiles*, *bricks* and *bitumen*. Pixels that are labeled as *shadows* were not taken into consideration for classification purposes because it is doubtful whether they represent man-made objects.

Supervised training was conducted using pixels that depict man-made and not man-made objects. In particular, we split the tagged parts of the aforementioned datasets

Table 1. Quantitative evaluation results for Pavia University dataset. Split ratio ranges from 5% to 75% of the size of the whole dataset and corresponds to the size of the training set.

	Pavia University - misclassification error (%)					
Split ratio	5%	10%	20%	30%	50%	75%
Our approach	**2.5773**	**2.1455**	**1.3203**	**0.7729**	**0.2645**	**0.1861**
RBF-SVM	3.6213	3.2620	3.0960	2.8825	2.7685	2.6976
Linear SVM	4.4065	4.1332	3.9775	3.9693	3.9339	3.9166

Table 2. Quantitative evaluation results for Pavia Centre dataset. Split ratio ranges from 5% to 75% of the size of the whole dataset and corresponds to the size of the training set.

	Pavia Centre - misclassification error (%)					
Split ratio	5%	10%	20%	30%	50%	75%
Our approach	**0.2281**	**0.1223**	**0.0759**	**0.0663**	**0.0329**	**0.0086**
RBF-SVM	0.6598	0.5553	0.4730	0.4057	0.3876	0.3578
Linear SVM	1.1418	1.0988	1.0323	1.0810	1.1004	1.0563

into three sets, *i.e.* training, validation and testing data. During experimental validation we tested different split ratios in order to evaluate the classification accuracy of the proposed system when different amounts of data are available. For quantifying the classification accuracy of the proposed system we used the percentage of misclassification on testing set. Finally, validation set is used to enable early stopping criteria during the training process.

4.2 Quantitative evaluation

Our method was compared against Support Vector Machines (SVM) approaches that use Radial Basis Function (RBF) and linear kernels. In order to conduct a fair comparison SVM should be able to exploit pixels spectral and spatial information during training. For this reason each pixel was represented by its own spectral responses as well as the responses of its $(s \times s) - 1$ closest neighbors. In other words, a pixels at location (x, y) on image plane was represented by the spectral responses of a patch centered at the same location. Although this representation is a 3D tensor, it was flattened to form a 1D vector, in order to be utilized for training the SVM.

We conducted the experiments separately for each one of the datasets. During the experiments we formed six varying size training datasets, whose size ranges from 5% to 75% of the size of the whole dataset. Due to the fact that training, validation and testing set were formed by *randomly* selecting samples according to a pre-specified splitting ratio, we replicated each one of the experiments 25 times. Therefore, misclassification error corresponds to average error and classification accuracy to average accuracy.

Table 1 and Table 2 present the quantitative evaluation of our proposed method against SVM-based approaches. Our method outperforms SVM-based methods in both datasets and for all training set sizes. It is capable of achieving high classification accuracy scores for very small training datasets, while at the same time it avoids over-fitting when larger training datasets are used. Fig.3 visualizes the outcomes of the evaluation in terms of classification accuracy.

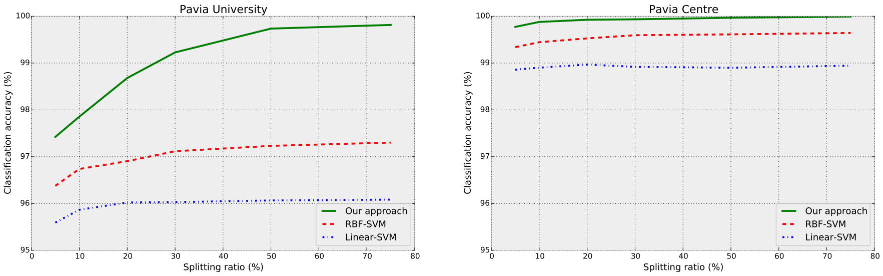

Fig. 3. Methods evaluation and comparison in terms of classification accuracy for both datasets.

4.3 Prediction capabilities

Furthermore, we examine the classification accuracy from a visual perspective using all the pixels of each image. In other words, pixels, corresponding to annotated and not-annotated regions, for each one of the hyperspectral images where classified using our deep learning approach. The classification results were obtained by setting the splitting ration equal to 5% and 75%.

