Ensemble Learning in Hyperspectral Image Classification: Towards Selecting a Favorable Bias-Variance Tradeoff

Andreas Merentitis, Christian Debes, and Roel Heremans AGT International, Hilpertstrasse 35, 64295, Darmstadt, Germany
Email: {amerentitis, cdebes, rheremans}@agtinternational.com

***Abstract*—Automated classification of hyperspectral images is a fast growing field with numerous applications in the areas of security and surveillance, agriculture, urban management, and environmental monitoring. While significant progress has been achieved in the various aspects of hyperspectral classification (e.g., feature extraction, feature selection, classification, and post-classification processing), the problem has not been addressed so far from a bias-variance decomposition point of view. In this work we introduce a consistent unified framework that jointly considers all steps in the hyperspectral image classification chain from a bias-variance decomposition point of perspective. Additionally, we show how state of the art techniques in feature extraction, ensemble-based classification, and post classification segmentation are related to the bias-variance tradeoff and how this relation can be used to improve classification accuracy. An important outcome of our analysis is that all the steps of the classification chain should be optimized jointly as this unified optimization can guide towards a more favorable bias-variance tradeoff. Experimental results of the proposed framework in the case of four hyperspectral datasets prove the effectiveness of our approach.**

***Index Terms*—Hyperspectral image, classification, ensemble methods, bagging, random forest, segmentation, bias-variance**

I. Introduction

Hyperspectral imaging (HSI) is an optical imaging technique operating in the visible and infrared light range. Every pixel of an acquired hyperspectral image can be seen as a spectral fingerprint that is unique to the materials in the respective spatial area. The ability to perform high-accuracy identification of materials in a scene based on their spectral fingerprints made HSI an important tool for various applications in security and surveillance [1], [2], [3], [4], agriculture [5], [6] and environmental monitoring [7], [8], [9] to name a few. More recently, HSI has also been applied in more industrial applications including nutrition analysis [10], [11] and waste recycling [12], [13].

Airborne HSI systems produce a large amount of data that to a large extent cannot even be visually inspected by humans. Automatic classification of the acquired images is thus of high practical importance and a significant amount of work has been focused in this field [14], [15], [16]. This holds for classification based on hyperspectral images solely, as well as in conjunction with other sensor modalities such as Synthetic Aperture Radar [17] or LiDAR [18], [19], [20]. Typically, classification of hyperspectral images consists of sequential steps such as pre-processing [21], [22], feature extraction [23], [24], feature selection [25], [26], [27], segmentation [28], [29], classification [30], [31] and post-processing [32], [33]. In each of those steps significant progress has been accomplished and several machine learning algorithms found their way into the hyperspectral community.

In the following, some representative works for advanced machine learning in HSI are outlined. Ham et al. [34] proposed a classifier that incorporates bagging of training samples and adaptive random subspace feature selection within a binary hierarchical classifier such that the number of features that is selected at each node of the tree is dependent on the quantity of associated training data. Tarabalka et al. [32] construct minimum spanning forests from region markers to enable hyperspectral image segmentation and classification. They consider the results of a pixel-wise classification to grow regions based on the class-conditional probabilities obtained from application of support vector machines (SVM). Fauvel et al. [16] proposed a classification scheme with support vector machines using the available spectral information and the extracted spatial information. Finally, a multiple classifier system is defined to produce relevant markers that are exploited to segment the hyperspectral image. Tuia et al. [35] introduced a semi-supervised SVM where the training was performed using two kernels - one for labeled and one for unlabeled data. Mura et al. [36] presented a technique for the classification of hyperspectral images based on independent component analysis (ICA) and extended morphological attribute profiles. The features extracted by the morphological processing where then classified with an SVM. However, despite the significant progress achieved in the previous and similar state of the art works, a holistic and systematic view on the whole classification chain in terms of bias-variance decomposition, one of the fundamental concepts of machine learning, is still missing.

The concept of bias-variance decomposition was introduced in machine learning initially for the case of mean squared error [37]. Later extended versions for "zero-one loss" (predictions are correct or false) were proposed by [38], [39], [40], [41], and [42]. This decomposition provides a theoretical background that explains why techniques such as regularization, or ensemble learning are particularly effective in a range of different machine learning applications.

In the context of machine learning, regularization is a process of incorporating additional information into the model,

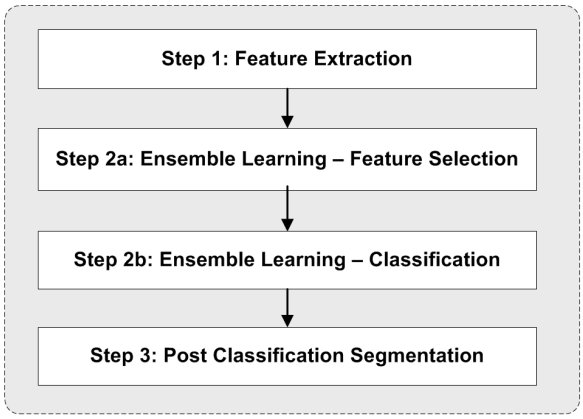

Fig. 1: *Flow diagram of hyperspectral image classification*

either in order to solve an ill-posed problem or to prevent overfitting. This additional information is most of the times some type of a penalty for complexity, such as restrictions for smoothness or bounds on the vector space norm. Therefore, regularization is targeting a significant reduction in the variance, with some small penalty in bias. More recently, a different mechanism has appeared that also allows for tradeoffs according to the bias-variance decomposition of the error: ensemble methods.

Ensemble learning techniques combine multiple models in an attempt to get closer to the unknown function one aims at approximating. More specifically, they construct a set of classifiers and consequently classify data points according to the (possibly weighted) vote of the independent predictions. As a supervised learning technique, an ensemble represents a single hypothesis, but one that is not necessarily contained within the space of the models used in the ensemble. One of the ensemble methods that have gained significant attention in recent years is the random forest algorithm, which is based on multiple classification tree instances [43].

In this work, we present a methodology for selecting a favorable bias-variance tradeoff for the classification of hyperspectral images. The proposed methodology considers all steps in the hyperspectral classification chain (Fig. 1) under the view of the bias-variance tradeoff. Having this view allows for a holistic and systematic way to assess steps like feature extraction and feature selection, choice of learning model (including ensemble learning algorithms such as the random forest), as well as application of post-processing algorithms. Each of these steps is critical and cannot be optimized without consideration of the other steps, since it can constrain the bias-variance tradeoff of later stages. Thus, good results are only feasible when a unified optimization that considers all the previous choices is applied.

Towards this goal, the primary contribution of this paper is the presentation of a unified framework for classification of hyperspectral imagery that considers the bias-variance tradeoff. In this direction, the paper demonstrates how modern machine learning algorithms, (i.e. ensemble methods) can be applied to hyperspectral image classification and how a consistent unified framework can improve the classification accuracy. Specifically, the following contributions are made:

- Discussion and application of ensemble methods for hyperspectral image classification
- Formulation of a feature selection scheme in a way that steers the later stages of our processing chain towards a favorable bias-variance tradeoff
- Formulation of a scheme to choose points that represent an optimal bias-variance tradeoff for the ensemble classifier
- Formulation of a Markov Random Field-based approach that aims at reducing post-classification variance, considering also the tradeoffs in previous steps

The rest of the paper is organized as follows. Section II presents briefly the bias-variance decomposition initially for the case of a regression and then the extension by Domingos et al. for the case of classification problems. Next, we investigate various feature extraction, processing and dimensionality reduction techniques from the scope of bias-variance decomposition in III. Ensemble learning methods are elaborated in Section IV. Moreover, their impact on the bias-variance tradeoff is discussed before a more thorough analysis for the case of the random forest algorithm. Section V discusses the application of Markov Random-Field-based techniques on the classification result in order to improve classification accuracy. An optimization scheme for automatically selecting the relevant parameters of the random field is presented. Various experimental results using the widely used datasets in hyperspectral imaging (Indian Pines, Pavia Center, Pavia University and Pavia Extended) are provided in Section VI where we also set a focus on discussing validation procedures and provide best practices. Finally, Section VII concludes the paper.

II. Bias-Variance Decomposition

As discussed earlier the concept of bias-variance decomposition provides a theoretical background that explains why techniques such as regularization or ensemble learning are effective. Before going into the details of the applicability of the decomposition in the steps of a hyperspectral image classification process depicted in Fig. 1 (feature extraction, learning model selection and configuration, post-classification segmentation) and the practical tradeoffs one has to consider in each step, a short introduction of the concept is required. We hereby follow the notation introduced by [41] and refer to [38], [39], [40], and [42] for more details or alternative definitions.

A. Bias-variance decomposition for regression

Let us assume the case of a regression model, where both the unknown true function u and our selected hypothesis f are continuous functions, while the loss function is the squared error. Let $\overline{f}$ be the average hypothesis (the mean of all hypotheses for different but finite training sets $\mathcal{D}$). We consider a specific test point of interest $\mathbf{x_o}$ that can be multi-dimensional according to the dimensionality of the feature space. In this point the value of our hypothesis f is y and

the actual observed value is $u_o(\mathbf{x_o}) = t + \varepsilon$ (where t is the value of the true function u and ε is normally distributed noise with zero mean and standard deviation σ). Then the Prediction Error (PE) in this point of interest $\mathbf{x_o}$ is defined as:

$$PE(\mathbf{x_o}) = \mathbb{E}\left\{(u_o(\mathbf{x}) - f(\mathbf{x}))^2 | \mathbf{x} = \mathbf{x_o}\right\} \tag{1}$$

and it can be proven easily (e.g., [37]) that the previous equation can be decomposed in the following constituents:

$$PE(\mathbf{x_o}) = \mathrm{B}^2(\mathbf{x_o}) + \mathrm{V}(\mathbf{x_o}) + \mathrm{N}(\mathbf{x_o}) \tag{2}$$

where $\mathrm{B}^2(\mathbf{x_o}) = (\overline{f}(\mathbf{x_o}) - u(\mathbf{x_o}))^2$ is the square of the bias, $\mathrm{V}(\mathbf{x_o}) = \mathbb{E}_{\mathcal{D}}\left[(f^{(\mathcal{D})}(\mathbf{x_o}) - \overline{f}(\mathbf{x_o}))^2\right]$ is the variance, and $\mathrm{N}(\mathbf{x_o}) = \mathbb{E}_{\varepsilon}\left[\varepsilon\right]^2$ is the noise on point $\mathbf{x_o}$. Therefore, we can consider the expectation of the Prediction Error with respect to $\mathbf{x}$ as:

$$\begin{aligned} \mathbb{E}_{\mathbf{x}}[PE(\mathbf{x})] = \mathbb{E}_{\mathbf{x}}\left[(\overline{f}(\mathbf{x}) - u(\mathbf{x}))^2\right] \\ + \mathbb{E}_{\mathcal{D},\mathbf{x}}\left[(f^{(\mathcal{D})}(\mathbf{x}) - \overline{f}(\mathbf{x}))^2\right] + \mathbb{E}_{\varepsilon,\mathbf{x}}\left[\varepsilon(\mathbf{x})\right]^2 \end{aligned} \tag{3}$$

where the noise term $\mathrm{N}(\mathbf{x}) = \mathbb{E}_{\varepsilon,\mathbf{x}}\left[\varepsilon(\mathbf{x})\right]^2$ expresses the lower bound on performance, the square of the bias term $\mathrm{B}^2(\mathbf{x}) = \mathbb{E}_{\mathbf{x}}\left[(\overline{f}(\mathbf{x}) - u(\mathbf{x}))^2\right]$ is the expected error due to model mismatch and the term $\mathrm{V}(\mathbf{x}) = \mathbb{E}_{\mathcal{D},\mathbf{x}}\left[(f^{(\mathcal{D})}(\mathbf{x}) - \overline{f}(\mathbf{x}))^2\right]$ is variation due to train sample properties and randomization, as well as randomness in the learning algorithm itself (e.g. neural net weight initialization [37]). The variance term has the superscript $\mathcal{D}$ to indicate its dependance on the training set.

Bias and variance are directly related with under- and over-fitting. Bias decreases and variance increases with respect to the complexity of the model; as more parameters are added to a model, its complexity rises and bias decreases monotonically, while variance becomes the main concern. For example, when additional polynomial terms are included to a linear regression, the resulting model has higher complexity and more capacity to explain the training set but it also becomes more susceptible to overfitting the particular training data, making generalization error prone. Since bias has a negative first-order derivative with respect to model complexity and variance has a positive one, there exists an intersection point where the predictive capability of the model is maximized. Finding this point analytically is not feasible, but in the case of parameterizable models with degrees of freedom cross-validation can be used to explore different tradeoffs [44] and select the one that maximizes the predictive capability.

B. Generalization of the bias-variance decomposition

According to the previous analysis, the bias-variance decomposition distinguishes between (1) the bias error, which is a systematic error component associated with the learning algorithm and the complexity of the hypotheses set, (2) the variance error, which is an error component associated with differences in the selected hypothesis for different training sets and (3) an error component associated with the inherent uncertainty in the domain. However, while this decomposition is easy and intuitive for regression functions and squared error, an one-to-one mapping in the case of multi-class classification problems with different loss functions is not straightforward and various formulations have been proposed [38], [39], [40], [41], and [42]. A generic formulation that offers several desirable theoretical properties was introduced in [41] and we adopt this formulation here.

For a given training set $\{(\mathbf{x}_1, u_o(\mathbf{x_1})), ..., (\mathbf{x}_n, u_o(\mathbf{x_n}))\}$, a learner produces a certain hypothesis f. Given a test point $\mathbf{x_o}$, this hypothesis generates a prediction $f(\mathbf{x_o}) = y$. Assuming again $u(\mathbf{x_o}) = t$ is the true value of the predicted variable for the test point $\mathbf{x_o}$, then a loss function $L(t, y)$ measures the cost of predicting y when the true value is t. Commonly used loss functions are squared loss $L(t, y) = (t - y)^2$, absolute loss $L(t, y) = |t - y|$, and zero-one loss $L(t, y) = 0$ if $y = t$, $L(t, y) = 1$ otherwise. The first two are broadly used in regression while the third one is the default option for classification problems. In the context of a given loss function the goal of learning can be phrased as producing a hypothesis with the smallest possible loss, meaning that the chosen hypothesis minimizes the average $L(t, y)$ over all points, with each point weighted according to its probability.

The optimal prediction y_* for a point $\mathbf{x_o}$ is the prediction that minimizes $\mathbb{E}_t[L(t, y_*)]$, where the subscript t indicates that the expectation is taken with respect to all possible values of t, weighted according to their probabilities given $\mathbf{x}$. The optimal hypothesis is the one for which $f(\mathbf{x}) = \mathrm{y}_*$ for every $\mathbf{x}$ and even this hypothesis will have non-zero loss. In the case of zero-one loss function, the optimal hypothesis is the Bayes classifier, and its loss is the Bayes rate [45]. Since the same learner generates different models for different training sets, $L(t, y)$ is a function of the training set. This dependency can be alleviated by averaging over training sets. Let $\mathbf{D}$ be a set of training sets. In this case the quantity of interest is the expected loss $\mathbb{E}_{\mathbf{D},t}[L(t, y)]$, where the expectation is taken with respect to t and the training sets in $\mathbf{D}$ (i.e., with respect to t and the predictions $y = f(\mathbf{x})$ produced for $\mathbf{x}$ by applying the learner to each training set in $\mathbf{D}$). Having this formulation in place, it is possible to define the main prediction for a loss function L and set of training sets $\mathbf{D}$ as:

$$\mathrm{y}_m^{L,\mathbf{D}} = \arg\min_{y'} \mathbb{E}_{\mathbf{D}}[L(y, y')] \tag{4}$$

Therefore the main prediction is the value y' that has the minimum average loss relative to all the predictions. It can be easily derived from this definition that in the case of squared loss function the main prediction is the mean of the predictions, in the case of absolute loss it is the median, and in the case of zero-one loss it is the mode (the prediction with the highest frequency) [41]. Proceeding further in this path, it is possible to define the bias of a learner on a point $\mathbf{x_o}$ as $\mathrm{B}(\mathbf{x_o}) = L(y_*, y_m)$. Therefore, the bias is the loss incurred by the main prediction with respect to the optimal prediction. Similarly, the variance of a learner on a point $\mathbf{x_o}$ can be defined as $\mathrm{V}(\mathbf{x_o}) = \mathbb{E}_D[L(y_m, y)]$. Therefore, the variance is the average loss incurred by predictions with respect to the main prediction. Finally, the noise at point $\mathbf{x_o}$ is $\mathrm{N}(\mathbf{x_o}) = \mathbb{E}_{\mathbf{D}}[L(t, y_*)]$. Therefore, noise is the component of the loss that cannot be avoided, and is incurred independently of the learning algorithm. It is also important to note that bias

and variance can be averaged over all points to produce the average bias $\mathbb{E}_{\mathbf{x}}[\mathrm{B}(\mathbf{x})]$ and the average variance $\mathbb{E}_{\mathbf{x}}[\mathrm{V}(\mathbf{x})]$. Building on the previous definitions, it was shown in [41] that, considering a test point $\mathbf{x_o}$ for which the true prediction is t, a learner that predicts y given a training set in $\mathbf{D}$, and an arbitrary loss function L, then the following decomposition of $\mathbb{E}_{\mathbf{D},t}[L(t,y)]$ holds:

$$\mathbb{E}_{\mathbf{D},t}[L(t,y)] = c_1\mathbb{E}_t[L(t,y_*)]+[L(y_*,y_m)]+c_2\mathbb{E}_{\mathbf{D}}[L(y_m,y)] \quad (5)$$

or,

$$\mathbb{E}_{\mathbf{D},t}[L(t,y)] = c_1 N(\mathbf{x}) + B(\mathbf{x}) + c_2 V(\mathbf{x}) \quad (6)$$

There are several important observations stemming from the analysis of [41]. First of all it can be easily seen that this decomposition falls back to the standard one for squared loss with $c_1 = c_2 = 1$, considering that for squared loss $y_* = \mathbb{E}_t[t]$ and $y_m = \mathbb{E}_{\mathbf{D}}[y]$, i.e. the prediction of the average hypothesis [37]. However, here we focus only on the part that is applicable for classification problems and consequently the zero-one loss function. The most important observation is that specifically for zero-one loss, variance can have a subtractive effect and this is derived from a self-consistent definition of bias and variance for zero-one and squared loss, even if the variance itself remains positive [41]. The fact that variance is additive in unbiased examples but subtractive in biased ones has significant implications: if a learner is biased on a given test point, increasing variance can decrease loss. This behavior is fundamentally different from that of squared loss, but is obtained with the same definitions of bias and variance, purely as a result of the different properties of zero-one loss. In effect, when zero-one loss is the evaluation criterion, there is a much higher tolerance for variance than if the bias-variance decomposition was strictly additive, because the increase in average loss caused by variance on unbiased examples is (partly) offset by its decrease on biased ones. The average loss over all points is the sum of noise, the average bias and what Domingos et al. define as the net variance, $\mathbb{E}_{\mathbf{x}}[c_2 V(x)]$:

$$\mathbb{E}_{\mathbf{D},t,\mathbf{x}}[L(t,y)] = \mathbb{E}_{\mathbf{x}}[c_1 N(\mathbf{x})] + \mathbb{E}_{\mathbf{x}}[B(\mathbf{x})] + \mathbb{E}_{\mathbf{x}}[c_2 V(\mathbf{x})] \quad (7)$$

derived by averaging Equation (5) over all test points $\mathbf{x}$, with c_2 positive for unbiased points and negative for biased ones. Equation (5) is also valid in the case of zero-one loss function for multiclass problems, with

$$c_1 = P_{\mathbf{D}}(y = y_*) - P_{\mathbf{D}}(y \neq y_*)P_t(y = t|y_* \neq t) \quad (8)$$

and

$$c_2 = 1 \text{ if } y_m = y_* \quad (9)$$

or

$$c_2 = -P_{\mathbf{D}}(y = y_*|y \neq y_m) \quad (10)$$

otherwise. This formulation shows that in multiclass problems not all variance on biased points contributes to reducing loss. Considering all training sets for which $y \neq y_m$, only some have $y = y_*$, and it is only in these points that loss is reduced. Therefore, the negative effect of variance is exacerbated as the number of classes increases and this can be an important factor in the selection of models or model parameters as will be shown in the later sections.

