ISPRS Journal of Photogrammetry and Remote Sensing 121 (2016) 60–76

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

Semi-supervised hyperspectral classification from a small number of training samples using a co-training approach

Michał Romaszewski *, Przemysław Głomb, Michał Cholewa

Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Bałtycka 5, 44-100 Gliwice, Poland

ARTICLE INFO

Article history:
Received 29 March 2016
Received in revised form 1 July 2016
Accepted 19 August 2016

Keywords:
Hyperspectral classification
Co-training
Tracking-Learning-Detection

ABSTRACT

We present a novel semi-supervised algorithm for classification of hyperspectral data from remote sensors. Our method is inspired by the Tracking-Learning-Detection (TLD) framework, originally applied for tracking objects in a video stream. TLD introduced the co-training approach called P-N learning, making use of two independent 'experts' (or learners) that scored samples in different feature spaces. In a similar fashion, we formulated the hyperspectral classification task as a co-training problem, that can be solved with the P-N learning scheme. Our method uses both spatial and spectral features of data, extending a small set of initial labelled samples during the process of region growing. We show that this approach is stable and achieves very good accuracy even for small training sets. We analyse the algorithm's performance on several publicly available hyperspectral data sets.

1. Introduction

Rising interest in hyperspectral remote sensing has resulted in rapid development of methods for annotation, classification and segmentation of hyperspectral images (Bioucas-Dias et al., 2013). Many successful approaches (Li et al., 2013; Tilton et al., 2012; Wang et al., 2014) are based on semi-supervised learning and employ both spatial and spectral features of data. They aim to extend the initial training set with unlabelled samples in order to improve the classifier performance. These methods typically achieve better results than supervised approaches based only on spectral classifiers, as shown e.g. in Camps-Valls et al. (2007).

The paradigm of semi-supervised learning (Zhu and Goldberg, 2009) has been effectively applied beyond the field of hyperspectral imaging. One of the areas is long-term tracking (LTT) of unknown objects in a video stream. The efficient self-learning approach, based on co-training (Blum and Mitchell, 1998), was introduced in Kalal et al. (2012) in the form of the Tracking Learning Detection (TLD) framework.

The TLD approach decomposes the tracking problem into three sub-tasks: tracking, learning and detection. These sub-tasks were addressed by three simultaneously working components. The tracker was responsible for following the object position from frame to frame. The detector localised templates based on previously seen appearances. The learning component was, among others, responsible for estimation of detector errors and its update in order to avoid these errors in the future. The TLD used the P-N learning paradigm with two types of 'experts' (also called learners) evaluating the performance of the detector. P-expert was focused on missed detections while N-expert was focused on false positives. The analysis of stability performed in Kalal et al. (2012) has shown that well-designed experts can form a self-stabilising algorithm with the final error at an acceptable level.

We can find similarities between the tasks of hyperspectral classification (HC) and LTT. First, both problems have independent, multi-dimensional feature spaces. For the LTT the first feature space corresponds to visual similarity of objects, while the second one corresponds to their temporal behaviour, i.e. the position of the tracking window in consecutive frames. In HC the pixel similarity can be formulated as spatial (i.e. the distance between positions in the neighbourhood of each other) or spectral (i.e. the similarity of mixtures of materials). The second similarity lies in the fact that both LTT and HC include the detection component – LTT performs pattern matching for image fragments while HC performs spectral matching for individual image pixels. The third similarity is the assumption of predictability in the spatial dimension. LTT assumes that objects are moving in a predictable (e.g. locally linear) trajectory and their position can be estimated from previous frames using e.g. the Kalman filter (Harvey, 1990). Analogically, if we define an HC problem in terms of extending a small set of initial

* Corresponding author.
E-mail addresses: michal@iitis.pl (M. Romaszewski), przemg@iitis.pl (P. Głomb), mcholewa@iitis.pl (M. Cholewa).

http://dx.doi.org/10.1016/j.isprsjprs.2016.08.011
0924-2716/

labels in a way similar to label propagation or region growing, the knowledge about class labels of pixels from the training set can be extended to their neighbourhood. It has been observed, e.g. in Dópido et al. (2013) and Tan et al. (2015), that the class label is highly correlated with spatial similarity of pixels. In other words, we can expect that pixels located close to one another are likely to have the same label.

Based on those observations we implement the hyperspectral classification in a co-training framework with the P-N learning approach. Our P-expert assumes the same class labels for spatially close pixels. The N-expert detects pixels with similar spectra. Newly classified pixels are used to retrain the spectral classifier and to improve the spatial similarity model. Both experts are independent and while their limited scope of data makes them prone to errors, we show that their local accuracy is enough to stabilize the learning process. The schematics of the method are presented in Fig. 1.

The first contribution of this paper is the formulation of the hyperspectral classification problem in terms of the P-N learning paradigm. The second contribution is the implementation of the hyperspectral classifier based on proposed spatial and spectral experts. We present results on five data sets: the Indian Pines and Salinas Valley, University of Pavia, La Selva Biological Station and Madonna, Villelongue, France.

The rest of the paper is organised as follows: Section 2 presents the related work. Section 3 explains our method while Section 4 describes the experiment on real data and its results. Conclusions are provided in Section 5.

2. Related work

The classification of images acquired from an overhead perspective is a key component in information extraction for remote sensing applications, e.g. providing the details about land use, vegetation health or mineral concentrations (Campbell and Wynne, 2011). The complexity of this task results from diversity of objects and structures that are to be identified in images; in one case (Cheng et al., 2014) the requirement can be to detect instances from a broad set of objects that are only partially visible, in other (Cheng et al., 2015) it may be to classify existing ground areas to predefined land use classes.

Hyperspectral imaging uses a specially designed cameras to capture images with a detailed ground spectral reflectance at each pixel. This information allows for much more efficient classification of remote sensing data (Bioucas-Dias et al., 2013) as more information about object material composition or process state is available. At the same time the processing of such data remains challenging, mainly because of its high dimensionality and limited availability of training samples (Landgrebe, 2005) which results in the curse of dimensionality e.g. in the form of the Hughes phenomenon (Hughes, 1968).

A conceptually simple approach for classification of remote sensing images uses a spectral classifier to assign labels to each pixel independently; this reduces a remote sensing classification to a standard classification problem, which allows to use well-known high performance classifiers, e.g. Support Vector Machines (SVM) (Melgani and Bruzzone, 2004). The richness of spectral information allows for its effective use with a proper choice of feature extraction and/or selection methods (Bioucas-Dias et al., 2013). However, even more performance can be gained by simultaneous consideration of spectral and spatial (pixel neighbourhood) component (Plaza et al., 2009).

Many approaches aim to combine spatial and spectral features of the hyperspectral image (Fauvel et al., 2013). For instance, the assumption about continuity of neighbouring labels led to using the probability map produced by a classifier with spatial post-processing with Markov Random Fields (MRF), e.g. by combining Multinomial Logistic Regression with MRF in Li et al. (2012). Another approach is to combine spectral classification and image segmentation, as in Tarabalka et al. (2010a), where outputs from Watershed, Gaussian Mixtures and RHSEG segmentation were combined to produce a spectral-spatial classification map. Other examples include: using a separate local and global probability maps (Khodadadzadeh et al., 2014), composite kernel machines (Li et al., 2013), construction of minimum spanning forest from most significant image elements (Tarabalka et al., 2010b) or processing of arbitrary-shaped superpixels with sparse features (Fang et al., 2015).

Limited availability of labelled training samples resulted in the development of methods defining the hyperspectral classification task in the form of semi-supervised learning (Zhu and Goldberg, 2009). Its key idea is to use a small number of initial training samples and extend this training set with unlabelled data. Notable approaches to the use of semi-supervised learning for hyperspectral data classification include application of transductive Support Vector Machines (TSVM) e.g. Bruzzone et al. (2006), neural networks e.g. Ratle et al. (2010), graph-based approaches combined with composite kernels e.g. Camps-Valls et al. (2007) or SVMs with cluster kernels (Tuia and Camps-Valls, 2009). A combination of robust regression and discriminant analysis is used in Cheng et al. (2016); this approach uses graph based manifold learning for improving class discrimination. Active learning based on concatenation of spatial and spectral features with hierarchical segmentation was used in de Morsier et al. (2016). An SVM with segmentation-based ensemble (S^2SVMSE) was presented in Tan et al. (2014). It uses an image segmentation and refines the classification results through majority voting in segmented regions. An

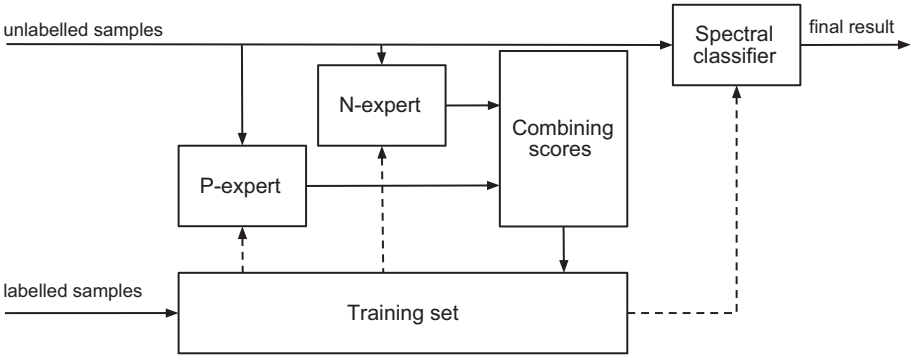

Fig. 1. The schema of the proposed algorithm. The initial training set is extended by two experts, scoring each pixel based on its spatial (P-expert) and spectral (N-expert) similarity. The scores are integrated and used to update the training set, which is then used for parametrization of the P- and N-experts and the spectral classifier.

approach based on label propagation on a spatial graph constructed using information extracted with the 2D Gabor filter was used in Wang et al. (2014). The assumption of global similarity of spectral features and local continuity of class labels was used in Dópido et al. (2013). Authors combined probabilistic classifiers with active learning algorithms in order to find the most informative samples. In a similar way, the method used in Tan et al. (2015) was based on observation of local pixel similarities. The authors showed how the neighbourhood of unlabelled pixels can be used to confirm the hypothesis about pixel labels obtained from the spectral classifier. When superpixels are considered, a custom spectral-spatial affinity score can be used for assignments of pixels to classes (Chen and Wang, 2016). The semi-supervised approaches form a very promising research direction in classification of hyperspectral remote sensing data, as they combine efficient spatial-spectral processing methods with application-realistic assumption of limited availability of labelled samples.

The Tracking-Learning-Detection (TLD) framework, presented in Kalal et al. (2012) was based on a semi-supervised approach called co-training (Blum and Mitchell, 1998). These methods assume that independent learners can mutually train each other if two independent feature spaces are available in the data. Learners are trained on a set of labelled samples and their votes are combined to classify the unlabelled ones. TLD implemented this learning process in the form of the P-N learning scheme which is similar to the technique of supervised bootstrap (Sung and Poggio, 1998).

3. Method

In this section we present the proposed co-training algorithm, inspired by the Tracking-Learning-Detection (TLD) (Kalal et al., 2012) approach. The TLD consists of three components. The first two are: the tracker that estimates the object motion between consecutive frames and the detector that treats every frame as independent, localising all appearances of the object in the image. The third component is a model that observes the performance of the tracker and the detector, identifies detection errors and performs retraining. Its key idea is that the detector errors can be identified by two processes called P-expert and N-expert. In the original TLD, the P-expert identified false negatives, N-expert identified false positives, and they mutually compensated for each other's errors.

We propose to use a similar self-learning scheme for classification of pixels in the hyperspectral image. As an analogy to the tracking of the moving object we perform the process of label-propagation for spatio-spectral data. In the TLD the task was to find the trajectory of a moving object from frame to frame. In our case, the task is to label the correct pixel in subsequent iterations of spatial label propagation.

The detection problem is similar in both cases. However, instead of using image templates (corresponding to pixel groups), we are comparing hyperspectral vectors of individual pixels. This is based on the assumption that closeness of two pixels in the spectral space corresponds to similar light reflectance properties of their materials.

The schema of the algorithm can be found in Fig. 1. Based on the training data, the processing sequence is as follows:

1. Unlabelled hyperspectral image pixels are scored by two independent experts, each producing a score value.
2. Scores are combined and the most probable pixels are used to extend the training set.
3. The parameters of scoring functions (experts) are updated based on the current training set.
4. Continue the above until the stopping criterion is met; then the final training set is used to train the spectral classifier (independent of experts).
5. Finally, the spectral classifier is used to predict labels for all unlabelled pixels in the image.