The classification results after the application of the developed framework are presented in Fig.4. The resulted classification maps along with the ground truths are shown for both datasets. Green and red colored pixels correspond to man-made objects and non man-made object, respectively. Blue colored pixels in ground truth images correspond to non annotated pixels. As we can see, by fusing spectral and spatial information for each pixel, classification process results to the formation of compact areas, avoiding the presence of noisy scatter points, while at the same time it retains the shape and details of the depicted objects. Finally, the compactness of the areas is getting stronger as the size of the training set is increased.

5 Conclusions

In this paper, we propose a deep learning based approach for man-made object detection using hyperspectral data. Through the deep learning paradigm, our approach hierarchically constructs high-level features that encode pixels spectral and spatial information. Experimental validation of the proposed method and comparisons against SVM based methods showcase the high potential of the developed man-made objects detection system. Finally, among the future perspectives is the application of the developed framework for the detection of human behavior from hyperspectral video sequences.

Acknowledgements

This research was funded from European Unions FP7 under grant agreement n.313161, eVACUATE Project (www.evacuate.eu)

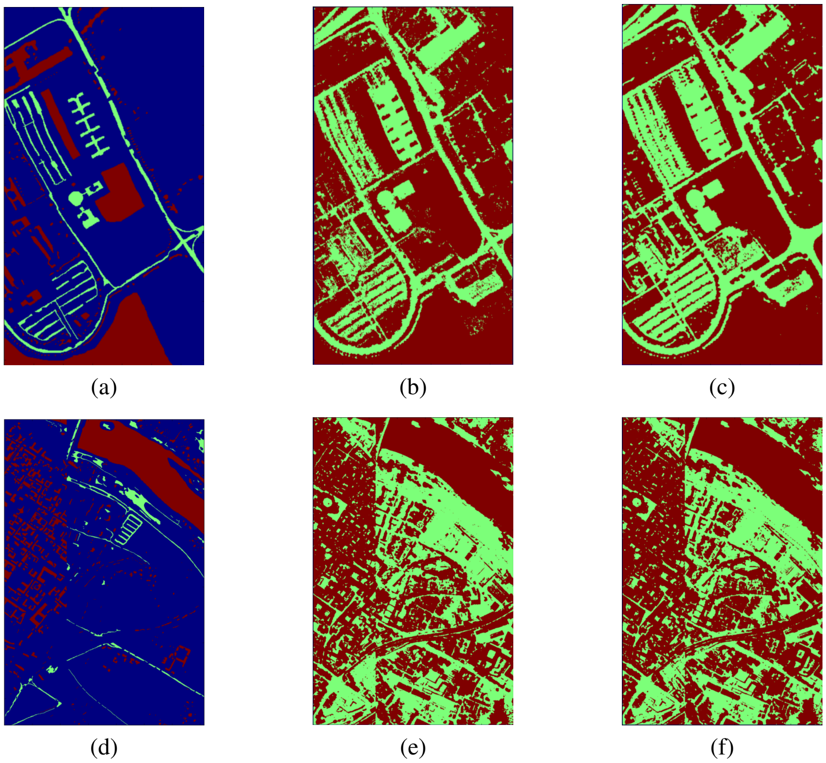

Fig. 4. Classification accuracy of the proposed method using annotated and non-annotated pixels of each hyperspectral image. (a) and (d) Ground truth images for Pavia University and Pavia Centre datasets, (b) and (e) classification results when splitting ratio equals 5% and (c) and (f) classification results when splitting ratio equals 75%.