III. Feature Extraction and Feature Selection

While the bias-variance decomposition is very commonly used to explore tradeoffs in the learning model, feature extraction methods (including transformations and dimensionality reduction techniques) as well as feature selection techniques (e.g., [46]) are already setting the scene by constraining or guiding the classifier towards specific regions of the curve. In this view feature extraction can be used to avoid bias error caused by limitations in the representation language of a hypothesis. More specifically, it can reduce bias error in the case of adding features that allow for models which initially could not be part of the hypothesis set, or by changing the priority with which the hypothesis set is traversed (sometimes mentioned as search bias [47] which is relevant in case one does not consider the entire hypothesis set).

On the other hand, adding a large number of features can result to the "curse of dimensionality" [48], which is encountered in case the volume of the feature space increases so much that the data becomes sparse in the feature space. This sparsity makes it difficult to achieve statistical significance for many learning methods because the definitions for density and distance that many classification methods employ at the learning phase become less useful. Dimensionality reduction methods [49] are the most common and general ways to alleviate this problem. The goal of such methods is to keep as much of the information as possible in a reduced number of features. In case the dimensionality reduction method allows for regularization, further reductions in variance are possible which is for example the case with the Minimum Noise Fraction transformation [50], [51]. In this section, we present a scheme for feature extraction in hyperspectral images that steers the later stages towards favorable bias-variance tradeoffs. The proposed scheme is comprised of four different methods:

- Minimum Noise Fraction Transformation
- Blind unmixing and derivation of abundance maps
- Nonparametric Weighted Feature Extraction
- Synthetic features

as depicted in Fig. 2 and detailed in the remainder of this section.

A. Minimum Noise Fraction Transformation

The first step of our feature extraction algorithm applies the regularized version of the kernel MNF transformation, where the goal is to maximize the Rayleigh quotient [52], [53] formulated as:

$$\frac{1}{NF} = \frac{b^T K^2 b}{b^T[(1-\lambda)K_N K_N^T + \lambda K]b} \quad (11)$$

i.e., the inverse noise fraction, NF, which can be defined as the variance of the noise with respect to a noise model, divided by the total variance. In Equation (11), b is the band to be transformed, λ is the regularization parameter, while K and K_N are kernelized and centered versions of the data and the noise components, respectively [52], [53]. Therefore, this transformation corresponds to maximizing the signal-to-noise ratio, $SNR = 1/NR - 1$, and as has been shown in

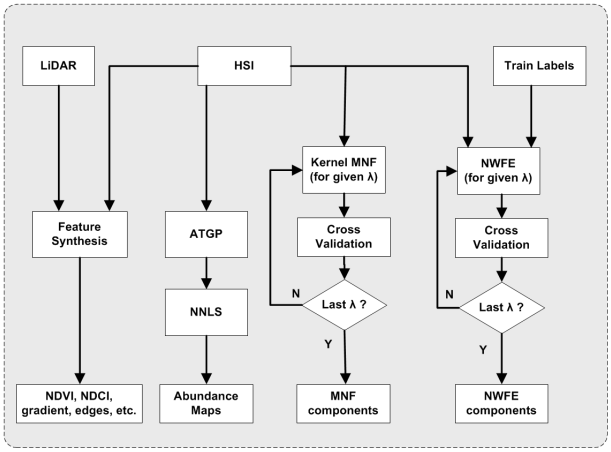

Fig. 2: *Feature extraction algorithm*

many application areas, an improvement in SNR shifts the entire plane towards more favorable bias-variance tradeoffs (e.g., [54]). Cross-validation is deployed in order to determine the optimal value of the regularization parameters so that the effect of noise on the selected features is minimized.

B. Blind unmixing and derivation of abundance maps

As a next step, our feature extraction algorithm applies blind unmixing based on Automatic Target Generation Procedure (ATGP) [2] to extract the basic spectra of the materials that are present in the HSI in a way that is resilient to distortions (e.g. from the presence of shadows). The Non-Negative Least-Squares (NNLS) method [55] was used in order to extract abundance maps. It is important to underline at this point that the abundance maps are very effective features, especially in hyperspectral imagery where many different signatures can be detected (e.g., in an urban environment). This is an important observation, given that in literature the use of abundance maps as features is mostly proposed as a dimensionality reduction method (e.g., [56]) and not as a way to enhance classification accuracy (apart from the improvement achieved by mitigating the Hughes phenomenon [48]). The proposed approach was found to have lower variance error than the use of a generic feature extraction and dimensionality reduction methods such as PCA (for the same number of features), especially in difficult parts of the hyperspectral image. The explanation is that in such cases feature extraction does not relax representational bias but it lowers the variance error of the method, in particular with respect to the feature selection step applied later.

C. Nonparametric Weighted Feature Extraction

Information from the original spectral bands is also used through separate application of the Nonparametric Weighted Feature Extraction (NWFE) transformation [57]. NWFE targets not so much to suppress noise but rather to do the dimensionality reduction in a way that maintains information needed for class discrimination. In fact it has been shown to often be more effective then Linear Discriminant Analysis (LDA) (e.g. [58]) in this particular problem. Again, cross-validation is deployed in order to determine the optimal value of the regularization parameter.

D. Synthetic features

The final step of our algorithm is the extraction of synthetic features. Particularly for hyperspectral image classification, bias reduction can be achieved by the generation of additional synthetic features (through non-linear transformation of specific bands) that are very good for discriminating types of material such as vegetation (e.g., Normalized Difference Vegetation Index - NDVI), water (e.g., Normalized Difference Cloud Index - NDCI), etc. Finally, in case LiDAR is also part of the dataset, synthetic features can include gradient or edge detection features. It should be noted that even if this is not the case, edge features can also be calculated in a segmentation map derived from purely hyperspectral information (e.g., using Watershed segmentation based on the first MNF component), with some reduction in performance, depending on the type of the datasetcompared, compared to having the actual LiDAR features.

E. Feature selection

Removing features that are irrelevant for the learner will not change the bias error [59]. As far as the learner is concerned, irrelevant features are treated as noise. Feature selection is generally aimed at variance reduction: fewer parameters need to be estimated, while the amount of relevant information that is removed is minimized. Removing relevant features may lead to an increase of intrinsic bias. However, features such as the MNF components and the extracted abundance maps of the hyperspectral image are expected to contain significant redundancy, therefore the risk of overfitting the relation between individual features and the target is high and eliminating some irrelevant features is likely to reduce the variance error.

It is important to note also that feature selection decisions about composite features, for example when one feature represents the information contained in several others, are less sensitive to sampling variance. Thus, extraction of synthetic features can also result in reduced variance error. Particularly for hyperspectral image classification, this applies on the calculation of synthetic features such as the NDVI, NDCI, and the synthetic LiDAR features. Such features are already the outcome of domain specific non-linear transformations and therefore are excluded from application of generic techniques such as MNF, NWFE, etc. In this work, feature selection is performed by the learning algorithm itself (random forest) therefore more details on the practical application of feature selection are given in the following section.

IV. Hyperspectral Image Classification Using Ensemble Methods: A Bias-Variance Approach

Formally, an ensemble is a technique for combining numerous weak learners in an attempt to produce a strong learner.

An ensemble is also a supervised learning method, since it has the capacity to be trained and then used to perform predictions. As such, the ensemble also represents a single hypothesis in the solution space. However, this hypothesis is not necessarily contained within the space of the models which were used to construct the ensemble. Therefore ensembles typically have more flexibility in the functions they can represent, which can result in a reduction of model bias [60]. Considering the typical bias-variance decomposition and the bias-variance tradeoff, an increase in model complexity is often associated with an increase in variance, since the more complex model is potentially more prone to overfitting the training data. This effect is encountered to a different extent in various ensemble methods, but some of them are robust since they are specifically designed to avoid it (e.g. bagging).

In terms of improvement, it can be theoretically proved that if an infinite number of models are available, each of them is better than simple guessing and their errors are statistically independent, then it is possible to construct an ensemble model with arbitrary accuracy [61]. These conditions never hold of course in practical problems, but it can be shown empirically that having a set of models which when used independently provide good predictions and their diversity (i.e. disagreement) is high is a good basis for ensemble methods [62]. Thus, many ensemble methods are often actively seeking to promote diversity among the models they combine.

A. Ensemble methods and the bias-variance analysis

Ensemble methods can be categorized with respect to the way they are achieving diversity and the way they are combining the outcomes of the weak learners. According to this categorization, some of the ensemble methods are applying multiple instances of the same model and achieve diversity by modifying the training set of each classifier (i.e., through resampling [63]), the features or both [64]. Such methods typically employ a simple (weighted) voting scheme where each instance casts a vote and the outcome of the ensemble is the class with the highest number of votes (e.g., bagging [65]). Weighted voting typically uses the classifiers confidence in its prediction (quantified as the estimated probability of the predicted class) or the error estimates of the classifier (e.g., boosting [66]).

A lot of effort has been focused in explaining formally why ensemble methods work so well. One of the main concepts [65] used to explain why the bagging ensemble method reduces zero-one loss was that of an order-correct learner. A learner is order-correct on a point $\mathbf{x_o}$ if and only if $\forall_{y \neq y*} P_{\mathbf{D}}(y) < P_{\mathbf{D}}(y_*)$. Breiman [65] showed that bagging transforms an order-correct learner into a nearly optimal one. We note that order-correctness and bias are closely related: a learner is order-correct on a point $\mathbf{x_o}$ if and only if $B(\mathbf{x_o}) = 0$ in the case of zero-one loss. The proof can be derived directly from the definitions, considering that y_m for zero-one loss is the most frequent prediction. Schapire et al. [67] proposed an explanation for why the boosting ensemble method works in terms of the notion of margin. For algorithms like bagging and boosting, which generate multiple hypotheses by applying the same learner to multiple training sets, their definition of margin M on a point $\mathbf{x_o}$ for a two class problem can be expressed as follows:

$$M(\mathbf{x_o}) = P_{\mathbf{D}}(y = t) - P_{\mathbf{D}}(y \neq t) \tag{12}$$

for which a positive margin indicates a correct classification by the ensemble, and a negative one an error. More recently [41] it was proven that the notion of margin is closely related to the bias-variance decomposition and specifically, the margin of a learner on a point $\mathbf{x_o}$ can be formulated in terms of its zero-one bias and variance as:

$$M(\mathbf{x_o}) = \pm[2B(\mathbf{x_o}) - 1][2V(\mathbf{x_o}) - 1] \tag{13}$$

with the positive sign applicable if $y_* = t$ and the negative sign applicable otherwise. Therefore the two main formal explanations of why ensemble methods such as bagging or boosting work are closely related.

