IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 19, 2020, accepted February 27, 2020, date of publication March 2, 2020, date of current version March 11, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2977454

Hyperspectral Band Selection Using Attention-Based Convolutional Neural Networks

PABLO RIBALTA LORENZO[1], (Student Member, IEEE), LUKASZ TULCZYJEW[1,2], MICHAL MARCINKIEWICZ[2], AND JAKUB NALEPA[1], (Member, IEEE)
[1]Department of Algorithmics and Software, Silesian University of Technology, 44 100 Gliwice, Poland
[2]KP Labs, 44 100 Gliwice, Poland

Corresponding author: Jakub Nalepa (jnalepa@ieee.org)

This work was supported in part by the Polish National Centre for Research and Development under Grant POIR.01.01.01-00-0356/17, and in part by the European Space Agency (HYPERNET and BEETLES projects). The work of Jakub Nalepa was supported by the Silesian University of Technology funds through The Rector's Habilitation Grant 02/020/RGH19/0185 and Grant 02/020/BKM19/0183.

ABSTRACT Hyperspectral imaging has become a mature technology which brings exciting possibilities in various domains, including satellite image analysis. However, the high dimensionality and volume of such imagery is a serious problem which needs to be faced in Earth Observation applications, where efficient acquisition, transfer and storage of hyperspectral images are key factors. To reduce the time (and ultimately cost) of transferring hyperspectral data from a satellite back to Earth, various band selection algorithms have been proposed. They are built upon the observation that for a vast number of applications only a subset of all bands convey the important information about the underlying material, hence we can safely decrease the data dimensionality without deteriorating the performance of hyperspectral classification and segmentation techniques. In this paper, we introduce a novel algorithm for hyperspectral band selection that couples new attention-based convolutional neural networks used to weight the bands according to their importance with an anomaly detection technique which is exploited for selecting the most important bands. The proposed attention-based approach is data-driven, re-uses convolutional activations at different depths of a deep architecture, identifying the most informative regions of the spectrum. Also, it is modular, easy to implement, seamlessly applicable to any convolutional network, and can be trained end-to-end using gradient descent. Our rigorous experiments, performed over benchmark sets and backed up with statistical tests, showed that the deep models equipped with the attention mechanism are competitive with the state-of-the-art band selection techniques and can work orders or magnitude faster, they deliver high-quality classification, and consistently identify significant bands in the training data, permitting the creation of refined and extremely compact sets that retain the most meaningful features. Also, the attention modules do not deteriorate the classification abilities, and slow down neither training nor inference of the deep models.

INDEX TERMS Attention mechanism, band selection, classification, convolutional neural network, deep learning.

I. INTRODUCTION

Current advancements in the sensor technology bring exciting possibilities in hyperspectral satellite imaging (HSI) which is being actively applied in various domains, including precision agriculture, surveillance, military, land cover applications [1], and more [2]–[4]. It captures a wide spectrum of light for each pixel [5]—such detailed information can be effectively exploited in HSI *classification* (assigning a class label to each pixel) and *segmentation* (determining the boundaries of objects of a given class in an input HSI) [6]. However, hyperspectral data's high dimensionality is an important challenge towards its efficient analysis, transfer, and storage. There are two dimensionality-reduction approaches for dealing with such noisy, almost always imbalanced, and often redundant data. *Feature extraction* algorithms generate new low-dimensional descriptors from hyperspectral images and elaborate low-dimensional embedding spaces, onto which the high-dimensional data is mapped [7], [8]. On the other hand, *feature selection* approaches retrieve a subset of all HSI bands carrying the most important information [9]. Although the former

The associate editor coordinating the review of this manuscript and approving it for publication was Qiangqiang Yuan.

approaches can be applied to reduced HSI sets, they are generally exploited to process raw HSI data, they are often computationally-expensive, can suffer from band noisiness, and may not be interpretable [10]. Band selection techniques are divided into *filter* (unsupervised) and *wrapper* (supervised) algorithms. Applied before classification, filter approaches do not require ground-truth data to select specific bands [11]–[13]. They, however, suffer from several drawbacks: (i) it is difficult to select the optimal dimensionality of the reduced feature space, (ii) band correlations are often disregarded, leading to the data redundancy [10], (iii) bands which might be informative when combined with others are removed, and (iv) noisy bands are often labeled as informative due to low correlation with other bands. Wrapper approaches use the classifier performance as the objective function for optimizing the subset of HSI bands [14], [15]. Although these methods decrease the memory requirements of the further HSI analysis, such algorithms induce serious computational overhead. In this work, we mitigate this problem, and incorporate the selection process into the deep network training. Such approaches have not been explored so far.