3.1. Preliminaries

Let $\mathfrak{C} = \{1, \ldots, C\}$ be a set of C class labels and $\mathfrak{P} = \{1, \ldots, n\}$ a set of n indices, indexing the n pixels of a hyperspectral image. By $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^b$ we denote hyperspectral vectors associated with image pixels, where b is the number of bands. A pixel label is denoted by $y \in \mathfrak{C}$. A set of training samples, called seeds, is defined as $\mathfrak{T} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_L, y_L)\}$, where L is the number of elements in the training set. $\mathfrak{T}^c$ denotes indices of seeds that belong to c class, that is $\mathfrak{T}^c = \{i \in \mathfrak{P} : y_i = c\}$, and L^c denotes the number of elements in $\mathfrak{T}^c$.

By $d(i,j) = \sqrt{(r_i - r_j)^2 + (c_i - c_j)^2}$ we denote the Euclidean distance between pixels with indices i and j, where (r_i, c_i) is the row and column of the pixel with index i. $d_{chess}(i,j) = \max(|r_i - r_j|, |c_i - c_j|)$ denotes the chessboard distance between them. By $w(i,j) = \|\mathbf{x}_i - \mathbf{x}_j\|$ we define the spectral Euclidean distance between pixels i and j. An l-th neighbourhood of a pixel indexed by i is defined as $N_l(i) = \{j \in \mathfrak{P} : d_{chess}(i,j) \leqslant l\}$; it is the set of all pixels with an index j such that $d_{chess}(i,j) \leqslant l$.

3.2. P-expert

The P-expert takes advantage of the spatial structure of the hyperspectral image – if a pixel label is known, its neighbours likely belong to the same class. A simple region growing step could be proposed by assuming that the neighbourhood $N_l(i)$ has the same label as a pixel i, as e.g. in Ugarriza et al. (2009). However, this property has been observed to hold locally in the immediate neighbourhood of a pixel. We thus propose that this probability should be dependent on the distance from the known class samples (seeds) and the score should be defined in the form of a function diminishing with distance. We model this gradient with a Gaussian density, and estimate the P-expert score for a set of training data S_L using Kernel Density Estimation (Duda et al., 2012). The density estimate for an image, based on a group of points $x_i, i = 1, \ldots, N$ is given by

$$\rho(y) \propto \frac{1}{Nh} \sum_{1}^{N} K(y - x_i) \tag{1}$$

where $K(u) = e^{-\frac{u^2}{2h^2}}$ denotes the Gaussian kernel and h is the smoothing parameter of the estimator called bandwidth. Considering the discretization of the image, we can integrate over the area of each pixel and compute the P-expert score S_p of pixel i within seeds of class c as

$$S_p^c(i) = \frac{1}{\theta h L^c} \sum_{j \in \mathfrak{T}^c} K(d(i,j)) \tag{2}$$

The θ parameter is the scaling factor corresponding to the slope of the gradient function in the region growing (Haralick and Shapiro, 1985), which controls the decay rate of the scoring function. It is used to limit the number of accepted samples and control the speed of the region growing.

3.3. N-expert

The N-expert uses the spectral similarity of pixels for the verification and possible rejection of a candidate sample. In a similar

way to Kalal et al. (2012) we can define our N-expert as a Nearest Neighbour classifier (NN), based on which we can generate a rejection score. To define the score of pixel i belonging to class c, we first locate its n closest spectral neighbours from the seeds set $\mathfrak{T}_i \subset \mathfrak{T}$. Then we compute

$$S_n^c(i) = 1 - \frac{\sum_{j \in \mathfrak{T}_i \cap \mathfrak{T}^c} w(i,j)^{-1}}{\sum_{j \in \mathfrak{T}_i} w(i,j)^{-1}} \tag{3}$$

where $w(i,j)$ is the Euclidean spectral distance between pixels i and j. This score formula is based on the probability estimation with the distance-weighted k-nearest neighbour rule (WKNN) (Biau and Devroye, 2015), and is at maximum when the pixel value is the least probable, based on the neighbours' estimate. The number of neighbours n is constant through the iterations of the algorithm and is the parameter of the method.

Typically classifiers require a large number of hyperspectral samples for effective classification, and may not work reliably when the training set is small (Plaza et al., 2009). This corresponds to the 'global' case, where the method is applied to the whole image. In the proposed approach the input is restricted only to local neighbourhood of the training set (the 'local' case), where one can expect that pixels are similar to those present in the training set.

3.4. Updating the model

To update parameters of both experts, the training set is extended. Therefore, scores for unlabelled pixels from both experts are combined. New seeds should have a high P-expert score $S_p^c(i)$ and low N-expert $S_n^c(i)$ score, corresponding to the situation where the sample is spatially close and not likely to be spectrally rejected. As a simple heuristic, we require that $S_p^c(i) > S_n^c(i)$. The final score of pixel i for class c is computed as

$$S^c(i) = S_p^c(i) - S_n^c(i) \tag{4}$$

The most probable label for pixel i is assigned as $y_i = \arg\max_c \{S^c(i)\}$. The pixel is then added to $\mathfrak{T}$ if $S^{y_i}(i) > 0$, corresponding to likelihood in favour of the P-expert. If $S^{y_i}(i) \leqslant 0$, in particular when $S_n^c(i) = 1$, the sample is rejected as always $S_p^c(i) \leqslant 1$; in other words, if the N-expert doesn't find any spectral neighbours of a given class, its negative vote is deciding.

Finally the scoring functions of both experts are retrained using the new, extended training set $\mathfrak{T}$. To illustrate the process, a visualisation of scoring for four consecutive iterations of the algorithm performed on the Indian Pines dataset is presented in Fig. 2.

3.5. Spectral classification

After the set number of iterations is reached, pixels that are still unlabelled are classified with the spectral classifier. Ideally, this final classification is only performed for a small number of samples, and can happen in two situations. First, when (e.g. due to performance constraints) the region growing process is stopped before being able to label all the pixels. Second, when classes are arranged in disjoint regions, with insufficient number of training seeds to cover all of them. In the latter case, the region growing process may be unable to 'jump' from one region to another, if the distance exceeds the P-expert's decay of the scoring function. For the final classification any spectral classifier trained on the final set $\mathfrak{T}$ can be used. In the next Section we evaluate 1-Nearest Neighbour and SVM for this role.

4. Experiments

To evaluate the proposed algorithm, we perform several experiments with a number of publicly available hyperspectral datasets. The objective of those experiments is to evaluate its quantitative and qualitative performance in different conditions of: varying scene composition (e.g. agricultural, urban and forest areas), spatial layout (e.g. class area regularity, convexity, class count imbalances), spectral features (e.g. diversity, type of mixing model) and resolution (pixel size).

4.1. Hyperspectral datasets

We have used six data scenarios; five correspond directly to the data sets used (Indian Pines, University of Pavia, Salinas Valley, Madonna, La Selva Biological Station) and one is based on the Indian Pines image, artificially modified to decrease class separation and reduce the regularity of field shapes.

4.1.1. Indian Pines

The hyperspectral image[1] was collected by the AVIRIS sensor over the Indian Pines region in Northwestern Indiana in 1992. It contains a mixed agricultural/forest area, early in the growing season. The image size is 145×145 pixels with 220 spectral channels, and spatial resolution of 20 m per pixel. Channels affected by noise and water absorption (104–108, 150–163 and 220) were removed leaving 200 channels. The ground reference data contains 16 classes representing mostly different types of crops. Because of the imbalance in the number of samples among classes, we reduced the number of training samples for small classes, in accordance with other works e.g. Li et al. (2013) and Wang et al. (2014). For classes Alfalfa, Grass/pasture-mowed, and Oats the number of training samples is at most $l = 10$.

4.1.2. Modified Indian Pines

Despite the diversity in spectral features of the Indian Pines dataset it is spatially regular and most of the classes are convex. Because of that, the majority of its area can be well classified using the spatial-spectral approach. While such regularity is common for many images representing agricultural and urban areas, it is not always the case. This particular layout is advantageous for spatially improved algorithms, as most spatial neighbours of a data point are very often in the same class. To test how proposed algorithm performs in less uniformly labelled areas, we changed the local spatial structure of the data without modifying global hyperspectral statistics of the dataset. This modification is carried out with the following algorithm, given parameters k, r:

1. $2k$ pixels $\{p_i\}_{i=1}^{2k}$ are selected from the ground truth, so that the distance from p_i to any data point y of a different non-background class is $d(p_i, y) > \frac{3}{2}r$.
2. For each pixel a patch is created by first generating a circle $c(p_i, r)$ with radius r and then selecting a random number of points $p_i^1, \ldots, p_i^t$ within $c(p_i, r)$ as centres of smaller circles $c(p_i^j, r_i^j)$, where $r_i^j \in (0, \frac{r}{2})$is selected randomly. That procedure generates irregular patch as a union of the circles $P_i = c(p_i, r) \cup c(p_i^1, r_i^1) \cup \ldots \cup c(p_i^t, r_i^t)$.
3. Background points from all the patches are eliminated.
4. The patches are randomly grouped into k pairs (P_i, P_j). For each pair we equalise the number of points in P_i, P_j by eliminating points from the more numerous patch.
5. For each pair, the patches P_i, P_j switch places.

[1] http://dynamo.ecn.purdue.edu/biehl/MultiSpec.

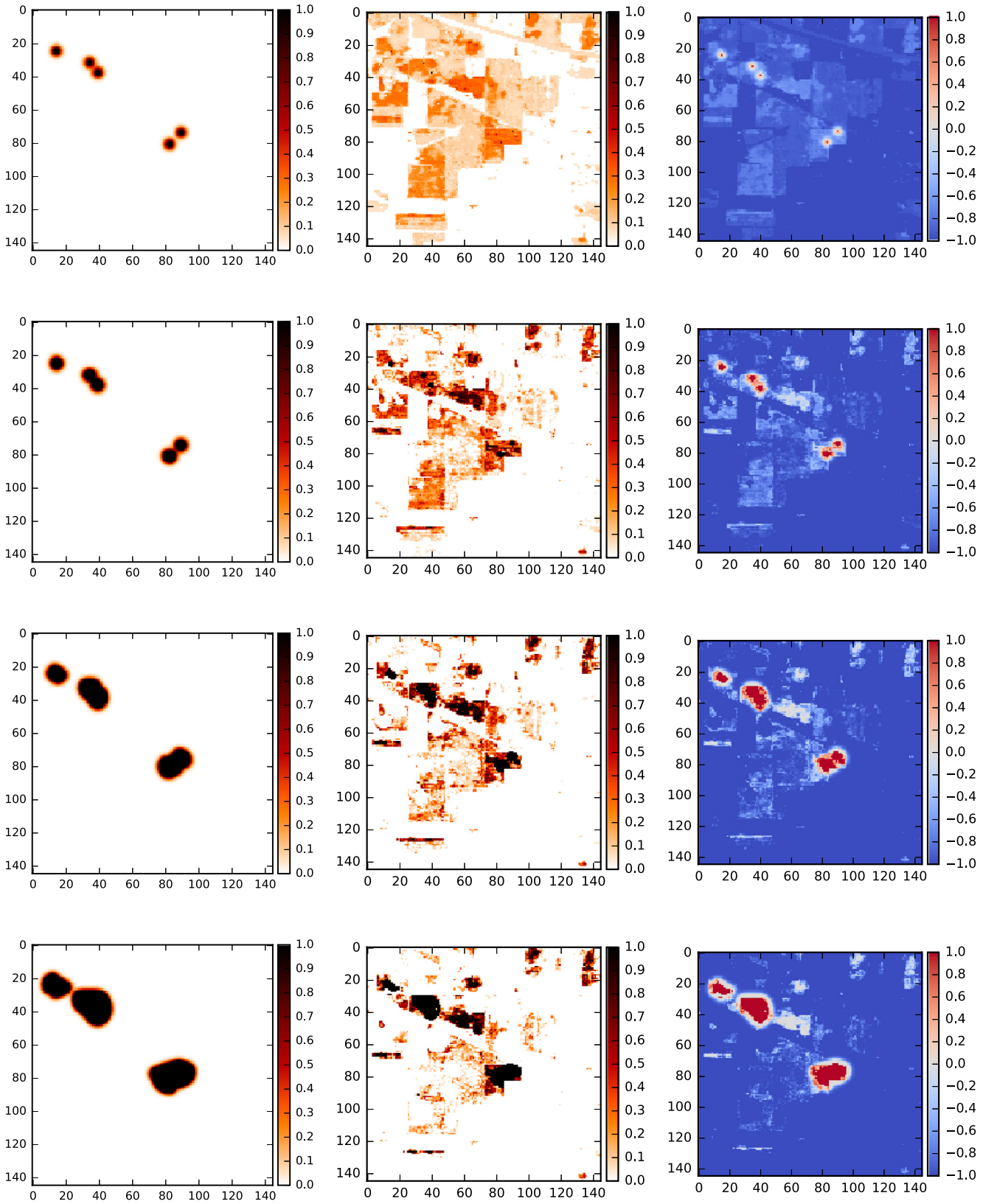

Fig. 2. Visualisation of the P and N-expert scores for four iterations of the algorithm. Results are provided for the Corn-notill class of the Indian Pines dataset and $n = 5$ initial seeds. The first column presents the P-expert score $S_p(i)$, the second column presents the N-expert score $S_n(i)$ and the third column presents $S_p(i) - S_n(i)$. Note that with subsequent iterations the N-expert gains more confidence as it gets more samples.