B. The random forest algorithm

The random forest algorithm that was introduced by Breiman [43] utilizes both bagging and random attribute subset selection for achieving diversity between the weak learners. Breiman's innovations were influenced by previous works such as the random subspace method of [68], as well as the random split selection algorithm of [69]. As indicated by various empirical studies (e.g., [43], [70], [71], [72], [73], random forests have emerged as serious contesters to state-of-the-art methods such as boosting [74] and Support Vector Machines [75]. Some of their key advantages include reasonable computational cost, inherent support of parallelism, easiness to implement, highly accurate predictions and ability to handle a very large number of input variables without overfitting. In fact, they are considered to be one of the most accurate general-purpose learning techniques available. For more details we refer to the survey by [72] which is an excellent introduction to the method, offering also useful practical guidelines.

Table I summarizes the random forest algorithm for classification of hyperspectral images. It is easy to show that this approach is equivalent to bagging when $mtry$ is equal to the number of features. In general, the selection of $mtry$ is related to the bias-variance tradeoff, with smaller $mtry$ values favoring greater reduction in variance at the cost of some additional bias. Breiman suggests three possible values for $mtry$: $1/2\sqrt{p}$, $\sqrt{p}$, and $2\sqrt{p}$, where p is the total number or predictor variables.

Although the algorithm of the random forest appears simple, it involves various different driving forces which make it difficult to analyse in a formal mathematical context. In fact as indicated in [5], its mathematical properties remain to some extent unknown and most theoretical studies have concentrated on isolated parts or simplified versions of the algorithm. Important attempts in this direction are done by [77], which showed the relation between random forests and adaptive nearest neighbor methods (further elaborated by [78]); [79], which explored the consistency of random forests in the context of conditional quantile prediction; and [80], which introduced consistency theorems for various simplified

TABLE I: The Random Forest Algorithm for hyperspectral image classification

Step 0.	*Collecting data*. Conduct the experiment and collect the hyperspectral image consisting of m bands, $b_1, ..., b_m$
Step 1.	*Extracting features*. Perform feature extraction (e.g., according to the algorithm in Section III)
Step 2.	*Selecting parameters*. Select the parameters of the random forest algorithm, mainly the number of trees T and the value of the feature set splitting variable $mtry$
Step 3.	*Resampling*. Sample the training data (random sampling with replacement) to create T different subsets of the data, each of size N, with N approximately 66% of the complete training set [43].
Step 4.	*Training the decision trees*. • For a given tree node select a subset of $mtry$ predictor variables at random from the set of all the predictor variables. • The predictor variable that provides the best split, according to the selected objective function (usually based on an impurity measure, such as the information gain or the Gini gain [76]), is used to do a binary split on that node. • At the next node, select a different set of $mtry$ variables at random from all predictor variables and repeat.
Step 5.	*Classifying*. After the training is done and the algorithm operates in classification mode, when a new input is entered it is run down all of the trees. If the range of valid predictions is $\mathcal{C} = \{1, ..., C\}$ where C is the total number of classes, then the estimated probability of predicting class $y \in \mathcal{C}$ for a given point $\mathbf{x_o}$ is: $$p(y|\mathbf{x_o}) = \frac{1}{T}\sum_{t=1}^{T} p_t(y|\mathbf{x_o})$$ with $p_t(y|\mathbf{x_o})$ being the estimated density of class labels on the leaf of the tth tree [43].

versions of random forests and other randomized ensemble predictors. Particularly in the case of hyperspectral image classification, it is possible to relate conceptually some of the previous findings with the selection of the random forest parameters, especially the number of trees T and the value of the feature set splitting variable $mtry$, given the properties of specific hyperspectral datasets. This will be further elaborated in Section VI.

V. Post-Classification Using Markov Random Fields

This section presents a novel post-processing scheme to improve classification accuracy. Specifically, it is considering the class-conditional probabilities $p(y_i = c)$ with $c \in \mathcal{C}$ obtained by classification methods such as the random forest presented in the previous section. The proposed scheme is a random-field based approach that reassigns class labels based on their spatial context and classification uncertainty. We will demonstrate how the proposed scheme can be configured in an optimal fashion using the data at hand.

A. MRF modeling of the classification outcome

Assume the following inverse problem describing the relationship between the true label field $\mathbf{g}$ and the classification outcome $\mathbf{y}$

$$\mathbf{y} = \mathbf{H}\,\mathbf{g} + \omega \tag{14}$$

where $\mathbf{H} \in \mathbb{R}^{m \times m}$ is a known mixing matrix which in the simplest case is the identity matrix and ω is an additive noise term with unspecified distribution. Consider a maximum a posteriori (MAP) estimator to estimate $\mathbf{g}$ from $\mathbf{y}$ as

$$\hat{\mathbf{g}} = \arg\max_{\mathbf{g}} \{p(\mathbf{g} \,|\, \mathbf{y})\} \tag{15}$$

which using Bayes' rule can be written as

$$\hat{\mathbf{g}} = \arg\max_{\mathbf{g}} \{p(\mathbf{g}) p(\mathbf{y} \,|\, \mathbf{g})\} \tag{16}$$

We propose modeling $p(\mathbf{g})$ by a Markov random field, i.e. [81]

$$p(\mathbf{g}) = \frac{1}{Z(\varrho)} \exp\left(-\varrho \left\|\mathbf{L}\,\mathbf{g}\right\|_2^2\right) \tag{17}$$

where $\mathbf{L}$ is a discretized differential operator which directs towards a smooth solution, ϱ is the attraction parameter and $\frac{1}{Z(\varrho)}$ the partitioning function [81].

A popular assumption would be the assumption of Gaussianity for $p(\mathbf{y} \,|\, \mathbf{g})$ which would be valid if ω in Equation (14) was Gaussian noise. In this case the optimization problem expands to

$$\hat{\mathbf{g}} = \arg\max_{\mathbf{g}} \left\{ \exp\left(-\frac{\left\|\mathbf{H}\,\mathbf{g} - \mathbf{y}\right\|_2^2}{2\sigma_\omega^2}\right) \cdot \exp\left(-\varrho \left\|\mathbf{L}\,\mathbf{g}\right\|_2^2\right) \right\} \tag{18}$$

or equivavently

$$\hat{\mathbf{g}} = \arg\min_{\mathbf{g}} \left\{ \left\|\mathbf{H}\,\mathbf{g} - \mathbf{y}\right\|_2^2 + \eta \left\|\mathbf{L}\,\mathbf{g}\right\|_2^2 \right\} \tag{19}$$

with $\eta = \frac{\varrho}{2\sigma_\omega^2}$ and σ_ω^2 being the noise variance.

Note that, however, as the probability density function of the additive noise term ω is unknown, $p(\mathbf{y} \,|\, \mathbf{g})$ can not be derived. We propose instead to approximate $p(\mathbf{y} \,|\, \mathbf{g})$ by the class-conditional distribution $p(y_i = c)$ obtained from the classifier. This implicitly assumes that the error analysis in the classifier matches the actual conditional distribution of $\mathbf{y}$ given the true label field $\mathbf{g}$.

Performing this approximation and inserting $p(\mathbf{g})$ from Equation (17) yields the following optimization problem

$$\hat{g}_i = \arg\max_{g_i} \left\{ p(y_i = g_i) \frac{1}{Z(\varrho)} \exp\left(\varrho \#\{y_j \in \mathcal{N}_{y_i} | y_j = y_j\}\right) \right\} \tag{20}$$

where $\#$ denotes the cardinality of a set and $\mathcal{N}_{y_i}$ is the neighbourhood of y_i. Here, without loss of generality, an Ising model was assumed [82] where the smoothing term is formulated by counting the number of pixels with the same label as y_i.

The optimization problem in Equation (20) can efficiently be solved using the iterated conditional modes (ICM) algorithm [83], [82]. The ICM can be considered as a Gibbs sampler at zero temperature, i.e. in every iteration the label $\hat{g}_i$ is updated by the mode of the posterior distribution $p(g_i|y_i)$ which allows fast convergence to a (local) maximum of $p(\mathbf{g} \,|\, \mathbf{y})$. The ICM has successfully been applied in the past in various remote sensing applications [84], [85], including hyperspectral imaging.

B. Optimality of the attraction parameter

One drawback of the ICM algorithm for solving Equation (20) is the necessity to determine the attraction parameter ϱ. This is crucial as too small values of ϱ will show no improvement of classification accuracy while too large values of ϱ will produce oversmoothing. One way to determine an optimal value for ϱ is to consider approaches like the iterated conditional expectation (ICE). Here, ϱ is determined based on a realization of a random field. This is not feasible in hyperspectral image classification as the training set typically consists of very few pixels that are not spatially connected.

We consider a different approach here which determines an optimal value for ϱ from the data at hand. First, we note that instead of following a MAP approach to solve the inverse problem in Equation (14) an alternative way is to minimize the Tikhonov functional [86], [87] which is given as

$$\mathcal{I}_\lambda(\mathbf{g}) = \|\mathbf{H}\,\mathbf{g} - \mathbf{y}\|_2^2 + \gamma\,\|\mathbf{L}\,\mathbf{g}\|_2^2 \tag{21}$$

where it is assumed that $\ker(\mathbf{H}) \cap \ker(\mathbf{L}) = \{\mathbf{0}\}$ which ensures a unique solution for $\hat{\mathbf{g}}$. Note that Equation (21) corresponds to a very similar problem formulation as Equation (20), the differences being that the Tikhonov functional is typically considered for minimizing the squared error (whereas in Equation (20) the data fidelity term remains undefined) and γ plays the role of a regularization parameter as opposed to ϱ which represents a weight for the clique potentials in a Markov random field.

We adopt the quasi-optimality criterion from Tikhonov [88], [89] which is used to determine the optimal regularization parameter γ as

$$\hat{\varrho} = \arg\min_{\varrho} \left\| \varrho \frac{\partial \hat{\mathbf{g}}_\varrho}{\partial \varrho} \right\|_2^2 \tag{22}$$

where $\hat{\mathbf{g}}_\varrho$ denotes the estimate for $\mathbf{g}$ as a function of ϱ. Note that Equation (22) allows automatic selection of the attraction parameter with the data at hand. The optimization problem in Equation (22) can be solved by e.g. gradient-based methods.