Deep learning (DL) has enabled unprecedented achievements and established the state of the art in a plethora of domains, including HSI analysis [16]. In general, the HSI classification and segmentation algorithms encompass *conventional* machine learning techniques which require feature engineering [17], and DL approaches [18]. DL can conveniently elaborate *spectral* features [19], or both *spectral* and *spatial* features without any user intervention. These features are intrinsically extracted by the deep nets operating on the full HSI. Therefore, we need to face the aforementioned challenges concerning the high HSI dimensionality in both conventional and DL-powered segmentation approaches. Attention mechanisms allow humans and animals to effectively process enormous amount of visual stimuli by focusing only on the most-informative chunks of data [20], [21]. An analogous approach can be applied in DL to localize the most informative parts of an input image to *focus* on. We build upon the *painless attention* mechanism which is trained during the network's forward-backward pass [22], and exploit it in our convolutional architectures for HSI band selection. To the best of our knowledge, attention mechanisms have been used neither for this purpose, nor for HSI classification/segmentation before, and our initial results on attention-based convolutional neural networks (CNNs) presented in this paper[1] laid foundations for other hyperspectral band weighting [25] and classification [26] techniques.

A. CONTRIBUTION

In this paper, we introduce a novel HSI band selection method (Section III) which exploits attention-based CNNs, and is applicable to any HSI data (i.e., with different numbers of bands). The goal of this system (Figure 1) is to learn which bands convey the most important information, as an outcome of the training process, alongside a ready-to-use trained deep convolutional model. It is in contrast to the wrapper band selection techniques which require training a supervised learner to assess the quality of selected subsets of bands during the optimization. Thus, our method is an *embedded* approach—the generation of attention heatmaps is embedded into the CNN training. These heatmaps quantify the *importance* of specific parts of the spectrum, and they are later processed using an anomaly detection algorithm. We build upon our observation that only a (very) small subset of all bands within an original HSI convey the important information (necessary to distinguish the underlying materials), and these bands can be seen as *outliers* (the other bands, which are in the majority, are not informative). The contribution of this work is multi-fold, and can be summarized in the following bullet points:

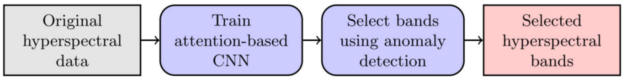
Original hyperspectral data
Train attention-based CNN
Select bands using anomaly detection
Selected hyperspectral bands

FIGURE 1. Flowchart of the proposed band selection algorithm. The bands from the input HSI are weighted according to their importance using the attention modules seamlessly incorporated into a convolutional deep network architecture, and the anomaly detection technique is later used to select those bands which convey the most important information about the underlying materials. Since the "important bands" are in the minority, they can be considered as "anomalous". A notable feature of our approach is that a fully-functional and ready-to-use deep model trained over the full hyperspectral data is an outcome of the process, alongside the set of selected bands.

- We introduce a new HSI band selection algorithm (Section III) which couples attention-based CNNs (Section III-B) and anomaly detection (Section III-C) to find the most important bands within HSI.
- We introduce new attention-based CNN architectures for extracting attention heatmaps that show which parts of the frequency spectrum are *important* for CNNs during their training, hence weight the bands according to their *importance*. Although the attention modules are inspired by [22] (slightly modified, as discussed in Section III-B), the deep architectures proposed in the work reported here are new and have been exploited neither for band selection nor HSI segmentation before. Our attention-based CNNs are *spectral* deep networks—they operate exclusively on the spectral information while classifying a pixel in an input HSI.
- We performed a rigorous experimental study over widely-used hyperspectral benchmarks to: (i) compare our technique with the state of the art in HSI band selection, (ii) verify the impact of band selection on various supervised learners (both conventional and DL), (iii) understand the impact of appending the attention modules to our CNNs on their classification abilities and training characteristics, and (iv) verify the statistical importance of the obtained results.

[1] Note that the initial results on our attention-based CNNs presented in this manuscript have been posted in arXiv back in October 2018. For more details, see our preprint [23] and the review paper by Sun and Du [24].

B. PAPER STRUCTURE

This paper is organized as follows. Section II reviews the current advances in automated HSI band selection. In Section III, which begins with a gentle introduction to the DL concepts used in this paper (Section III-A), we present our band selection algorithm. Experimental results are discussed in Section IV. Section V concludes the paper.

II. RELATED LITERATURE

The number of spectral bands in HSI can easily reach hundreds, and it has become a very useful source of information in various remote sensing applications. However, its huge volume brings challenges in efficient analysis, transfer (especially of HSI acquired on board of a satellite back to Earth is extremely time-consuming and costly), and storage of such imagery. Also, data redundancy is a serious practical issue—the neighboring bands are often correlated, therefore only their small subset contributes to the HSI classification process. Finally, generating ground-truth (manually-annotated) data for supervised classification and segmentation methods is extremely difficult, time-consuming, and prone to human errors, and exploiting small (in terms of the number of each-class examples) and very high-dimensional datasets can easily deteriorate the performance of supervised learners (this phenomenon is known as the *curse of dimensionality* [27]). To deal with these issues, HSI is subjected to either *feature extraction* (generating new, perhaps more informative, less redundant, and compressed features from HSI) or *feature selection* (determining a subset of all HSI bands which convey the most important and useful information).