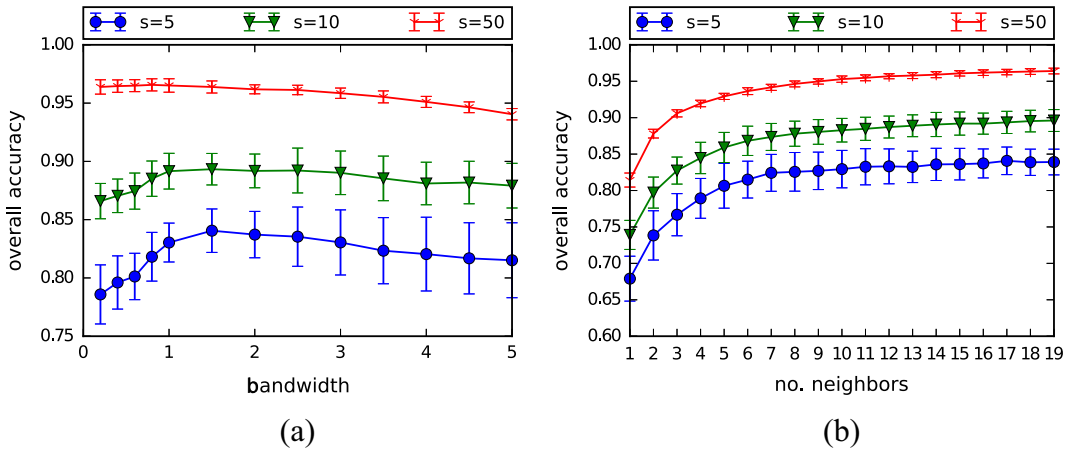

Fig. 3. The effect of algorithm parameters (P-expert's KDE bandwidth h, number of neighbours n for the N-expert classifier) on the classification accuracy for the Indian Pines dataset. Bars show the standard deviation. Values are presented for the training sets of 5, 10 and 50 initial samples/classes. In the experiments we use $h = 2$ and for the Indian Pines $n = 16$ (see text).

4.1.3. University of Pavia

The image[2] was captured using the ROSIS optical sensor over the urban area of the University of Pavia in Italy. The original Pavia University image size is 610×610, but a large part of this dataset contains no class information and was removed. The actual image size is therefore 610×340 with 115 spectral channels, with a spectral range of 0.43–0.86 and spatial resolution of 1.3 m per pixel. The ground reference data contains 9 classes representing different materials including gravel, trees, metal, bare soil, bricks and shadows.

4.1.4. Salinas Valley

The image was collected by the AVIRIS sensor over Salinas Valley in California. The image size is 512×217 with 224 spectral channels and spatial resolution of 3.7 m per pixel. Water absorption bands 108–112, 154–167 and 224 were discarded. The ground reference data contains 16 classes representing different bare soil, vegetables and vineyard fields.

4.1.5. Madonna, Villelongue, France

The hyperspectral dataset 'Madonna' was acquired by the Hyspex hyperspectral scanner over Villelongue, France. The observed area contains a forest and a village with some agricultural fields. The dataset is remarkable for the fact that it contains a mixing of material components (Dobigeon et al., 2014). This dataset does not contain a ground truth, but following the example outlined in Dobigeon et al. (2014) we estimated a ground truth for a selected 50×50 pixel forest area, where three kinds of foliage (chestnut, oak tree and additional non-planted tree component) are known to exist. We use the SISAL (Bioucas-Dias, 2009) algorithm for the endmember estimation and use $n = 100$ spectrally closest pixels to each endmember as a reference ground truth, obtaining three classes. The remaining pixels are marked as background, in a similar manner to the other datasets used in this study.

4.1.6. La Selva Biological Station

The dataset was captured by the HYDICE hyperspectral sensor over the La Selva Biological Station in March, 1998. It was created to test the methods for automated spices-level classification of individual tree crowns in a tropical rainforest. The processing and georeferencing procedures used for creation of the dataset are explained in detail in Clark et al. (2005). The original image was registered from multiple runs and its size is 2188×1725 and contains atmospherically-corrected surface reflectance of tree canopies. It has 161 bands ranging from 437.2 to 2434.3 nm with a spatial resolution of 1.6 m per pixel. The dataset contains 7 classes representing specific tree species. For performance reasons experiments are performed on the 400×400 fragment of the original image, where samples from all classes are present.

4.2. Estimation of parameters

The behaviour of the algorithm depends on the parameters of the two experts: for the N-expert, the shape of decision boundary is influenced by the number of neighbours n, for the P-expert, the region growing process is controlled by the bandwidth h and scaling θ parameters. Additionally, the algorithm runs for predefined number of iterations I.

The P-expert's parameters regulate the behaviour of the KDE function. The bandwidth h and scaling θ shape the spatial neighbourhood of labelled samples that is considered by the P-expert. In literature, often the first or third order neighbourhood ($N_1(\cdot)$ or $N_3(\cdot)$) of pixels are commonly considered for analysis (Ugarriza et al., 2009). Based on initial experiments and considering the results of Tan et al. (2015) we propose the KDE kernel to have a maximum score for the $N_1(\cdot)$ neighbourhood of labelled samples. As the θ parameter determines the decay rate of the kernel gradient function, for each class c we estimate $\theta = \min_i \left(\frac{1}{hL^c} \sum_{j \in \mathfrak{T}^c} K(d(i,j)) \right)$ where $i = N_l(k), k \in \mathfrak{T}^c$. The bandwidth h is responsible for the sensitivity of the P-expert and its effect on the algorithm's accuracy can be observed in Fig. 3. For a typical large scale hyperspectral image the best performing value range is $h = 1$ to $h = 3$. In our experiments we used $h = 2$.

The N-expert is controlled by a single parameter n: the number of neighbours used by the kNN probability estimator. The small training set can result in the N-expert being undertrained and 'overeager' to reject samples. Larger values help to stabilize this effect.[3] We propose to set this value to the number of classes; this value is sufficiently high in most hyperspectral classification scenarios for stable performance of the estimator. Visualisation of the algo-

[2] http://www.ehu.eus/ccwintco.

[3] Consider two N-experts working with samples randomly distributed in the spectral domain, one with $n = 1$ and the other with $n = 10$. The first one can be expected to reject the sample with probability $p = \frac{1}{n_{class}}$, the other $p = \frac{1}{n^{10}_{class}}$. We can see from this simple example that the influence of spectral noise decreases with n.

Table 1
Overall (OA), average accuracy (AA) and Cohen's kappa (κ) for five datasets: Indian Pines (IP), Salinas Valley (SV), Pavia University (PU), Modified Indian Pines (MIP), La Selva Biological Station (LS). Results are provided for $s = 5$–50 initial labelled samples and $I = 10$ iterations. Experiments were repeated 10 times.

s		SV(KNN)	SV(SVM)	PU(KNN)	PU(SVM)	IP(KNN)	IP(SVM)	MIP(KNN)	MIP(SVM)	LS(KNN)	LS(SVM)
5	OA	95.20 ± 1.1	95.35 ± 1.3	86.45 ± 2.8	88.11 ± 2.9	82.11 ± 2.7	81.77 ± 3.1	69.25 ± 3.0	69.36 ± 3.0	58.54 ± 8.2	59.82 ± 8.0
	AA	96.39 ± 1.0	96.48 ± 1.0	90.43 ± 1.3	91.53 ± 1.3	88.60 ± 1.2	88.64 ± 1.3	81.11 ± 1.9	81.26 ± 1.9	74.78 ± 3.9	76.17 ± 3.8
	κ	94.67 ± 1.2	94.83 ± 1.5	82.55 ± 3.3	84.64 ± 3.5	79.74 ± 3.0	79.38 ± 3.5	65.35 ± 3.3	65.46 ± 3.3	37.79 ± 6.7	39.24 ± 6.7
10	OA	97.18 ± 0.5	97.36 ± 0.5	92.91 ± 2.2	93.85 ± 2.2	89.18 ± 1.4	89.01 ± 1.7	76.28 ± 1.9	76.36 ± 1.9	72.31 ± 8.9	74.53 ± 9.1
	AA	98.04 ± 0.4	98.13 ± 0.4	94.47 ± 0.5	95.32 ± 0.6	92.58 ± 1.1	92.60 ± 1.1	86.41 ± 1.2	86.52 ± 1.2	86.03 ± 2.8	87.37 ± 3.0
	κ	96.86 ± 0.6	97.06 ± 0.5	90.73 ± 2.8	91.95 ± 2.8	87.70 ± 1.7	87.51 ± 1.9	73.20 ± 2.0	73.29 ± 2.1	55.40 ± 9.9	58.41 ± 10.5
15	OA	98.10 ± 0.5	98.30 ± 0.4	93.23 ± 3.2	93.77 ± 3.4	91.73 ± 2.0	91.80 ± 2.1	80.58 ± 2.2	80.63 ± 2.2	82.53 ± 7.2	84.11 ± 7.4
	AA	98.50 ± 0.3	98.60 ± 0.3	95.48 ± 1.0	95.96 ± 1.1	94.17 ± 1.1	94.26 ± 1.1	88.91 ± 1.4	88.99 ± 1.4	88.60 ± 2.9	89.37 ± 2.8
	κ	97.89 ± 0.5	98.11 ± 0.5	91.19 ± 4.0	91.89 ± 4.3	90.58 ± 2.2	90.66 ± 2.3	78.05 ± 2.4	78.11 ± 2.4	68.44 ± 9.7	71.02 ± 10.4
20	OA	98.42 ± 0.2	98.61 ± 0.2	96.41 ± 0.8	96.90 ± 0.9	93.15 ± 1.1	93.22 ± 1.1	83.23 ± 1.0	83.29 ± 1.0	87.02 ± 2.9	88.82 ± 2.6
	AA	98.65 ± 0.2	98.75 ± 0.2	97.03 ± 0.4	97.47 ± 0.4	94.87 ± 0.8	94.96 ± 0.7	90.42 ± 0.8	90.52 ± 0.8	91.30 ± 1.2	91.94 ± 1.1
	κ	98.24 ± 0.2	98.45 ± 0.2	95.25 ± 1.0	95.90 ± 1.2	92.19 ± 1.3	92.27 ± 1.2	80.99 ± 1.1	81.06 ± 1.1	74.94 ± 4.5	77.96 ± 4.2
25	OA	98.23 ± 0.5	98.38 ± 0.5	96.99 ± 1.3	97.40 ± 1.3	94.05 ± 0.9	94.00 ± 1.0	84.04 ± 1.6	84.12 ± 1.6	91.56 ± 2.3	92.47 ± 2.1
	AA	98.70 ± 0.2	98.78 ± 0.2	97.39 ± 0.6	97.77 ± 0.6	95.50 ± 0.5	95.52 ± 0.6	90.94 ± 0.9	91.03 ± 0.9	92.45 ± 1.9	92.82 ± 1.8
	κ	98.03 ± 0.5	98.19 ± 0.5	96.03 ± 1.7	96.57 ± 1.7	93.22 ± 1.0	93.16 ± 1.2	81.91 ± 1.7	82.00 ± 1.8	82.60 ± 4.3	84.27 ± 4.0
30	OA	98.75 ± 0.1	98.92 ± 0.1	96.90 ± 0.8	97.21 ± 0.9	94.49 ± 0.7	94.54 ± 0.7	85.50 ± 0.9	85.60 ± 0.8	–	–
	AA	98.87 ± 0.1	98.96 ± 0.1	97.43 ± 0.3	97.75 ± 0.4	95.58 ± 0.5	95.64 ± 0.5	91.80 ± 0.8	91.90 ± 0.8	–	–
	κ	98.60 ± 0.2	98.79 ± 0.2	95.90 ± 1.0	96.32 ± 1.2	93.71 ± 0.8	93.77 ± 0.8	83.53 ± 0.9	83.65 ± 0.9	–	–
35	OA	98.63 ± 0.2	98.76 ± 0.2	97.83 ± 0.7	98.08 ± 0.7	95.28 ± 0.4	95.36 ± 0.5	86.51 ± 1.1	86.57 ± 1.1	–	–
	AA	98.84 ± 0.1	98.91 ± 0.2	97.93 ± 0.3	98.20 ± 0.3	95.91 ± 0.3	95.97 ± 0.3	92.17 ± 0.7	92.23 ± 0.7	–	–
	κ	98.47 ± 0.3	98.62 ± 0.3	97.12 ± 0.9	97.46 ± 0.9	94.61 ± 0.5	94.69 ± 0.5	84.65 ± 1.2	84.72 ± 1.3	–	–
40	OA	98.72 ± 0.1	98.87 ± 0.1	97.90 ± 0.4	98.12 ± 0.4	95.61 ± 0.5	95.72 ± 0.5	86.52 ± 1.3	86.61 ± 1.3	–	–
	AA	98.96 ± 0.1	99.05 ± 0.1	98.01 ± 0.3	98.28 ± 0.3	96.33 ± 0.5	96.41 ± 0.5	92.38 ± 0.8	92.47 ± 0.8	–	–
	κ	98.57 ± 0.1	98.74 ± 0.1	97.21 ± 0.5	97.50 ± 0.5	94.97 ± 0.6	95.10 ± 0.5	84.67 ± 1.4	84.78 ± 1.4	–	–
45	OA	98.87 ± 0.1	99.01 ± 0.1	97.99 ± 0.7	98.18 ± 0.7	95.67 ± 0.6	95.76 ± 0.6	87.15 ± 1.0	87.27 ± 0.9	–	–
	AA	99.02 ± 0.1	99.10 ± 0.1	98.11 ± 0.2	98.35 ± 0.3	96.09 ± 0.5	96.19 ± 0.5	92.39 ± 0.8	92.49 ± 0.8	–	–
	κ	98.75 ± 0.1	98.90 ± 0.1	97.34 ± 0.9	97.58 ± 1.0	95.05 ± 0.7	95.14 ± 0.7	85.38 ± 1.1	85.51 ± 1.1	–	–
50	OA	98.93 ± 0.1	99.08 ± 0.1	98.24 ± 0.3	98.42 ± 0.3	96.18 ± 0.4	96.23 ± 0.5	86.99 ± 0.8	87.10 ± 0.8	–	–
	AA	99.09 ± 0.1	99.17 ± 0.1	98.27 ± 0.3	98.46 ± 0.3	96.21 ± 0.5	96.28 ± 0.5	91.80 ± 0.9	91.92 ± 0.9	–	–
	κ	98.81 ± 0.2	98.97 ± 0.1	97.66 ± 0.4	97.90 ± 0.4	95.62 ± 0.4	95.67 ± 0.5	85.19 ± 0.9	85.31 ± 0.9	–	–