C. Extension to multi-class problems

Note that in both approaches - MAP estimation with a Markov Random Field and Tikhonov regularization - the regularization or attraction parameter is typically a scalar. In our case, as $g_i \in \{1, ..., C\}$ with typically $C \gg 2$ a class-independent parameter can yield limited performance. In practical hyperspectral imaging applications there are classes that are likely to coexist in one neighborhood (e.g. grass next to a residential building) whereas others are rather unlikely (e.g. a residential building on a highway). We propose extending ϱ to a vector of dimension C, thus changing Equation to (20)

$$\hat{g}_i = \arg\max_{g_i} \Big\{ p(y_i = g_i) \frac{1}{Z(\varrho(g_i))} \exp\left(\varrho(g_i) \right. \\ \left. \#\{y_j \in \mathcal{N}_{y_i} | y_j = y_j\}\right)\Big\} \tag{23}$$

Note that the optimal $\varrho = [\varrho(1), \varrho(2), ..., \varrho(C)]^T$ can be estimated similarly to Equation (22) using the Cauchy-Schwarz inequality as,

$$\hat{\varrho} = \arg\min_{\varrho} \|\varrho\|_2^2 \cdot \left\| \frac{\partial \hat{\mathbf{g}}_\varrho}{\partial \varrho} \right\|_2^2 \tag{24}$$

Similarly to Equation (22), Equation (24) allows estimating the optimal vector of attraction parameters using the data at hand only.

D. Relation to the bias-variance decomposition

Note that the choice of the regularization or attraction parameter directly affects the bias-variance tradeoff. We note that $\varrho = 0$ reduces the MAP estimator to a maximum likelihood estimator that estimates $\mathbf{g}$ as the label field that maximizes $p(\mathbf{y} \mid \mathbf{g})$. Equivalently, $\gamma = 0$ reduces the Tikhonov regularization to an ordinary least squares solution. Choosing a regularization or attraction parameter > 0 introduces a bias and decreases variance. This can be explained by the matrix $\mathbf{L}$ in Equations (19) and (21) which can be interpreted as a smoothing operator on the solution $\mathbf{g}$, i.e. the higher γ or ϱ, the stronger the impact of $\mathbf{L}$, the smoother the solution $\hat{\mathbf{g}}$ and consequently the lower the variance of the estimator. It is noteworthy that it has been shown in [90] that Equation (21) achieves the uniform Cramér-Rao lower bound [91], i.e. from all linear and nonlinear estimators of $\mathbf{g}$ in Equation (14) Tikhonov regularization minimizes the total variance provided that ω is zero-mean Gaussian distributed.

VI. Experimental Results

In order to assess the performance of the proposed approach under various real world situations we applied it to a number of different hyperspectral image classification problems, such as 'Indian Pines', 'Pavia Center', 'Pavia University', and 'Pavia Extended'. Comparisons with some of the most widely used methods in the field of hyperspectral classification are performed and some generic observations with respect to the bias-variance tradeoff discussed in previous sections are outlined.

A. Hyperspectral datasets

The first classification problem is the 'Indian Pines', one of the most common and well studied cases in the hyperspectral literature. The image was acquired by the AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) sensor over the agricultural Indian Pine test site in Northwestern Indiana. Its size is 145 $\times$ 145 pixels and the spatial resolution is 20m per pixel. Twenty water absorption bands have been removed as indicated in the literature, leaving the remaining 200-band image for classification. The ground truth contains sixteen classes, mainly covering different types of crops: corn-no till, corn-min till, corn, coybeans-no till, soybeans-min till, soybeans-clean till, alfalfa, grass/pasture, grass/trees, grass/pasture-mowed, hay-windrowed, oats, wheat, woods, bldg-grass-tree-drives, and stone-steel towers. The training set consists of 50 samples for each class that have been randomly chosen from the reference data, except for classes alfalfa, grass/pasturemowed and oats that have very few members, thus only 15 samples

for each of these classes were randomly picked to be used as training samples. All other samples composed the test set.

The second classification problem is Pavia. In this case three distinct maps ('Pavia University', 'Pavia Center', and 'Pavia Extended') are considered in order to derive conclusions for different types of urban environments. The flight over the city of Pavia, Italy, was organized by DLR (Deutsche Luft- und Raumfahrtgesellschaft, the German Aerospace Agency) in the framework of the HySens project, sponsored by the European Union. The used sensor in this case was the ROSIS-03, which provides 115 bands with a spectral coverage ranging from 0.43 to 0.86μm and a bandwidth of 4nm. The spatial resolution in this case is 1.3m per pixel.

In the 'Pavia University' dataset, the dimensions of the picture are 610 $\times$ 340 pixels. Twelve noisy bands have been removed, leaving 103 spectral bands. The classes of interest are the following nine: tree, asphalt, bitumen, gravel, metal sheet, shadow, bricks, meadow, and soil. The second dataset is 'Pavia Center', with a size of 1096 $\times$ 489 pixels. The classes of interest are nine: water, tree, meadow, brick, soil, asphalt, bitumen, tile, and shadow. The third dataset is 'Pavia Extended'. The original size of the image was 1096 $\times$ 1096 pixels, including a 381 pixel wide black band in the left-hand part of the image that was removed, resulting in a two-part image of 1096 $\times$ 715 pixels. Thirteen noisy bands have been removed, leaving 102 spectral bands. The classes of interest are the same nine as in the 'Pavia Center' case. For each of the Pavia datasets we consider a training set that is slightly smaller than the ones in [16], [36] and specifically it includes 5174 training samples for 'Pavia Center', 2197 for 'Pavia University' and 7409 for 'Pavia Extended'.

B. Model parameters and feature selection

We start the assessment with a comparison between the random forest algorithm and a standard state of the art classification based on SVM. For both models we progressively add more features, according to the four steps of our feature extraction methodology (Fig. 2), starting from MNF, going to abundance maps (AM), proceeding with NWFE, and finally adding the synthetic features. The SVM model is using the Radial Basis Function (RBF) kernel with its parameters being selected using cross validation. For random forest, a relaxed choice in terms of using the larger of the three proposed values for the feature set splitting variable ($mtry = 2\sqrt{p}$) was made, since the initial assumption is that we do not want to be too aggressive in reducing variance (at the expense of bias) compared to the case of using just the tree bagging ensemble ($mtry = all$). The comparison is focused on the two most challenging of the four datasets, namely 'Indian Pines' and 'Pavia University' since these two are the most difficult of the datasets and some significant performance difference can be achieved as more features are added.

As can be seen from Tables II and III, the random forest algorithm outperforms the SVM seven out of eight times, even with the optimal parameter selection (through cross-validation) for the SVM. The tables also show that the impact of data preparation (e.g. extracting the right features) on both bias and variance reduction can be quite significant. Another important observation is that the SVM for the 'Indian Pines' case is getting its best accuracy with only the MNF components and the abundance maps as features, while the random forest has a small degradation in accuracy with NWFE but gains significantly when the synthetic features are added. This implies that the random forest is less prone to overfitting the training set as the number of predictor variables increases, more robust to the Hughes phenomenon, or both.

Indian Pines	**SVM (%)**	**RF $2\sqrt{p}$ (%)**
MNF	70.41	**72.59**
+ AM	**81.34**	80.92
+ NWFE	79.45	**80.48**
+ Synthetic	81.08	**87.44**

TABLE II: *SVM versus RF results for Indian Pines*

Pavia University	**SVM (%)**	**RF $2\sqrt{p}$ (%)**
MNF	86.93	**88.57**
+ AM	89.21	**90.85**
+ NWFE	91.08	**91.12**
+ Synthetic	93.65	**95.01**

TABLE III: *SVM versus RF results for Pavia University*

In the second experiment we consider how the parameters of the random forest algorithm affect the bias-variance tradeoffs and what is the impact of feature selection. Three cases for feature selection are considered that are 'All' (all features are kept), 'Select > 0' (keep only features with a positive impact factor) and 'Select > 0.3' (keep only features with an impact factor > 0.3). In all cases we consider the full feature set (MNF, abundance maps, NWFE, and synthetic) for initialization. We evaluate a version of the random forest that, as in the previous case, is less aggressive in reducing variance ($mtry = 2\sqrt{p}$) and an evaluation is done with 100 and 500 trees. As can been seen from Tables IV and V, all models that have 500 trees are benefiting from the application of some form of feature selection (either the conservative one that selects all features with positive impact or the more aggressive one). With less trees the estimation of important features can be unstable, therefore using all the features might be the best choice (e.g., in the 'Indian Pines' case).

Particularly for 'Pavia University', the models perform best with aggressive feature selection ('Select > 0.3'). In the 'Indian Pines' case the random forest performs better with conservative feature selection ('Select > 0') and the tree bagger performs better with the aggressive selection. This is justified if we consider that random forest inherently reduces variance by limiting the number of features used in every node while tree bagger relies on explicit feature selection for the same effect. Finally, both models benefit from an increase in the number of trees in the ensemble. A higher number of trees increases the variance reduction benefit of bagging, while for the random forest case, it provides more opportunities for key features to be selected, further stabilizing the selected hypothesis. The best classification results for the 'Indian Pines' and 'Pavia University' cases are presented in Fig. 3 and 4, both before and after segmentation.