References

1. Vescoukis, V., Doulamis, N., Karagiorgou, S.: A service oriented architecture for decision support systems in environmental crisis management. Future generation computer systems **28** (2012) 593–604
2. Pesaresi, M., Huadong, G., Blaes, X., Ehrlich, D., Ferri, S., Gueguen, L., Halkia, M., Kauffmann, M., Kemper, T., Lu, L., Marin-Herrera, M., Ouzounis, G., Scavazzon, M., Soille, P., Syrris, V., Zanchetta, L.: A Global Human Settlement Layer From Optical HR/VHR RS Data: Concept and First Results. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **6** (2013) 2102–2131
3. Karantzalos, K.: Recent Advances on 2D and 3D Change Detection in Urban Environments from Remote Sensing Data. In Helbich, M., Jokar Arsanjani, J., Leitner, M., eds.: Computational Approaches for Urban Environments. Geotechnologies and the Environment. (2015) 237–272

4. Florczyk, A., Ferri, S., Syrris, V., Kemper, T., Halkia, M., Soille, P.: A new european settlement map from optical fine scale remote sensed data. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (2015)
5. Gruen, A., Kuebler, O., Agouris, P.: Automatic Extraction of Man-made Objects from Aerial and Space Images I. Birkhaeuser, Basel (1995)
6. Karantzalos, K., Argialas, D.: A Region-Based Level Set Segmentation for Automatic Detection of Man-Made Objects from Aerial and Satellite Images. Photogrammetric Engineering and Remote Sensing **75** (2009)
7. Rottensteiner, F., Sohn, G., Gerke, M., Wegner, J.D., Breitkopf, U., Jung, J.: Results of the ISPRS benchmark on urban object detection and 3d building reconstruction. ISPRS Journal of Photogrammetry and Remote Sensing **93** (2014) 256 – 271
8. Vakalopoulou, M., Karantzalos, K., Komodakis, N., Paragios, N.: Simultaneous Registration and Change Detection in Multitemporal, Very High Resolution Remote Sensing Data. In: IEEE Computer Vision and Pattern Recognition Workshops (CVPRW'15). (2015)
9. Vakalopoulou, M., Karantzalos, K., Komodakis, N., Paragios, N.: Building Detection in Very high Resolution Multispectral Data with Deep Learning Features. In: IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2015). (2015)
10. Karantzalos, K., Sotiras, A., Paragios, N.: Efficient and automated multi-modal satellite data registration through mrfs and linear programming. IEEE Computer Vision and Pattern Recognition Workshops (2014) 1–8
11. Manolakis, D., Truslow, E., Pieper, M., Cooley, T., Brueggeman, M.: Detection algorithms in hyperspectral imaging systems: An overview of practical algorithms. Signal Processing Magazine, IEEE **31** (2014) 24–33
12. Chen, Y., Lin, Z., Zhao, X., Wang, G., Gu, Y.: Deep learning-based classification of hyperspectral data. Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of **7** (2014) 2094–2107
13. Makantasis, K., Karantzalos, K., Doulamis, A., Doulamis, N.: Deep Supervised Learning for Hyperspectral Data Classification through Convolutional Neural Networks. In: IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2015). (2015)
14. Camps-Valls, G., Bruzzone, L.: Kernel Methods for Remote Sensing Data Analysis. J. Wiley and Sons, NJ, USA (2009)
15. Camps-Valls, G., Tuia, D., Bruzzone, L., Atli Benediktsson, J.: Advances in hyperspectral image classification: Earth monitoring with statistical learning methods. Signal Processing Magazine, IEEE **31** (2014) 45–54
16. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86** (1998) 2278–2324
17. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313** (2006) 504–507
18. Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H.: Greedy layer-wise training of deep networks. In: Advances in NIPS 19. (2007) 153–160
19. Kruger, N., Janssen, P., Kalkan, S., Lappe, M., Leonardis, A., Piater, J., Rodriguez-Sanchez, A.J., Wiskott, L.: Deep hierarchies in the primate visual cortex: What can we learn for computer vision? PAMI, IEEE Trans. on **35** (2013) 1847–1871
20. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. Pattern Analysis and Machine Intelligence, IEEE Trans. on **35** (2013) 1798–1828
21. Halko, N., Martinsson, P., Tropp, J.: Finding structure with randomness: Stochastic algorithms for constructing approximate matrix decompositions, 2009. URL http://arxiv.org/abs/0909 (**4061**)
22. Mura, M.D., Villa, A., Benediktsson, J.A., Chanussot, J., Bruzzone, L.: Classification of hyperspectral images by using extended morphological attribute profiles and independent component analysis. Geoscience and Remote Sensing Letters, IEEE **8** (2011) 542–546