Table 2
Comparison of our method, denoted as PNGrow, with the results reported in: [1] Wang et al. (2014), [2] Tan et al. (2014), [3] Tan et al. (2015), for the Indian Pines Dataset, bolded values denote the best result for a column.

Method	Training samples				
	5	10	15	20	25
SVM[1]	50.23 ± 1.74	55.56 ± 2.04	58.58 ± 0.80	62.93 ± 0.64	65.12 ± 0.63
LapSVM[1]	52.31 ± 0.67	56.36 ± 0.71	59.99 ± 0.65	64.13 ± 1.19	65.36 ± 0.62
TSVM[1]	62.57 ± 0.23	63.45 ± 0.17	65.42 ± 0.02	64.43 ± 0.20	67.68 ± 1.67
SCS^{3}VM[1]	55.42 ± 0.35	60.86 ± 5.08	67.24 ± 0.47	68.34 ± 1.57	72.42 ± 1.21
SS-LPSVM[1]	56.95 ± 0.95	64.74 ± 0.39	78.76 ± 0.04	80.29 ± 0.80	84.11 ± 0.08
GCK[1]	69.63 ± 3.22	80.79 ± 1.69	85.32 ± 1.88	88.19 ± 0.93	89.21 ± 1.46
S^2SVM[2]	**82.33**	85.47	91.72	*n/d*	*n/d*
MLR[3]	70.99	86.01	90.44	*n/d*	*n/d*
PNGrow	82.11 ± 2.69	**89.18 ± 1.54**	**91.80 ± 2.07**	**93.22 ± 1.10**	**94.05 ± 0.86**

Table 3
Comparison of our method, denoted as PNGrow, with the results reported in: [1] Wang et al. (2014), [2] Tan et al. (2014), [3] Tan et al. (2015) for the University of Pavia dataset, bolded values denote the best result for a column.

Method	Training samples				
	5	10	15	20	25
SVM[1]	53.73 ± 1.30	61.53 ± 1.14	60.43 ± 0.94	64.89 ± 1.14	68.01 ± 2.62
LapSVM[1]	65.72 ± 0.34	68.26 ± 2.20	68.34 ± 0.29	65.91 ± 0.45	68.88 ± 1.34
TSVM[1]	63.43 ± 1.22	63.73 ± 0.45	68.45 ± 1.07	73.72 ± 0.27	69.96 ± 1.39
SCS[1]	56.76 ± 2.28	64.25 ± 0.40	66.87 ± 0.37	68.24 ± 1.18	69.45 ± 2.19
SS-LPSVM[1]	69.60 ± 2.30	75.88 ± 0.22	80.67 ± 1.21	78.41 ± 0.26	85.56 ± 0.09
S^2SVM[2]	85.64	90.93	**96.64**	*n/d*	*n/d*
MLR[3]	76.46	85.47	88.08	*n/d*	*n/d*
PNGrow	**88.11 ± 2.87**	**93.85 ± 2.23**	93.77 ± 3.42	**96.90 ± 0.90**	**97.40 ± 1.34**

Table 4
Comparison of our method, denoted as PNGrow, with the results reported in [1] Wang et al. (2014) for the Salinas Valley Dataset dataset, bolded values denote the best result for a column.

Method	Training samples				
	5	10	15	20	25
SVM[1]	73.90 ± 1.91	75.62 ± 1.73	79.08 ± 1.45	77.89 ± 1.20	78.05 ± 1.49
LapSVM[1]	75.31 ± 2.31	76.34 ± 1.77	77.93 ± 2.42	79.40 ± 0.73	80.56 ± 1.33
TSVM[1]	60.43 ± 1.40	67.47 ± 1.05	69.12 ± 1.32	71.03 ± 1.78	71.83 ± 1.16
SCS[1]	74.12 ± 2.44	78.49 ± 2.02	81.83 ± 0.93	81.22 ± 1.27	77.08 ± 0.80
SS-LPSVM[1]	86.79 ± 1.75	90.36 ± 1.35	90.86 ± 1.36	91.77 ± 0.96	92.11 ± 1.07
PNGrow	**95.35 ± 1.33**	**97.36 ± 0.49**	**98.30 ± 0.45**	**98.61 ± 0.19**	**98.38 ± 0.46**

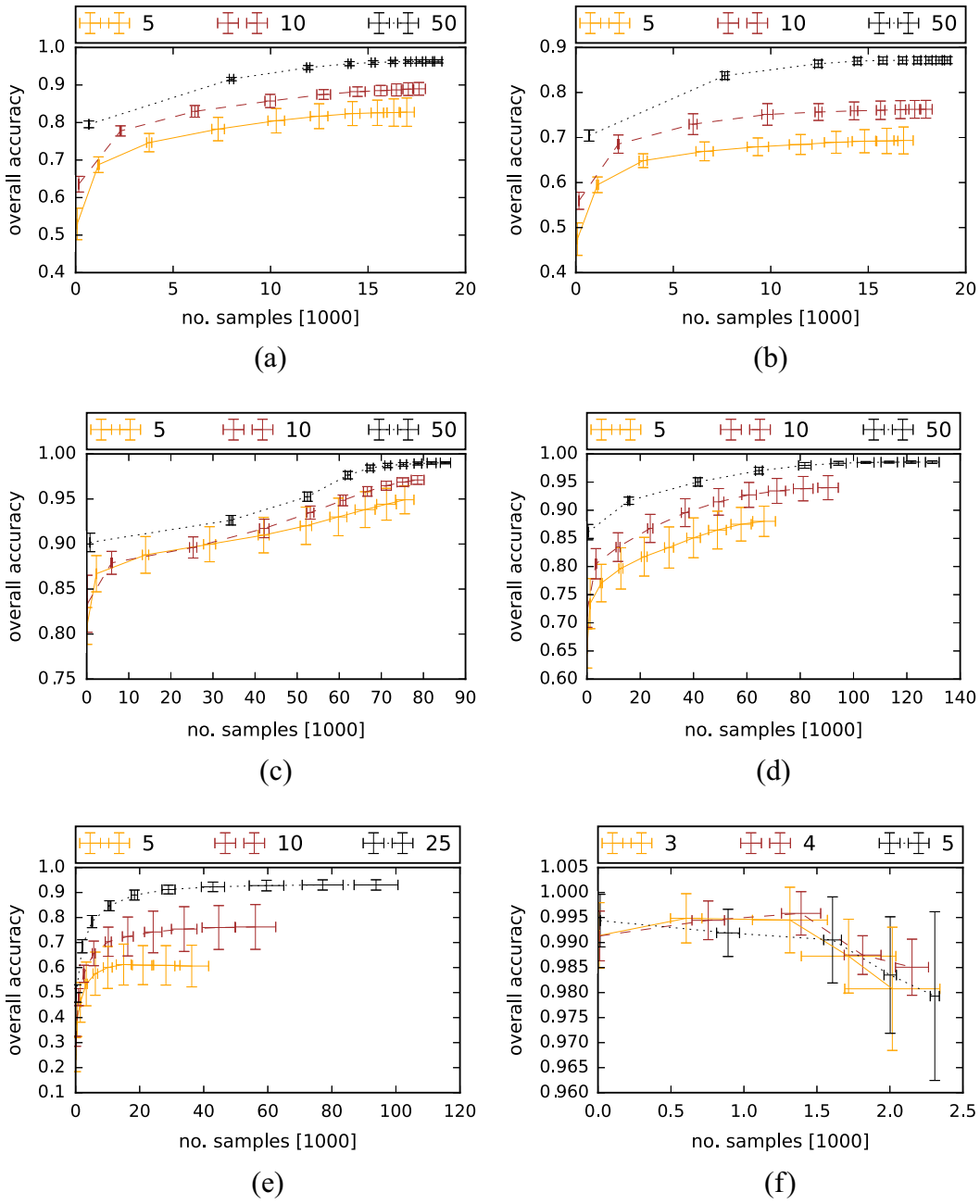

Fig. 4. Algorithm accuracy presented as a function of the number of samples labelled by the semi-supervised learning. Results are presented for 5–50 samples for the six datasets: (a) Indian Pines, (b) Modified Indian Pines, (c) Salinas Valley, (d) University of Pavia; 5–25 samples for (e) La Selva Biological Station and 3–5 samples for (f) Madonna. Each mark on the plot represents the standard deviation for $n = 10$ repeats of one iteration. Note that the number of new seeds gained at each iteration is dependent on the number of starting points, so subsequent markers are not aligned.

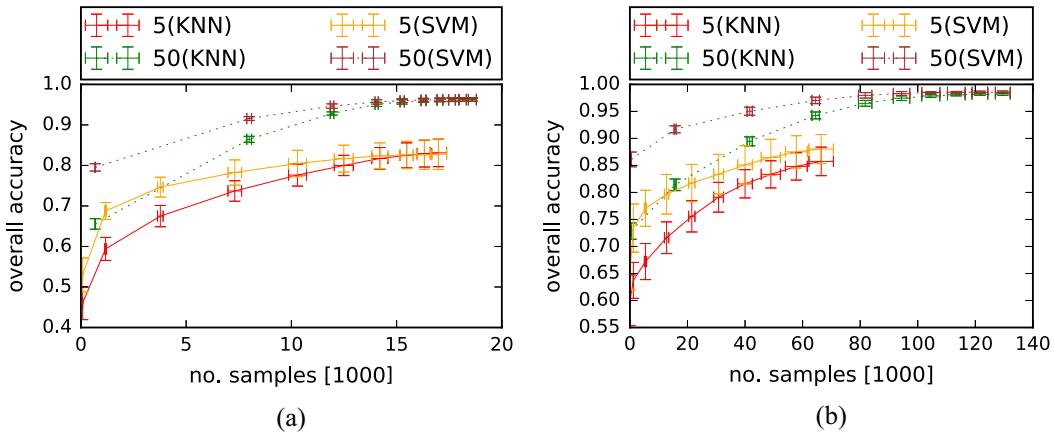

Fig. 5. A comparison of SVM and 1-NN in the role of the final spectral classifier. Results are presented for 5 and 50 samples for two representative datasets: (a) Indian Pines (b) University of Pavia. For datasets with a simple spatial structure e.g. (a), even for small number of initial training samples the difference in the final performance (denoted by the end of the curves) of both spectral classifiers is minimal. This results from the fact that the majority of the image is classified by experts. On the other hand, for spatially challenging datasets e.g. (b) the SVM achieves better final accuracy for small initial training sets.