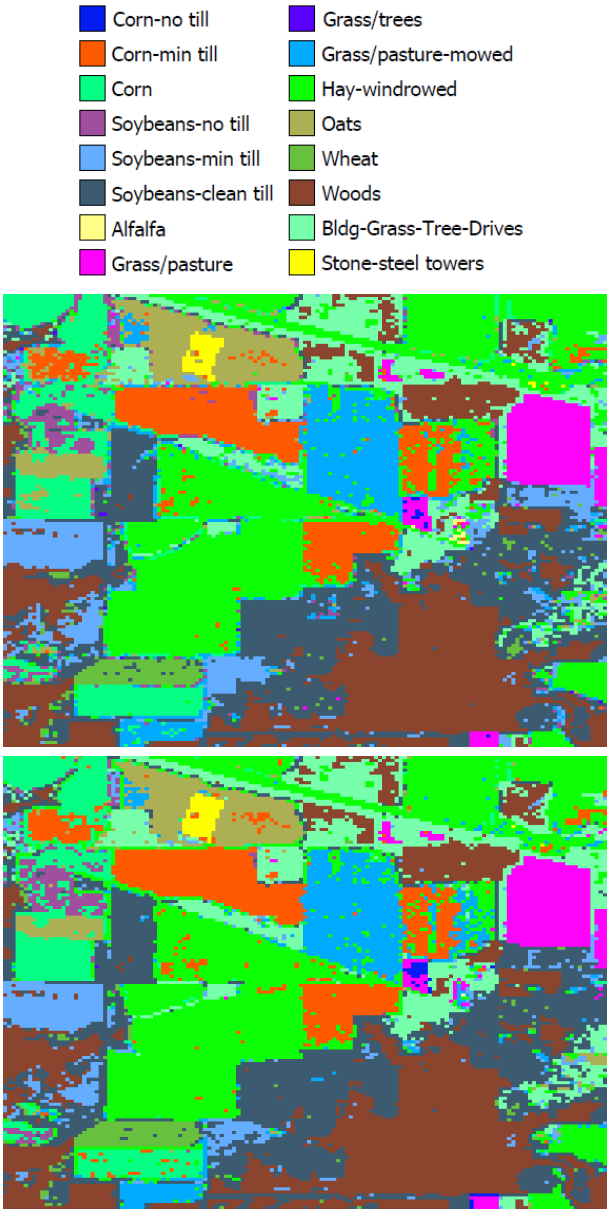

Fig. 3: *Classification results for Indian Pines using random forests before (up) and after (down) segmentation*

Indian Pines	**All (%)**	**Select >0 (%)**	**Select >0.3 (%)**
RF ($2\sqrt{p}$), 100	87.44	87.31	86.56
RF ($2\sqrt{p}$), 500	87.62	**88.07**	86.91
Tree bagger, 100	86.83	85.24	85.84
Tree bagger, 500	85.49	85.60	87.58

TABLE IV: *Evaluating the effect of feature selection: Results for Indian Pines*

Pavia University	**All (%)**	**Select >0 (%)**	**Select >0.3 (%)**
RF ($2\sqrt{p}$), 100	95.01	95.28	95.71
RF ($2\sqrt{p}$), 500	95.42	95.72	**96.01**
Tree bagger, 100	95.45	94.59	94.94
Tree bagger, 500	95.21	94.86	95.10

TABLE V: *Evaluating the effect of feature selection: Results for Pavia University*

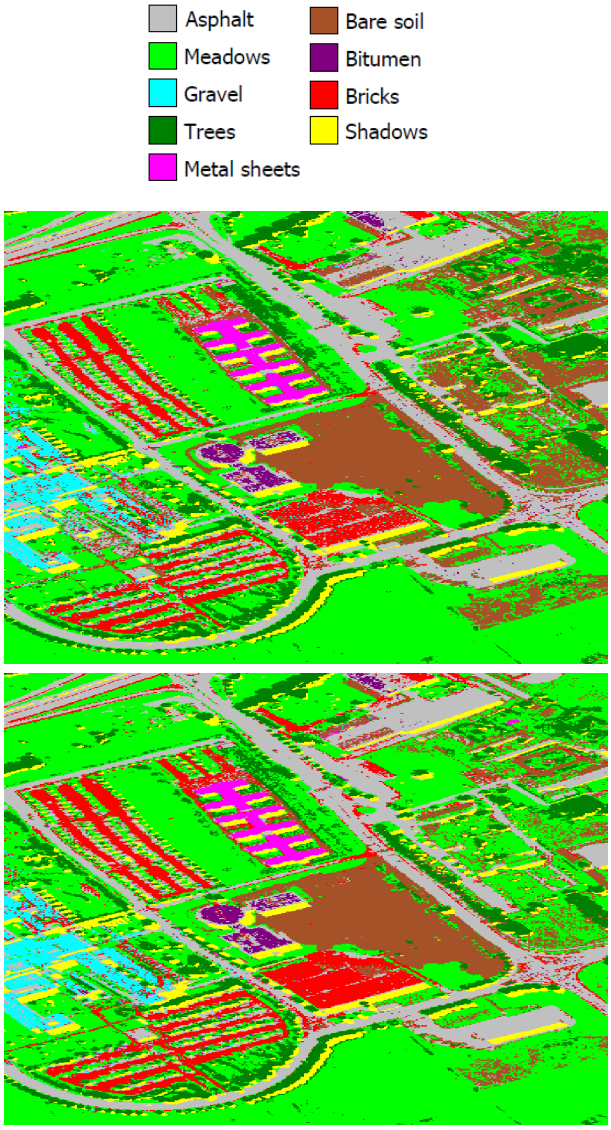

Fig. 4: *Classification results for Pavia University using random forests before (up) and after (down) segmentation*

C. Impact of post-classification segmentation

Finally, we consider the comparison of the best results achieved in the previous tables (e.g., with all features, optimal feature selection, 500 trees, etc.) for all the considered models (SVM, tree bagger ensemble, and three realizations of the random forest for the three different values of $mtry$ proposed in the literature). On top of the classification outcome, in this case we apply also post-classification segmentation, as described in Section V. Specifically, Equation (20) is solved using the ICM algorithm where the optimal attraction parameter vector ϱ is found by minimizing the gradient of the estimated label field as per Equation (24). In this experiment we consider all the available hyperspectral images, to allow a comparison of the different classifier models in a broad range of cases, but also to assess the outcome of segmentation in classification results that vary significantly covering a range between 81% and 99% accuracy (Tables VI - Table IX).

Indian Pines	Classification (%)	Segmentation (%)
Tree bagger	87.58	**93.03**
RF ($2\sqrt{p}$)	**88.07**	92.31
RF ($\sqrt{p}$)	86.24	90.93
RF ($0.5\sqrt{p}$)	85.38	91.43
SVM	81.08	89.22

TABLE VI: *Classification results for Indian Pines*

Pavia University	Classification (%)	Segmentation (%)
Tree bagger	95.45	96.72
RF ($2\sqrt{p}$)	**96.01**	**97.99**
RF ($\sqrt{p}$)	95.35	97.37
RF ($0.5\sqrt{p}$)	94.16	97.14
SVM	93.65	96.67

TABLE VII: *Classification results for Pavia University*

A first observation is that the application of the proposed post-classification scheme improves classification accuracy for all datasets and classifiers. We observe that the random forest algorithm provides the best classification result in two ('Indian Pines' and 'Pavia University') of the four cases, however after application of segmentation it provides the best overall result only in the case of 'Pavia University' and the tree bagging ensemble overtakes it as the best choice for 'Indian Pines'. This is an important observation that shows in practice that the different steps of the hyperspectral image classification problem are closely related in terms of the bias-variance tradeoff: the tree bagger ensemble had (significantly) more variance and less bias after just classification compared to the random forest, but since a large part of the variance was addressed by the segmentation, it emerged as a better overall choice.

Another important observation is that the SVM model (using the optimal parameters derived from cross-validation) does relatively poorly in the 'Indian Pines' case but excellent in the 'Pavia Center', where it also is the best model. In general the more complex (in terms of allowing additional variance) models are doing better in the easier cases like 'Pavia Center' and 'Pavia Extended', while they are doing less well in the more difficult problems such as 'Indian Pines' and 'Pavia University'. A more challenging dataset (e.g., with presence of shadows) would probably further underline this property, favoring models that are better at reducing variance.

Finally, it is also interesting to observe that the post segmentation results for 'Pavia Extended' are remarkably close in terms of accuracy, an indication that the models have squeezed almost every available performance point and are close to the theoretical maximum.

Pavia City	Classification (%)	Segmentation (%)
Tree bagger	98.73	99.25
RF ($2\sqrt{p}$)	98.34	99.25
RF ($\sqrt{p}$)	98.41	99.21
RF ($0.5\sqrt{p}$)	98.36	99.21
SVM	**98.76**	**99.53**

TABLE VIII: *Classification results for Pavia Center*

Pavia Extended	Classification (%)	Segmentation (%)
Tree bagger	**99.07**	**99.56**
RF ($2\sqrt{p}$)	99.01	99.55
RF ($\sqrt{p}$)	98.81	99.52
RF ($0.5\sqrt{p}$)	98.76	99.55
SVM	98.93	99.52

TABLE IX: *Classification results for Pavia Extended*

D. Best practices for hyperspectral image classification

We conclude the experimental Section with some generic guidelines and best practices for classification of hyperspectral imagery. Attempting to lower the bias by using complex learners is only useful when parameters can be estimated reliably. This requires a relatively large training set and/or small amount of measurement noise. In fact, since all steps of the classification process risk increasing variance error, the stability of the hypothesis should be tested: if a method, when applied to different training subsets, produces hypotheses with significantly different predictions, this indicates that the variance component of error might be dominant.

In the case of noisy hyperspectral images (e.g., presence of shadows, noisy bands or other artifacts), the models can suffer from significant variance, therefore choices should be more focused at reducing the variance component rather than selecting an optimal bias. This applies not only to the learning algorithm itself (favoring variance-focused tradeoffs) but to the other stages as well (e.g., favoring a smaller value for $mtry$ or more aggressive feature selection). However, the application of post-classification segmentation can change the game, allowing less variance reduction in the classification part (like with the tree bagging ensemble or SVM) since the segmentation will further improve the variance error. This means that a hypothesis that is not the absolute best one in terms of classification performance due to variance error can result in a better outcome compared to the best classification result after segmentation has been applied to both.

With respect to the selection of models and models parameters, cross-validation is the most frequently used method, but as an empirical technique it can be misused, resulting in non-optimal selections. One of the most known pitfalls is using cross-validation to assess several parameters of the model, and only reporting the outcome for the hypothesis with the best results. If the number of parameters to be selected is significant this contaminates validation to a train scenario and the error estimate becomes strongly biased. Another problem is performing the initial analysis to identify the most informative features using the entire data set; if feature selection or model tuning is required by the modeling procedure (like in the NWFE transform in our case), this must be performed on each training set. Furthermore, if cross-validation is used to decide which features to keep, an inner cross-validation to carry out the feature selection on every training set must be performed. Finally, having some of the training data also included in the validation set (e.g., due to "twinning", a case in which some identical or almost identical samples are present in the data set) can result to misleading error estimations. As long as such pitfalls are avoided, cross-validation remains the most systematic way of selecting between models.

Our final remark goes to the application of Markov-Random field based methods in post-classification. We stress the fact that the parameters of an MRF typically cannot be optimized via cross-validation as most training sets will not contain enough spatially connected pixels to estimate e.g. the attraction or regularization parameter. Instead, approaches such as Tikhonov's quasi-optimality criterion or methods such as augmented Tikhonov regularization [88] are elegant ways to optimize the respective parameters using only the data at hand.

VII. Conclusion

We presented a unified framework for hyperspectral image classification. Its novelty lies in the fact that it considers all steps of a classification chain such as feature extraction, feature selection, classification, and post-processing from a bias-variance decomposition point of view. This allows for the formulation of a consistent framework that jointly optimizes the steps of the classification process and thus improves accuracy. Further contributions include the formulation and systematic evaluation of ensemble learning, feature selection and Markov Random-Field-based post-processing in four standard hyperspectral imaging datasets. Our analysis revealed that random forests with embedded feature selection are highly effective methods in hyperspectral image classification that can be steered towards a favorable bias-variance tradeoff based on the properties of each dataset. Finally, the proposed Markov Random-Field-based post-processing scheme yielded an increase in classification accuracy in all considered cases. The fact that it was designed in a fully automatic way, estimating its parameters from the data at hand, makes it an attractive tool in classification of hyperspectral imagery.

Acknowledgement

The authors would like to thank Prof. Paolo Gamba from University of Pavia, Italy and Prof. Mathieu Fauvel from INPT-ENSAT, Toulouse, France for providing the 'Pavia' datasets.

References

[1] C.-C. Liu. Processing of FORMOSAT-2 daily revisit imagery for site surveillance. *IEEE Transactions on Geoscience and Remote Sensing*, 44(11):3206–3214, 2006.

[2] C. Ren and C.-I. Chang. Automatic spectral target recognition in hyperspectral imagery. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1232–1249, 2003.

[3] A. Banerjee, P. Burlina, and J. Broadwater. Hyperspectral video for illumination-invariant tracking. In *First Workshop on Hyperspectral Image and Signal Processing*, pages 1–4, 2009.

[4] S. Matteoli, M. Diani, and G. Corsini. A tutorial overview of anomaly detection in hyperspectral images. *IEEE Aerospace and Electronic Systems Magazine*, 25(7):5–28, 2010.

[5] P.S. Thenkabail, J.G. Lyon, and A. Huete, editors. *Hyperspectral remote sensing of vegetation*. CRC Press, 2011.

[6] D. Haboudane, J.R. Miller, E. Pattey, P.-J. Zarco-Tejada, and I.B. Strachan. Hyperspectral vegetation indices and novel algorithms for predicting green LAI of crop canopies: Modeling and validation in the context of precision agriculture. *Remote Sensing of Environment*, 90(3):337 – 352, 2004.

[7] E. Hirsch and E. Agassi. Detection of gaseous plumes in ir hyperspectral images, performance analysis. *IEEE Sensors Journal*, 10(3):732–736, 2010.

[8] V. Farley, M. Chamberland, P. Lagueux, A. Vallires, A. Villemaire, and J. Giroux. Chemical agent detection and identification with a hyperspectral imaging infrared sensor. In *Proc. SPIE 6661, Imaging Spectrometry XII*, 2007.

[9] Richard J. Ellis and Peter W. Scott. Evaluation of hyperspectral remote sensing as a means of environmental monitoring in the st. austell china clay (kaolin) region, cornwall, UK. *Remote Sensing of Environment*, 93(1??2):118 – 130, 2004.

[10] S. Serranti, A. Gargiulo, and G. Bonifazi. Dried fruits quality assessment by hyperspectral imaging. In *Proc. SPIE 8369, Sensing for Agriculture and Food Quality and Safety IV*, 2012.

[11] A.A. Gowen, C.P. O'Donnell, P.J. Cullen, G. Downey, and J.M. Frias. Hyperspectral imaging - an emerging process analytical tool for food quality and safety control. *Trends in Food Science & Technology*, 18(12):590 – 598, 2007.

[12] S. Serranti, A. Gargiulo, and G. Bonifazi. Characterization of post-consumer polyolefin wastes by hyperspectral imaging for quality control in recycling processes. *Waste Management*, 31(11):2217 – 2227, 2011.

[13] S. Serranti and Bonifazi G. Hyperspectral imaging based recognition procedures in particulate solid waste recycling. *World Review of Science, Technology and Sustainable Development*, 7(3):271–281, 2010.

[14] J. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot. Hyperspectral remote sensing data analysis and future challenges. *IEEE Geoscience and Remote Sensing Magazine*, 1(2):6–36, 2013.

[15] J. Chanussot, M. M. Crawford, and B.-C. (eds.) Kuo. Special issue on hyperspectral image and signal processing. *IEEE Transactions on Geoscience and Remote Sensing*, 48(11), November 2010.

[16] M. Fauvel, Y. Tarabalka, J. Benediktsson, J. Chanussot, and J. Tilton. Advances in spectral–spatial classification of hyperspectral images. *Proceedings of the IEEE*, 101(3):652–675, 2013.

[17] D.G. Goodenough, Hao Chen, A. Dyk, A. Richardson, and Geordie Hobart. Data fusion study between polarimetric SAR, hyperspectral and lidar data for forest information. In *IEEE International Geoscience and Remote Sensing Symposium*, volume 2, pages II–281–II–284, 2008.

[18] A. Brook, E. Ben-Dor, and R. Richter. Fusion of hyperspectral images and LiDAR data for civil engineering structure monitoring. In *Proceedings of the Hyperspectral Workshop*, 2010.

[19] M. Dalponte, L. Bruzzone, and D. Gianelle. Fusion of hyperspectral and LiDAR remote sensing data for classification of complex forest areas. *IEEE Transactions on Geoscience and Remote Sensing*, 46(5):1416–1427, 2008.

[20] L. Liu, Y. Pang, W. Fan, Z. Li, and M. Li. Fusion of airborne hyperspectral and LiDAR data for tree species classification in the temperate forest of northeast china. In *19th International Conference on Geoinformatics*, pages 1–5, 2011.

[21] G. Chen and S.-E. Qian. Denoising of hyperspectral imagery using principal component analysis and wavelet shrinkage. *IEEE Transactions on Geoscience and Remote Sensing*, 49(3):973–980, 2011.

[22] P. Zhong and R. Wang. Multiple-spectral-band CRFs for denoising junk bands of hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 51(4):2260–2275, 2013.

[23] J.M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot. Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 5(2):354–379, 2012.

[24] I. Dopido, A. Villa, A. Plaza, and P. Gamba. A quantitative and comparative assessment of unmixing-based feature extraction techniques for hyperspectral image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 5(2):421–435, 2012.

[25] S. Samiappan, S. Prasad, and L.M. Bruce. Automated hyperspectral imagery analysis via support vector machines based multi-classifier system with non-uniform random feature selection. In *IEEE International Geoscience and Remote Sensing Symposium*, pages 3915–3918, 2011.

[26] R. Zhang, J. Ma, X. Chen, and Q. Tong. Feature selection for hyperspectral data based on modified recursive support vector machines. In *IEEE International Geoscience and Remote Sensing Symposium*, volume 2, pages II–847–II–850, 2009.

[27] R. Fandos, C. Debes, and A.M. Zoubir. Resampling methods for quality assessment of classifier performance and optimal number of features. *Signal Processing*, 93(11):2956 – 2968, 2013.

[28] Y. Tarabalka, J. Chanussot, and J.A. Benediktsson. Segmentation and classification of hyperspectral images using watershed transformation. *Pattern Recognition*, 43(7):2367 – 2379, 2010.

[29] M. Grana, I. Villaverde, J.O. Maldonado, and C. Hernandez. Two lattice computing approaches for the unsupervised segmentation of hyperspectral images. *Neurocomputing*, 72(10-12):2111 – 2120, 2009.

[30] W. Liao, R. Bellens, A. Pizurica, W. Philips, and Y. Pi. Classification of hyperspectral data over urban areas using directional morphological profiles and semi-supervised feature extraction. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 5(4):1177–1190, 2012.

[31] G. Matasci, D. Tuia, and M. Kanevski. SVM-based boosting of active learning strategies for efficient domain adaptation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 5(5):1335–1343, 2012.

[32] Y. Tarabalka, J. Chanussot, and J. Benediktsson. Segmentation and classification of hyperspectral images using minimum spanning forest grown from automatically selected markers. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 40(5):1267–1279, 2010.

[33] X. Huang and L. Zhang. An SVM ensemble approach combining spectral, structural, and semantic features for the classification of high-resolution remotely sensed imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 51(1):257–272, 2013.

[34] J. Ham, Yangchi Chen, M. M. Crawford, and J. Ghosh. Investigation of the random forest framework for classification of hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing*, 43(3):492–501, 2005.

[35] D. Tuia and G. Camps-Valls. Urban image classification with semisupervised multiscale cluster kernels. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 4(1):65–74, 2011.

[36] M.D. Mura, A. Villa, J.A. Benediktsson, J. Chanussot, and L. Bruzzone. Classification of hyperspectral images by using extended morphological attribute profiles and independent component analysis. *IEEE Geoscience and Remote Sensing Letters*, 8(3):542–546, 2011.

[37] E. Bienenstock S. Geman and R.Doursat. Neural networks and the bias/variance dilemma. *Neural Computation*, 4:1–58, 1992.

[38] J. Friedman. On bias, variance, 0/1-loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery*, 1:55–77, 1997.

[39] R. Kohavi and D.H. Wolpert. Bias plus variance decomposition for zero-one loss functions. In *Proceedings of the thirteenth international conference on Machine Learning*, pages 275 – 283, 1996.

[40] L. Breiman. Bias, variance, and arcing classifiers. *Technical Report, Statistics Department, University of California.*, 1996.

[41] P. Domingos. A unified bias-variance decomposition and its applications. In *Seventeenth International Conference on Machine Learning*, pages 231–238, 2000.

[42] G.M. James. Variance and bias for general loss functions. *Machine Learning*, 51:115–135, 2003.

[43] L. Breiman. Random forests. *Machine Learning*, 45(1):5 – 32, 2001.