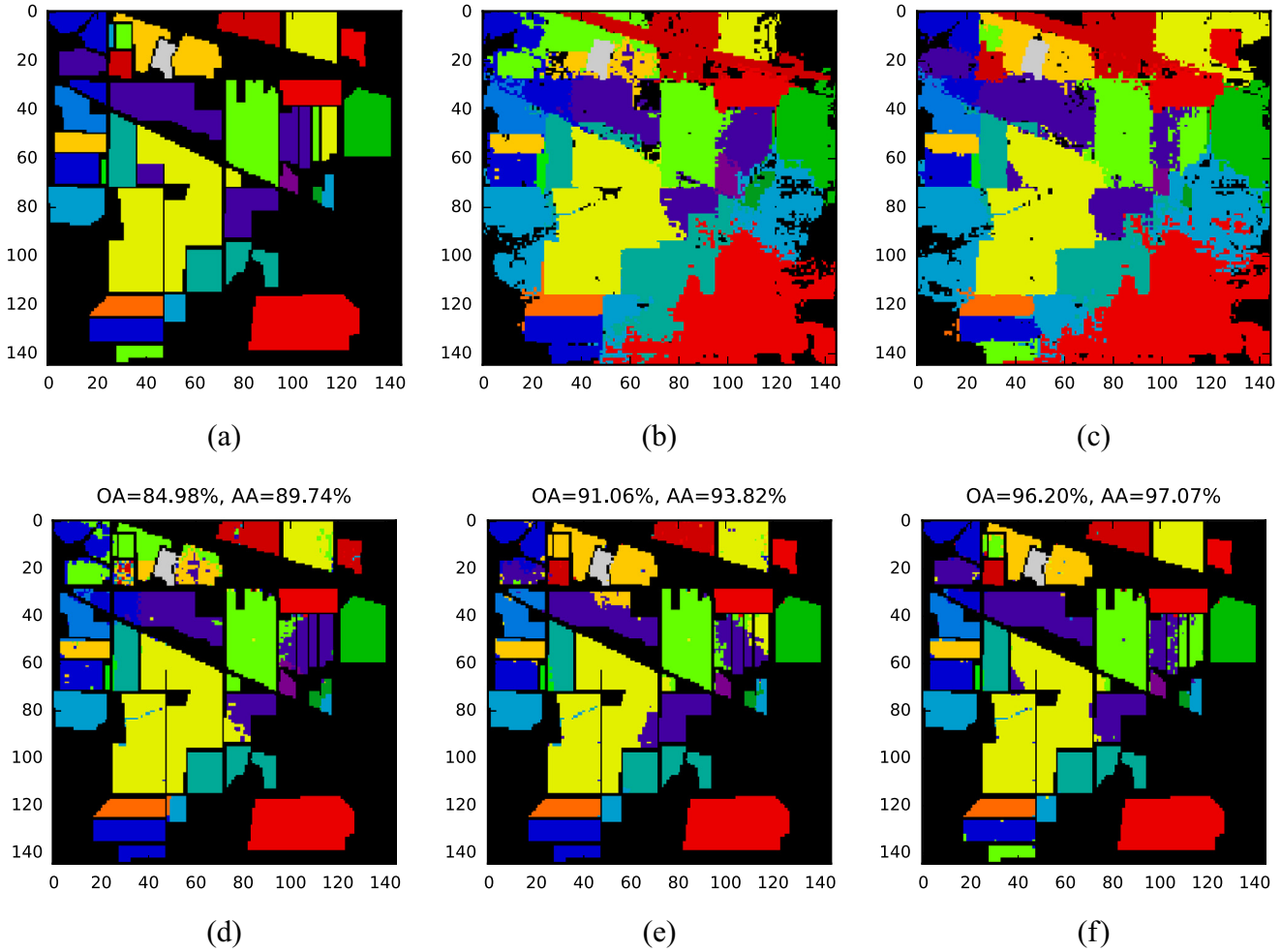

Fig. 6. Classification results for the Indian Pines dataset. Plot (a) presents the ground truth. Plots (b) and (c) are the extended training sets for 5 and 50 initial samples/classes. The bottom plots present an example of final classification results for initial 5 (d), 10 (e) and 50 (f) samples/classes.

rithm's accuracy as a function of n for the Indian Pines dataset is presented in Fig. 3.

The number of iterations I is a straightforward choice, as the accuracy generally increases with each iteration. With more iterations, more pixels are classified with the P/N experts part, and less with the comparatively weaker spectral-only classifier. The only consideration is the processing time. Note that without supplying this value, the algorithm will run through all unlabelled samples,

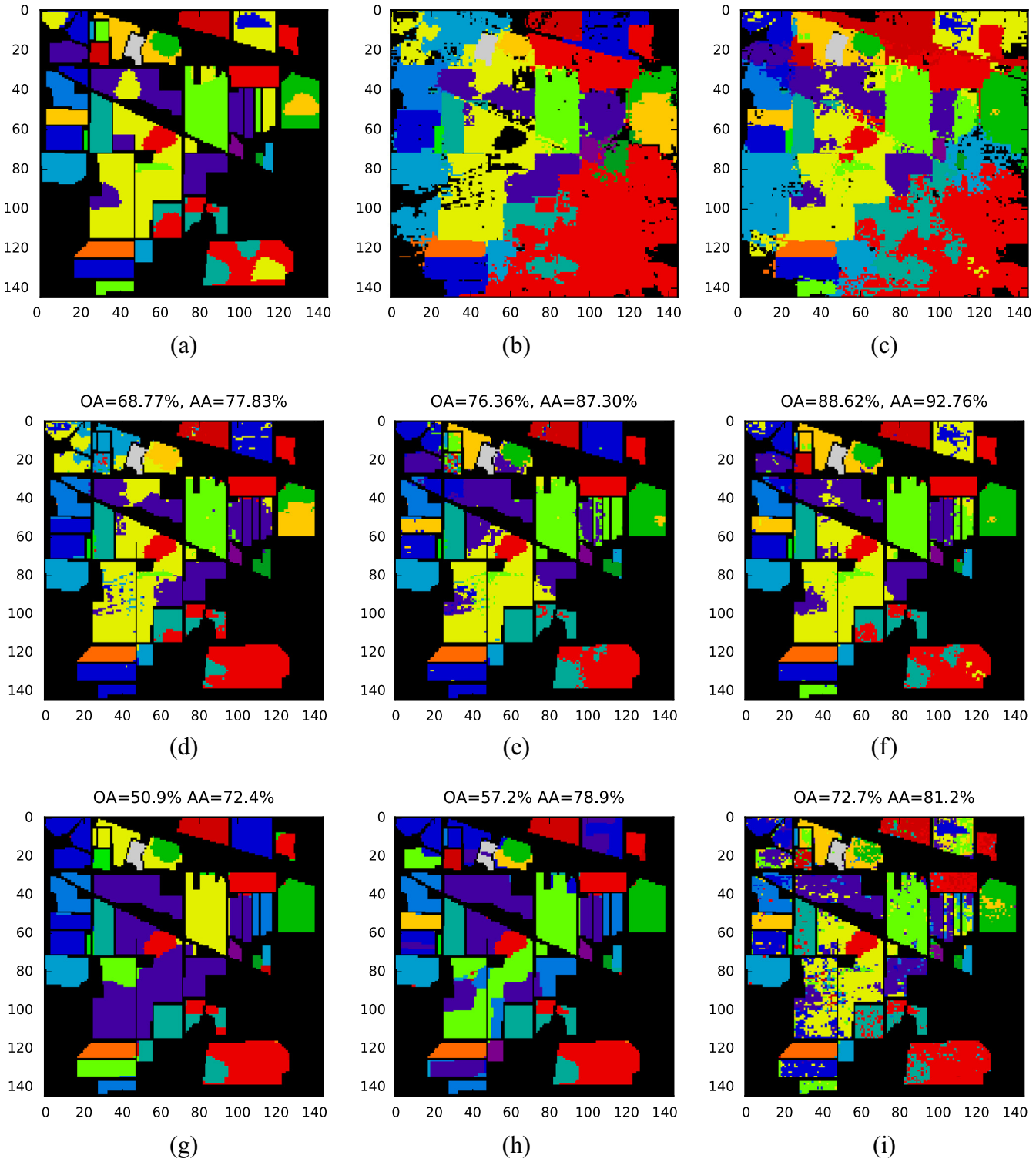

Fig. 7. Classification results for the Modified Indian Pines dataset. Plot (a) presents the ground truth. Plots (b) and (c) present extended classes for 5 and 50 training samples. The bottom plots present an example of final classification results for initial 5 (d), 10 (e) and 50 (f) training samples. The bottom plots present reference classification results obtained using the SVM + MRF classifier and 5 (g), 10 (h) and 50 (i) training samples.

and those rejected by P/N experts will be decided on by the spectral classifier. In the experiments, we set $I = 10$ as a value balancing processing time and classification performance, which results in P/N experts classifying e.g. $\geqslant 90\%$ ground truth samples for the Indian Pines dataset.

In the final step the parameters of the spectral classifier are selected using a standard cross validation procedure. We evaluate two classifiers: SVM with radial basis function kernel and a simple 1-NN.

4.3. Results

The algorithm was evaluated in each previously discussed scenario with varying number of initial samples per classes ($s = 5$ to

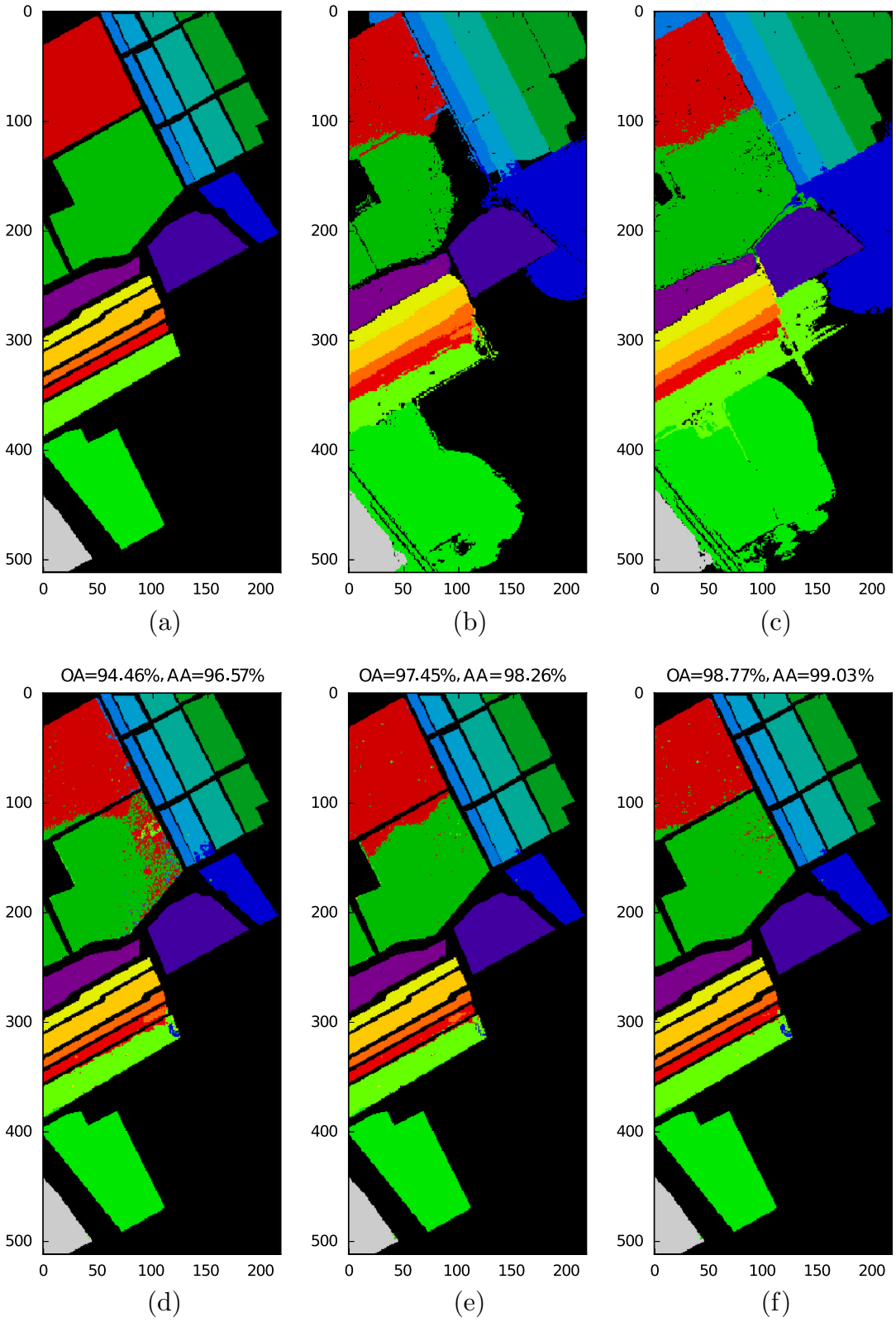

Fig. 8. Algorithm results for the Salinas Valley dataset. Plot (a) presents the ground truth. Plots (b) and (c) are the extended training sets for 5 and 50 initial training samples. The bottom plots present an example of final classification results for 5 (d), 10 (e) and 50 (f) initial training samples.

$s = 50$ with increments of 5[4]). The presented results were averaged over $n = 10$ independent runs of the algorithm. For the first three scenarios (Indian Pines, University of Pavia, Salinas Valley), the results were compared with the state of art referenced in the literature. For the remaining three scenarios (Modified Indian Pines, Madonna, La Selva Biological Station) the results were compared with baseline method of SVM classifier with MRF spatial

[4] Except for Madonna, where the values used were $s = 3, 4, 5$ and La Selva Biological Station with $s = 5$ to $s = 25$.

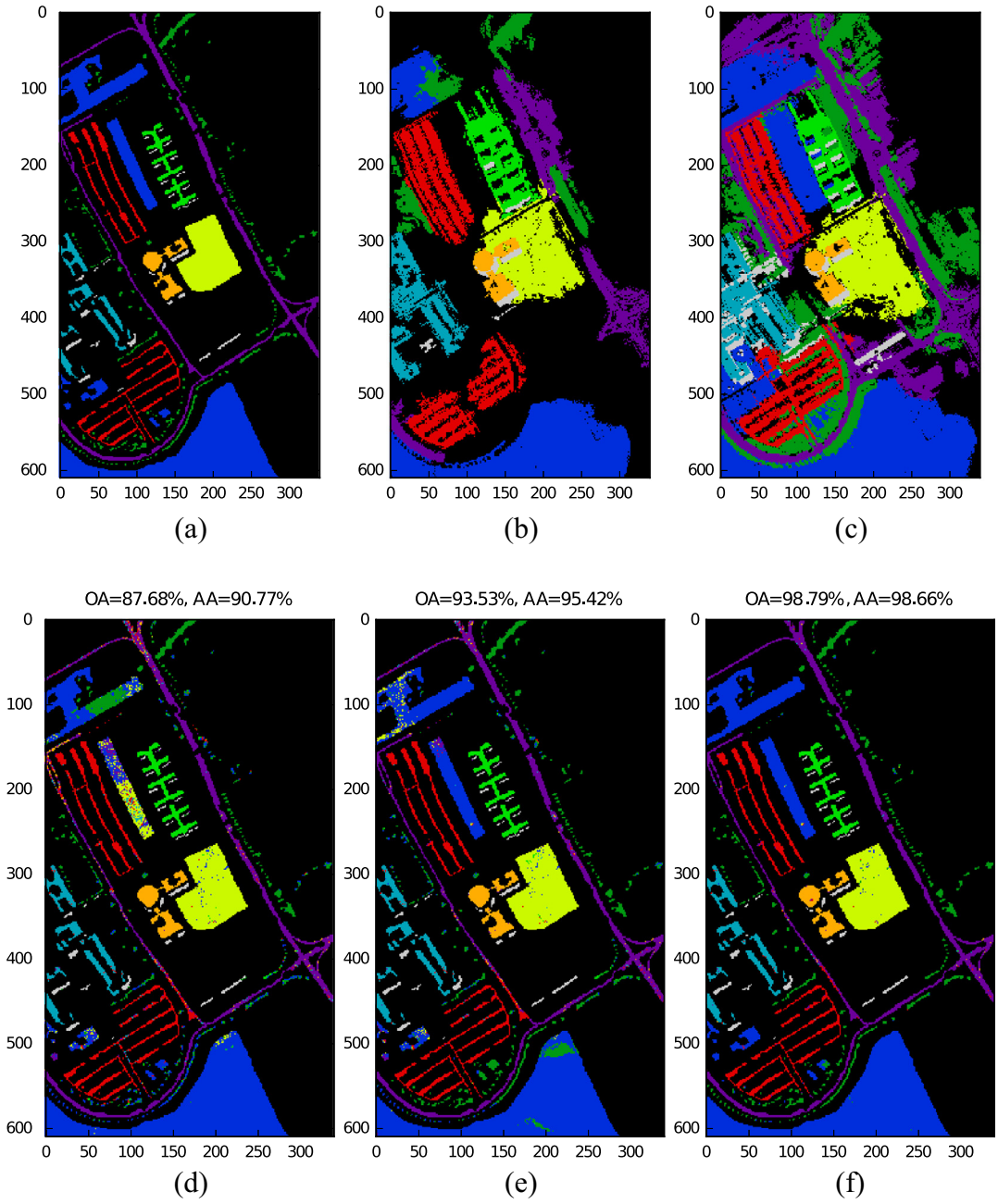

Fig. 9. Algorithm results for the University of Pavia dataset. Plot (a) presents the ground truth. Plots (b) and (c) are the extended training sets for 5 and 50 initial training samples. The bottom plots present an example of final classification results for 5 (d), 10 (e) and 50 (f) initial training samples.

processing.[5] Additionally, three supplementary experiments were performed. The first was designed to compare the effectiveness of different final spectral classifiers (SVM vs 1-NN). The second was focused on observation of the change in accuracy with subsequent iterations. The third performed statistical analysis of the results.

The summary of accuracy scores is presented in Table 1 in the form of the Overall Accuracy (OA), Average Accuracy (AA) and Cohen's Kappa (κ) (Foody, 2004). Presented values are the mean and the standard deviation across experimental runs. The results show good performance of the proposed method: in all tested cases, high recognition scores were achieved even with relatively small number of training data points. While having more training data increases accuracy, a stabilization point is reached around $s = 15$ to $s = 20$. Beyond this value, the increase in accuracy is small, and additional training samples contribute more to a decrease of error variation. Notably good results were achieved for images with regular class layout (Indian Pines, Salinas Valley and University of Pavia); even a small initial sample set is enough to provide accuracy over 90%. For those three scenarios, we were able to compare the accuracy scores with the reference results, presented in Tables 2–4 respectively. Reference values were provided in Wang et al. (2014) for the method created by its authors, called Spatial-Spectral Label Propagation based on the Support Vector Machines (SSLPSVM), and for the standard SVM, the Laplacian Support Vector Machine (LapSVM) proposed in Belkin et al. (2006), the Transductive Support Vector Machine (TSVM)

[5] The parameters were selected using the cross validation, with $C \in \{10^0, \ldots, 10^4\}, \gamma \in \{10^{-1}, \ldots, 10^2\}$ and MRF smoothness parameter $\beta \in \{10^0, \ldots, 10^3\}$.

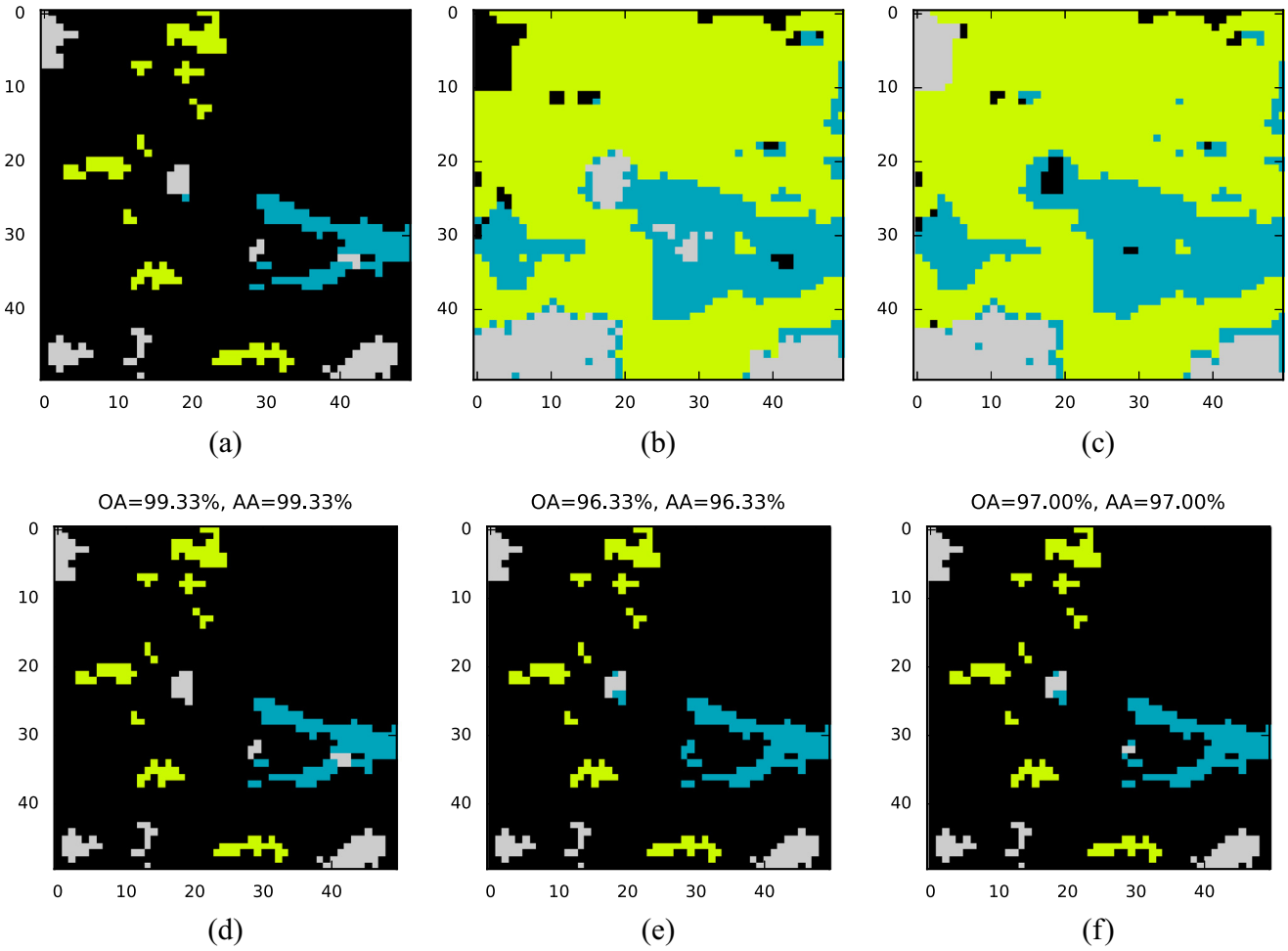

Fig. 10. Classification results for the Madonna dataset. Plot (a) presents the ground truth. Plots (b) and (c) are the extended training sets for 2 and 5 of the initial training samples. The bottom plots present an example of final classification results for 3 (d), 4 (e) and 5 of the (f) initial training samples.

explained in Joachims (1999) and the Spatial-Contextual Semi-Supervised Support Vector Machine (SCS^{3}VM) presented in Kuo et al. (2010). For the Indian Pines dataset we included a comparison to the spatial-spectral classification method, based on Generalised Composite Kernel, presented in Li et al. (2013). Additionally, the best reported results from Tan et al. (2015) for the MLR + KNN + SNI method (denoted MLR) and Tan et al. (2014) for S^{2}SVM are included in Tables 2 and 3. In the majority of scenarios, in particular for small datasets of 10 initial samples/classes or less, our approach outperforms other algorithms.