[44] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*, IJCAI'95, pages 1137–1143, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[45] K. Fukunaga. *Introduction to statistical pattern recognition (2nd ed.)*. Academic Press Professional, Inc., San Diego, CA, USA, 1990.

[46] H.L. Wei and S.A. Billings. Feature subset selection and ranking for data dimensionality reduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1):162–166, 2007.

[47] Joseph O'Sullivan. Integrating initialization bias and search bias in neural network learning, 1996.

[48] G. Hughes. On the mean accuracy of statistical pattern recognizers. *IEEE Transactions on Information Theory*, 14(1):55–63, September 2006.

[49] L.J.P. van der Maaten, E. O. Postma, and H. J. van den Herik. Dimensionality reduction: A comparative review. Technical report, 2008.

[50] P. Switzer A.A. Green, M. Berman and M.D. Craig. A transformation for ordering multispectral data in terms of image quality with implications for noise removal. *IEEE Transactions on Geoscience and Remote Sensing*, 26:65–74, 1988.

[51] J.W. Boardman and F.A.Kruse. Automated spectral analysis: a geological example using AVIRIS data, north grapevine mountains, nevada. In *Tenth Thematic Conference on Geologic Remote Sensing*, 1994.

[52] A. A. Nielsen. Kernel maximum autocorrelation factor and minimum noise fraction transformations. *IEEE Transactions on Image Processing*, 20 (3):612–624, 2011.

[53] A. A. Nielsen L. Gomez-Chova and G. Camps-Valls. Explicit signal to noise ratio in reproducing kernel hilbert spaces. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, pages 3570 – 3573, 2011.

[54] B. Boashash, P. O'Shea, and M. J. Arnold. Algorithms for instantaneous frequency estimation: a comparative study. In F. T. Luk, editor, *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 1348, pages 126–148, November 1990.

[55] C. L. Lawson and R. J. Hanson. *Solving least squares problems*. Society for Industrial and Applied Mathematics, 3 edition, 1995.

[56] B. Luo and J. Chanussot. Hyperspectral image classification based on spectral and geometrical features. In *IEEE International Workshop on Machine Learning for Signal Processing*, pages 1–6, 2009.

[57] B.C. Kuo and D.A. Landgrebe. Nonparametric weighted feature extraction for classification. *IEEE Transactions on Geoscience and Remote Sensing*, 42(5):1096–1105, 2004.

[58] C. Wan-Wei R. Hsuan and P. Yen-Nan. Nonparametric weighted feature extraction for noise whitening least squares. *Proc. SPIE 6756, Chemical and Biological Sensors for Industrial and Environmental Monitoring III, 67560E*, 2007.

[59] P. Van Der Putten and M. Van Someren. A bias-variance analysis of a real world learning problem: The coil challenge 2000. *Machine Learning*, 57:177–195, 2004.

[60] D. Wolpert. Stacked generalization. *Neural Networks*, 5 (2):241–259, 1992.

[61] M. Prior and T. Windeatt. An ensemble dependence measure. In *Proceedings of the 17th international conference on Artificial neural networks*, ICANN'07, pages 329–338, Berlin, Heidelberg, 2007. Springer-Verlag.

[62] L. Kuncheva and C. Whitaker. Measures of diversity in classifier ensembles. *Machine Learning*, 51:181–207, 2003.

[63] B. Efron. Bootstrap Methods: Another Look at the Jackknife. *The Annals of Statistics*, 7(1):1–26, 1979.

[64] R. Harris G. Brown, J. Wyatt and X. Yao. Diversity creation methods: a survey and categorisation. *Information Fusion*, 6 (1):5–20, 2005.

[65] L. Breiman. Bagging predictors. *Machine Learning*, 24, 1996.

[66] M. Kearns. Thoughts on hypothesis boosting. *Unpublished manuscript*, 1988.

[67] P. Bartlett R.E. Schapire, Y. Freund and W.S. Lee. Boosting the margin: A new explanation for the efectiveness of voting methods. In *Proceedings of the Fourteenth International Conference on Machine Learning*, 1997.

[68] T.K. Ho. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20 (8):832–844, 1998.

[69] T. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees : Bagging, boosting and randomization. *Machine Learning*, pages 1–22, 1999.

[70] V. Svetnik, A. Liaw, C. Tong, J. Culberson, R. Sheridan, and B. Feuston. Random forest: A classification and regression tool for compound classification and QSAR modeling. *Journal of Chemical Information and Computer Sciences*, 43:1947–1958, 2003.

[71] R. Diaz-Uriarte and S. Alvarez de Andres. Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, 7 (3):1–13, 2006.

[72] J.-M. Poggi R. Genuer and C. Tuleau. Random forests: Some methodological insights. *arXiv:0811.3619*, 2008.

[73] J.-M. Poggi R. Genuer and C. Tuleau-Malot. Variable selection using random forests. *Pattern Recognition Letters*, 31:2225–2236, 2010.

[74] Y. Freund and R. Shapire. Experiments with a new boosting algorithm. In In L. Saitta, editor, *Proceedings of the 13th International Conference on Machine Learning*, pages 14–156, San Francisco, 1996.

[75] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press,, 2004.

[76] Lidia Ceriani and Paolo Verme. The origins of the gini index: extracts from variabilit e mutabilit (1912) by corrado gini. *Journal of Economic Inequality*, 10(3):421–443, September 2012.

[77] Y. Lin and Y. Jeon. Random forests and adaptive nearest neighbors. *Journal of the American Statistical Association*, 101:578–590, 2006.

[78] G. Biau and L. Devroye. On the layered nearest neighbour estimate, the bagged nearest neighbour estimate and the random forest method in regression and classification. *Journal of Multivariate Analysis*, 101:2499–2518, 2010.

[79] N. Meinshausen. Quantile regression forests. *Journal of Machine Learning Research*, 7:983–999, 2006.

[80] G. Biau, L. Devroye, and G. Lugosi. Consistency of random forests and other averaging classifiers. *Journal of Machine Learning Research*, 9:2015–2033, 2008.

[81] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.
[82] G. Winkler. *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods*. Springer, 2003.
[83] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society*, 48:259–302, 1986.
[84] C. Debes, J. Hahn, A.M. Zoubir, and M.G. Amin. Target discrimination and classification in through-the-wall radar imaging. *IEEE Transactions on Signal Processing*, 2011. to appear.
[85] N.D.A. Mascarenhas and A.C. Frery. SAR image filtering with the ICM algorithm. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, volume 4, pages 2185–2187, 1994.
[86] A. N. Tikhonov. Solution of incorrectly formulated problems and the regularization method. *Soviet. Math. Dokl.*, 5:1035–1038, 1963.
[87] A. N. Tikhonov and V. Y. Arsenin. *Solution of Ill-Posed Problems*. V.H. Winston, 1977.
[88] Bangti Jin and Jun Zou. Augmented tikhonov regularization. *Inverse Problems*, 25(2):025001, 2009.
[89] K. Ito and B. Jin. A new approach to nonlinear constrained tikhonov regularization. *Inverse Problems*, 27(10):105005, 2011.
[90] Y.C. Eldar. Minimum variance in biased estimation: bounds and asymptotically optimal estimators. *IEEE Transactions on Signal Processing*, 52(7):1915–1930, 2004.
[91] A.O. Hero, J.A. Fessler, and M. Usman. Exploring estimator bias-variance tradeoffs using the uniform CR bound. *IEEE Transactions on Signal Processing*, 44(8):2026–2041, 1996.

PLACE PHOTO HERE

Andreas Merentitis Dr. Andreas Merentitis received the B.Sc., M.Sc., and Ph.D. degrees from the Department of Informatics and Telecommunications, National Kapodistrian University of Athens (NKUA) in 2003, 2005, and 2010 respectively. He holds an award from the Greek Mathematical Society and received a scholarship as one of the two highest performing M.Sc. students in his year. Dr. Merentitis has worked as Senior Researcher at the NKUA and has participated in numerous European funded projects, in many cases as a member of the proposal writing team and sub-workpackage leader of key workpackages. Since 2011 he works as Senior Researcher at AGT International. He has more than 25 publications in the thematic areas of machine learning, embedded systems, distributed systems, and remote sensing, including publications in flagship conferences and journals. He was awarded an IEEE Certificate of Appreciation as a core member of the team that won the first place in the Best Classification Challenge of the 2013 IEEE GRSS Data Fusion Contest. He is a member of the IEEE and the IEEE Computer and Communication Societies.

PLACE PHOTO HERE

Christian Debes Dr. Christian Debes (S07-M09-SM13) received the MSc. and Dr.-Ing. degree with highest honor from Technische Universität Darmstadt, Darmstadt, Germany in 2006 and 2010, respectively. Since 2010 he is lecturer at the department of electrical engineering and information technology at Technische Universität Darmstadt. Dr. Debes joined the research center of AGT International in 2011 where he now holds a position as research architect in data analytics. His research interests include data fusion, classification and image processing for remote sensing applications. He is recipient of the IEEE GRSS 2013 Data Fusion Best classification award and has more than 25 journal and conference papers in target detection, classification and image processing. Dr. Debes is a senior member of the IEEE and member of the editorial board of Elsevier Digital Signal Processing since 2013.

PLACE PHOTO HERE

Roel Heremans Dr. Roel Heremans received the MSc. and Dr. degree in elementary particle physics from the Free university Brussels (VUB), Belgium in 1996 and 2001, respectively. From 2002 until 2011 Dr. Roel Heremans joined the Royal Military Academy in Brussels where he worked in the Signal and Image Center (SIC) on synthetic aperture processing in the field of RADAR, SONAR and THz. In 2004 he was research assistant at the NATO underwater research center, La Spezia, Italy, while working on the reconstruction and motion compensation of synthetic aperture sonar data. He developed PolInSAR features in a project for the Belgian Federal Science Policy Office (BELSPO). He was active in the field of Hyperspectral Image processing where he developed change detection algorithms and developed a method to determine gas pollutant concentrations above the petrochemical industry in Antwerp, Belgium. He joined AGT Germany since 2012 where he works as Senior Researcher on machine learning and data analytics. He was a scientific evaluator of projects undertaken by the European Space Agency and Institute for Scientific Research and Innovation (IRSIB). His research interest include data fusion, classification, change- and anomaly detection.