Two of the remaining scenarios (Modified Indian Pines, MIP and La Selva, LS) are particularly interesting. They were chosen specifically to pose a challenge for the P- and N-expert, respectively. The MIP dataset has high spatial diversity with classes scattered across the image to increase the number of inter-class borders within the ground truth. This lowers the effectiveness of the P-expert, as it increases the chance that pixel's neighbours belong to other classes. On the other hand, the LS dataset's classes have a high spectral diversity, which challenges the N-expert's classifier in the initial iterations. The results show that in both cases the other expert is able to compensate and achieve a high accuracy. The remaining dataset (Madonna) includes additional challenges with irregular class structure and a presence of nonlinear spectral mixtures. The comparison with baseline methods (SVM + MRF) is presented in Fig. 12. The PNGrow outperforms baseline methods for the MIP and LS scenarios, in some cases by as much as 20%.

In the third case, Madonna, global classifiers perform slightly better than the proposed approach. Quantitative evaluation of the differences show that they are located mostly on spatial class boundaries. We note that in those regions the estimated ground truth is less trustworthy, as it coincides with class spectral boundaries, which are created by a simple spectral distance assignment to the endmembers estimated by unmixing. The accuracy of the proposed method in this scenario is still very high, with $OA \approx 97\%$.

Graphical visualisations of the results are presented in Figs. 6–11. Each of them presents a ground truth, intermediate results of final iterations for two edge cases $s = 5$ and $s = 50$ (except La Selva $s = 5$ and $s = 25$ and Madonna $s = 3$ and $s = 5$), and output of the algorithm for three cases of initial training sample counts. The Figs. 7 and 11 also present the results of reference SVM + MRF methods. Several notable behaviours can be observed e.g.: cases when the area without a seed remains unclassified by the experts and is corrected by the final classifier (e.g. Fig. 7 (b) and (d)); when the spectral classifier is able to successfully supplement the experts even for a very small number of training samples (e.g. Fig. 9(b) and (c)); for spectrally challenging scenarios, additional training samples are needed (e.g. Fig. 11(b) and (c)).

Fig. 4 illustrates the impact of the training set extension on classification accuracy in consecutive iterations, while Fig. 5 shows the difference in the performance of a spectral classifier used in the final step of our algorithm, as described in Section 3.5. When the training set size is sufficiently high and for the datasets with sim-

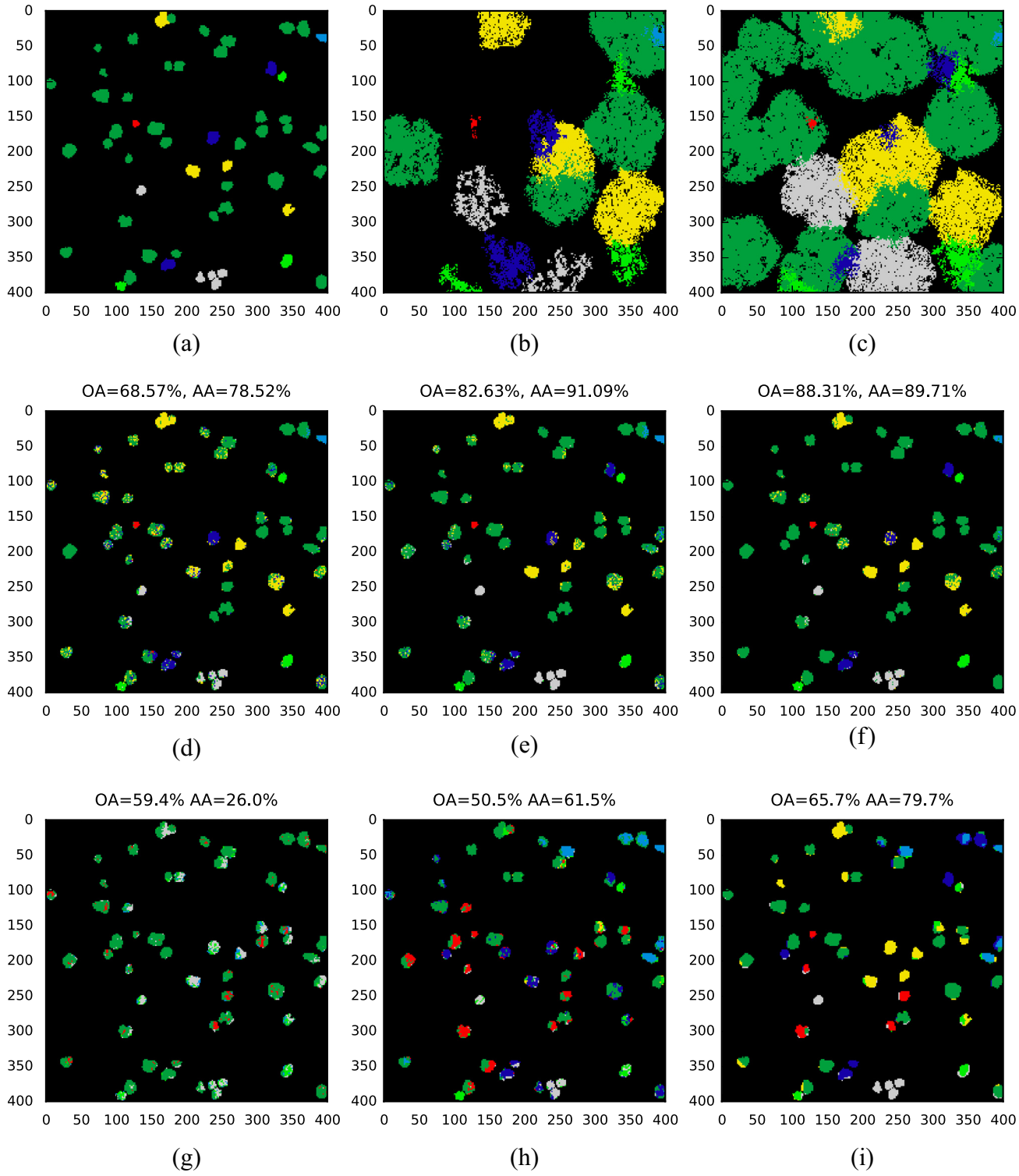

Fig. 11. Algorithm results for the La Selva Biological Station dataset. Plot (a) presents the ground truth. Plots (b) and (c) are the extended training sets for 5 and 25 initial training samples. The middle plots present an example of final classification results for 5 (d), 10 (e) and 25 (f) initial training samples. The bottom plots present reference classification results obtained using the SVM + MRF classifier and 5 (g), 10 (h) and 25 (i) training samples.

ple spatial structure (e.g. Indian Pines and Salinas Valley) the difference between KNN and SVM is minimal. This is because the majority of the image is labelled by experts. On the other hand, for a small number of samples and more challenging datasets (e.g. University of Pavia and La Selva Biological Station) SVM outperforms the simpler classifier, albeit at the cost of higher computational complexity and the need to estimate a number of parameters.

To verify the effectiveness of presented algorithm we conducted an additional statistical analysis of the results. For a given dataset D, let us define the advantage of the PNGrow algorithm over the best reference method on D, for s initial samples per class

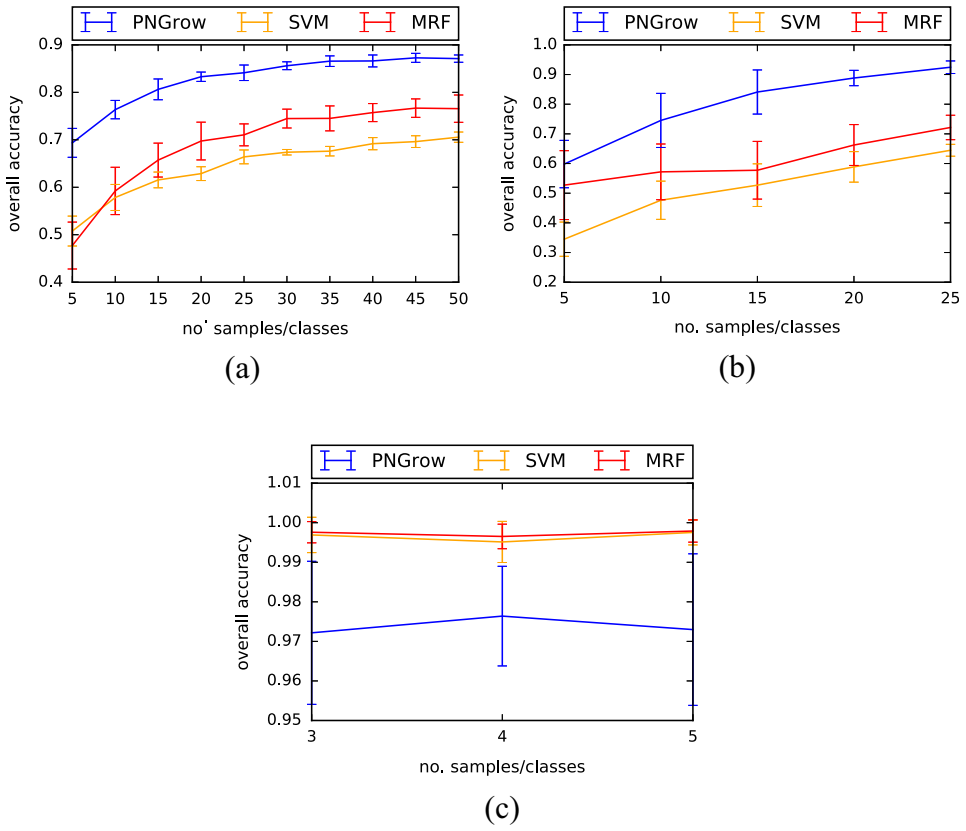

Fig. 12. Classification results for the proposed method and the reference classifiers (SVM and MRF) for the (a) Modified Indian Pines, (b) the La Silva Biological Station and (c) the Madonna dataset.

$$SM(D,s) = SM = OA_{PNGrow}(D,s) - OA_{\text{best}}(D,s) \tag{5}$$

representing how much the OA of proposed algorithm is higher than mean OA of the best reference method on dataset D with s initial seeds per class. We then considered the confidence interval of probability that $SM > 0$. For this, we performed $n = 100$ randomized selections of initial training sets from the Indian Pines dataset and computed OA of the PNGrow algorithm. The statistical inference is performed for significance levels $\alpha = 0.05, 0.10$ and 0.15. The confidence intervals of the proposed algorithm obtaining the OA higher than best reference method's average result ($SM > 0$) is presented in Table 5. We conclude that for selected cases of $s = 10, 20, 25$ the proposed method obtains a higher OA than the best reference method (MLR, GCK and GCK respectively) with 0.95 confidence. For $s = 5$ and $s = 15$, with the best reference being S^2SVM, we are unable to make the same claim for $1 - \alpha = 0.95$, however, the reverse hypothesis is also unsustainable. However, assuming $\alpha = 0.15$ we can conclude with confidence 0.85 that for $s = 15$ proposed algorithm also outperforms the best reference method (S^2SVM).

4.4. Discussion

The proposed algorithm was tested in a number scenarios, chosen to reflect potential applications. Its main advantage lies in the ability of experts to compensate each other errors, which is important as the real-life hyperspectral images provide a number of challenging situations. For example, imbalances in class lengths and the effect of 'oversmoothed' annotation, which can include different features under the same labelling (e.g. a road and a field sharing the same label) commonly appear in test datasets. While class imbalances are challenging for a global classifier, local classification inherent in the region-growing algorithms helps to overcome this problem due to the rapid extension of the training sets in the first iterations. On the other hand, the problem of unlucky sample selection (such as the training set being 'contaminated' by samples of outlier materials, e.g. an agricultural field with additional non-planted component) is compensated because the N-expert is designed to initially avoid rejecting new samples, and to gain confidence after a few subsequent iterations (as can be observed in Fig. 2). The most significant chal-

Table 5
Probabilities that the proposed method achieves higher OA than the best reference method. Results were calculated over the 100 experimental runs for the Indian Pines dataset and $s = 5, 10, 15, 20, 25$ initial training samples.

α	Training samples				
	5	10	15	20	25
0.05	$0.33 < p < 0.53$	$0.85 < p < 0.97$	$0.48 < p < 0.68$	$p = 1.0$	$p = 1.0$
0.10	$0.35 < p < 0.51$	$0.86 < p < 0.96$	$0.50 < p < 0.68$	$p = 1.0$	$p = 1.0$
0.15	$0.36 < p < 0.51$	$0.87 < p < 0.95$	$0.51 < p < 0.65$	$p = 1.0$	$p = 1.0$

lenge occurs when a class is scattered through the image and some of its patches are not seeded. For scenarios with a regular class structure e.g. the Indian Pines Dataset, this usually happens only for the small training sets. In such cases, the final labels in isolated areas depend on the spectral classifier used in the last step of the algorithm only. However, even in those scenarios when the experts perform incomplete labelling of the unknown pixels, the extended training set is sufficient to reliably train the final spectral classifier.

The experts work very well together for many common class shapes. This is especially visible for images of man-made agricultural or urban areas, which tend to have clear class boundaries and continuous area shapes. For example, classes representing long lines of roads or parallel, rectangular buildings (e.g. in the University of Pavia image) could be challenging for a region growing algorithm as pixels may have only a small number of neighbours from their own class. However, since such classes usually posses unique spectral structure, the N-expert is able to correct the process of spatial growing; this situation occurs commonly e.g. on the boundary between natural and man-made areas. Almost the opposite scenario can occur e.g. in mixed forest areas, when the spectral structure is diverse, but classes may be confined locally. For example, while canopies of a tropical rainforest can be successfully classified based on their spectral properties, as shown in Clark et al. (2005), their images contain significant spectral variation within populations. Nevertheless, individual tree canopies have a spatial consistency that benefits the P-expert.

The proposed method is also able to adapt to a non-uniform spectral class structure, e.g. when it changes with spatial dimension, as can be observed for the 'Grapes-untrained' class in the Salinas Valley image in Fig. 8 and when it contains nonlinear mixtures, as can be seen in the Madonna image in Fig. 10. These advantages of the co-training paradigm are consisted with results presented in Kalal et al. (2012), where a TLD has proven to work well, albeit for different classes of feature spaces.

Regarding the performance of the proposed algorithm; the method can be efficiently implemented as in every iteration the experts work independently, except for the step of combining the scores. The P-expert computes the neighbourhood of seeds in order to determine θ and the $S_p^c(i)$ for C classes, which can be performed in separate processes. The N-expert computes the spectral neighbourhood graph using the KD-tree to estimate the spectral distance. The processing steps can be done in parallel, with the most computationally intensive one being either the estimation of KDE score or spectral neighbourhood graph creation. In our experiments, the completion of one iterations takes values on the order of tens of seconds.

5. Conclusions

Our results show that the proposed approach based on co-training is very effective in semi-supervised hyperspectral classification. Our algorithm successfully employs the idea presented in the TLD framework. Classification results are comparable to or outperforming the state of art methods. The method is also robust in regards to parameter selection. The algorithm can be extended with a more complex spectral classifier e.g. SVM with dedicated kernels (Li et al., 2013) or an ensemble classifier (Cholewa and Głomb, 2016). Also, a reinitialization and addition of new seeds in unlabelled fragments of an image, as in Tilton et al. (2012) may eliminate problems that exist for small datasets. This opens up a possibility of using the proposed approach as an unsupervised segmentation algorithm based on P-N learning.

Acknowledgements

This work has been partially supported by the project 'Representation of dynamic 3D scenes using the Atomic Shapes Network model' financed by the Polish National Science Centre, decision DEC-2011/03/D/ST6/03753. The authors would like to thank Prof. Matthew L. Clark and Prof. Nicolas Dobigeon for making available datasets La Selva Biological Station and Madonna respectively. Authors would also like to thank anonymous reviewers for their insightful comments and their help in improving this manuscript.

References

Belkin, M., Niyogi, P., Sindhwani, V., 2006. Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. J. Machine Learning Res. 7, 2399–2434.
Biau, G., Devroye, L., 2015. Weighted k-nearest neighbor density estimates. In: Lectures on the Nearest Neighbor Method. Springer, pp. 43–51.
Bioucas-Dias, J.M., 2009. A variable splitting augmented Lagrangian approach to linear spectral unmixing. In: First Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, WHISPERS '09, pp. 1–4.
Bioucas-Dias, J.M., Plaza, A., Camps-Valls, G., Scheunders, P., Nasrabadi, N.M., Chanussot, J., 2013. Hyperspectral remote sensing data analysis and future challenges. IEEE Geosci. Remote Sensing Mag. 1 (2), 6–36.
Blum, A., Mitchell, T., 1998. Combining labeled and unlabeled data with co-training. In: Proceedings of the Eleventh Annual Conference on Computational Learning Theory. ACM, pp. 92–100.
Bruzzone, L., Chi, M., Marconcini, M., 2006. A novel transductive SVM for semisupervised classification of remote-sensing images. IEEE Trans. Geosci. Remote Sensing 44 (11), 3363–3373.
Campbell, J.B., Wynne, R.H., 2011. Introduction to Remote Sensing. The Guilford Press.
Camps-Valls, G., Marsheva, T.V.B., Zhou, D., 2007. Semi-supervised graph-based hyperspectral image classification. IEEE Trans. Geosci. Remote Sensing 45 (10), 3044–3054.
Chen, Z., Wang, B., 2016. Spectral-spatial classification based on affinity scoring for hyperspectral imagery. IEEE J. Select. Topics Appl. Earth Observ. Remote Sensing PP (99), 1–16.
Cheng, G., Han, J., Guo, L., Liu, Z., Bu, S., Ren, J., 2015. Effective and efficient midlevel visual elements-oriented land-use classification using vhr remote sensing images. IEEE Trans. Geosci. Remote Sensing 53 (8), 4238–4249.
Cheng, G., Han, J., Zhou, P., Guo, L., 2014. Multi-class geospatial object detection and geographic image classification based on collection of part detectors. ISPRS J. Photogr. Remote Sensing 98, 119–132.
Cheng, G., Zhu, F., Xiang, S., Wang, Y., Pan, C., 2016. Semisupervised hyperspectral image classification via discriminant analysis and robust regression. IEEE J. Select. Topics Appl. Earth Observ. Remote Sensing 9 (2), 595–608.
Cholewa, M., Głomb, P., 2016. Two stage SVM classification for hyperspectral data. In: Proceedings of the 5th International Conference on Pattern Recognition Application and Methods (ICPRAM), INSTICC, pp. 387–391.
Clark, M.L., Roberts, D.A., Clark, D.B., 2005. Hyperspectral discrimination of tropical rain forest tree species at leaf to crown scales. Remote Sensing Environ. 96 (3), 375–398.
de Morsier, F., Borgeaud, M., Gass, V., Thiran, J.P., Tuia, D., 2016. Kernel low-rank and sparse graph for unsupervised and semi-supervised classification of hyperspectral images. IEEE Trans. Geosci. Remote Sensing 54 (6), 3410–3420.
Dobigeon, N., Tourneret, J.-Y., Richard, C., Bermudez, J., McLaughlin, S., Hero, A.O., 2014. Nonlinear unmixing of hyperspectral images: models and algorithms. IEEE Signal Process. Mag. 31 (1), 82–94.
Dópido, I., Li, J., Marpu, P.R., Plaza, A., Dias, J.M.B., Benediktsson, J.A., 2013. Semisupervised self-learning for hyperspectral image classification. IEEE Trans. Geosci. Remote Sensing 51 (7), 4032–4044.
Duda, R.O., Hart, P.E., Stork, D.G., 2012. Pattern Classification. John Wiley & Sons.
Fang, L., Li, S., Kang, X., Benediktsson, J.A., 2015. Spectral-spatial classification of hyperspectral images with a superpixel-based discriminative sparse model. IEEE Trans. Geosci. Remote Sensing 53 (8), 4186–4201.
Fauvel, M., Tarabalka, Y., Benediktsson, J.A., Chanussot, J., Tilton, J.C., 2013. Advances in spectral-spatial classification of hyperspectral images. Proc. IEEE 101 (3), 652–675.
Foody, G.M., 2004. Thematic map comparison. Photogr. Eng. Remote Sensing 70 (5), 627–633.
Haralick, R.M., Shapiro, L.G., 1985. Image segmentation techniques. Comput. Vision Graphics Image Process. 29 (1), 100–132.
Harvey, A.C., 1990. Forecasting, Structural Time Series Models and the Kalman Filter. Cambridge University Press.
Hughes, G.P., 1968. On the mean accuracy of statistical pattern recognizers. IEEE Trans. Inform. Theory 14 (1), 55–63.
Joachims, T., 1999. Transductive inference for text classification using support vector machines. ICML, vol. 99, pp. 200–209.
Kalal, Z., Mikolajczyk, K., Matas, J., 2012. Tracking-learning-detection. IEEE Trans. Pattern Anal. Machine Intell. 34 (7), 1409–1422.

Khodadadzadeh, M., Li, J., Plaza, A., Ghassemian, H., Bioucas-Dias, J.M., Li, X., 2014. Spectral–spatial classification of hyperspectral data using local and global probabilities for mixed pixel characterization. IEEE Trans. Geosci. Remote Sensing 52 (10), 6298–6314.

Kuo, B.-C., Huang, C.-S., Hung, C.-C., Liu, Y.-L., Chen, I., et al., 2010. Spatial information based support vector machine for hyperspectral image classification. In: 2010 IEEE International Geoscience Remote Sensing Symposium (IGARSS). IEEE, pp. 832–835.

Landgrebe, D.A., 2005. Signal Theory Methods in Multispectral Remote Sensing, vol. 29. John Wiley & Sons.

Li, J., Bioucas-Dias, J.M., Plaza, A., 2012. Spectral–spatial hyperspectral image segmentation using subspace multinomial logistic regression and Markov random fields. IEEE Trans. Geosci. Remote Sensing 50 (3), 809–823.

Li, J., Reddy Marpu, P., Plaza, A., Bioucas-Dias, J.M., Atli Benediktsson, J., 2013. Generalized composite kernel framework for hyperspectral image classification. IEEE Trans. Geosci. Remote Sensing 51 (9), 4816–4829.

Melgani, F., Bruzzone, L., 2004. Classification of hyperspectral remote sensing images with support vector machines. IEEE Trans. Geosci. Remote Sensing 42 (8), 1778–1790.

Plaza, A., Benediktsson, J.A., Boardman, J.W., Brazile, J., Bruzzone, L., Camps-Valls, G., Chanussot, J., Fauvel, M., Gamba, P., Gualtieri, A., 2009. Recent advances in techniques for hyperspectral image processing. Remote Sensing Environ. 113, S110–S122.

Ratle, F., Camps-Valls, G., Weston, J., 2010. Semisupervised neural networks for efficient hyperspectral image classification. IEEE Trans. Geosci. Remote Sensing 48 (5), 2271–2282.

Sung, K.-K., Poggio, T., 1998. Example-based learning for view-based human face detection. IEEE Trans. Pattern Anal. Machine Intell. 20 (1), 39–51.

Tan, K., Hu, J., Li, J., Du, P., 2015. A novel semi-supervised hyperspectral image classification approach based on spatial neighborhood information and classifier combination. ISPRS J. Photogr. Remote Sensing 105, 19–29.

Tan, K., Li, E., Du, Q., Du, P., 2014. An efficient semi-supervised classification approach for hyperspectral imagery. ISPRS J. Photogr. Remote Sensing 97, 36–45.

Tarabalka, Y., Benediktsson, J.A., Chanussot, J., Tilton, J.C., 2010a. Multiple spectral–spatial classification approach for hyperspectral data. IEEE Trans. Geosci. Remote Sensing 48 (11), 4122–4132.

Tarabalka, Y., Chanussot, J., Benediktsson, J.A., 2010b. Segmentation and classification of hyperspectral images using minimum spanning forest grown from automatically selected markers. IEEE Trans. Syst. Man Cybernet. Part B: Cybernet. 40 (5), 1267–1279.

Tilton, J.C., Tarabalka, Y., Montesano, P.M., Gofman, E., 2012. Best merge region-growing segmentation with integrated nonadjacent region object aggregation. IEEE Trans. Geosci. Remote Sensing 50 (11), 4454–4467.

Tuia, D., Camps-Valls, G., 2009. Semisupervised remote sensing image classification with cluster kernels. IEEE Geosci. Remote Sensing Lett. 6 (2), 224–228.

Ugarriza, L.G., Saber, E., Vantaram, S.R., Amuso, V., Shaw, M., Bhaskar, R., 2009. Automatic image segmentation by dynamic region growth and multiresolution merging. IEEE Trans. Image Process. 18 (10), 2275–2288.

Wang, L., Hao, S., Wang, Q., Wang, Y., 2014. Semi-supervised classification for hyperspectral imagery based on spatial-spectral label propagation. ISPRS J. Photogr. Remote Sensing 97, 123–137.

Zhu, X., Goldberg, A.B., 2009. Introduction to semi-supervised learning. Synth. Lectures Artif. Intell. Machine Learning 3 (1), 1–130.