Feature extraction (also referred to as feature learning) methods *extract* new features from the original HSI [28]. This process leads to obtaining a low-dimensional embedding space in which the physical meaning of specific bands is lost, and such features are not trivial to interpret. Additionally, the data noisiness can significantly affect the feature extraction process and lead to a low-quality reduced feature space [29]. The most popular feature extraction methods applied for hyperspectral data include principal component analysis [30] alongside its multiple modifications [31]–[34] (encompassing both linear and non-linear variants [35]), local Fisher's discriminant analysis [36] and local geometric structure Fisher's analysis [37], sparse-adaptive hypergraph discriminant analysis [38], independent component analysis-based techniques [39], tensor-based algorithms [40], singular spectrum analysis [29], manifold learning [41]–[43], sparse and low-rank linear embeddings [44], [45], wavelet-based feature extraction [46], [47], tensor locality alignment [48], various information fusion-enhanced techniques [49], hierarchical [50] and multiple [51] feature learning, saliency-guided unsupervised feature learning [52], non-parametric feature extraction [53], and more [54]. Also, there are techniques which combine multiple feature learning [55] and band selection [56]. For a more detailed review of the feature extraction methods for the multi- and hyperspectral satellite data, we refer to the recent survey by Setiyoko et al. [57].

Band selection methods are aimed at determining a (usually small) subset of all hyperspectral bands from HSI which convey the most useful information exploited during the classification and/or segmentation process. Such techniques can be divided into two groups, and they include *filter* (unsupervised) and *wrapper* (supervised) algorithms. Applied before classification, filter approaches do not require ground-truth data to select specific bands [58]. They can utilize ranking algorithms to score bands [11], [58], [59], sparse representations to weight them [60], evolutionary algorithms [12], and various clustering-based techniques [13], [61]–[68]. However, filtering approaches suffer from several drawbacks: (i) it is difficult to select the optimal dimensionality of the reduced feature space, (ii) band correlations are often disregarded, leading to the data redundancy—some methods exploit mutual band information and their (dis)similarity characteristics [10], [69]–[71], (iii) bands which might be informative when combined with others (but are not particularly useful on their own) are removed, and (iv) noisy bands are often labeled as informative due to low correlation with other bands [72]. There exist the algorithms which hybridize different approaches, e.g., clustering and ranking techniques [73]. In [74], the authors proposed a method which utilizes a fast and robust PCA on Laplacian graph to select bands from HSI and showed that their approach can outperform other techniques in terms of the classification accuracy and computational cost. The band selection problem was also tackled using kernel-based algorithms, including weighted kernel regularization [75].

Wrapper approaches use the classifier performance as the objective function for optimizing the subset of HSI bands [14], [15]. These methods encompass various heuristics and meta-heuristics, including biologically-inspired techniques [76]–[79], gravitational searches [80], and artificial immune systems [81]. Cao et al. proposed a semisupervised approach in which their exploited the edge preserved filtering to improve the pixel-wised classification maps (and to assess the quality of the selected band subsets) [82]. In [83], Zhang et al. introduced a multi-objective optimization model for selecting bands from HSI. It utilizes two (potentially contradicting) objective functions: the amount of preserved information, and the redundancy within the selected bands. This technique utilized an immune system to effectively balance the exploration and exploitation of the solution space (the experiments showed that the immune system is able to outperform other multi-objective algorithms, including a fast and elitist multi-objective genetic algorithm [84]). Although the wrapper methods alleviate the computational burden of the HSI analysis, such algorithms induce serious overhead, especially in the case of classifiers which are time-consuming to train (e.g., deep neural nets [85]). In this work, we mitigate this problem, and incorporate the selection process into the training of our attention-based convolutional neural network (we propose an *embedded* band selection algorithm).

To the best of our knowledge, such approaches have not been explored in the literature so far. For a detailed and insightful review of the current hyperspectral band selection techniques, we refer to an excellent survey by Sun and Du [24].

III. METHOD

This section is started with a brief introduction to the basic concepts (convolutions and pooling) in CNNs (Section III-A). These building components are exploited in our attention-based CNNs which produce the attention heatmaps (Section III-B). The heatmaps are finally subjected to the anomaly detection using the Elliptical Envelope algorithm (Section III-C). The important bands in an input HSI are treated as the *anomaly* in the data—the majority of the bands convey very similar information about the underlying material, hence can be safely discarded without deteriorating the classification performance, and only a small fraction of bands (the minority of them) are "informative".

A. A GENTLE INTRODUCTION TO CONVOLUTIONAL NEURAL NETWORKS

CNNs have been tremendously successful in solving various computer vision and pattern recognition tasks, and have established the state of the art in many subfields of science and engineering [16]. Goodfellow *et al.* [86] defined CNNs as the neural networks which use *convolution* instead of a general matrix multiplication in at least one of their layers. There are two types of layers in CNNs: *convolutional* and *pooling* ones (which stacked together form the *feature extraction* part of a CNN). The feature extractor is then followed by the classification part of the deep network, commonly built with *fully-connected* layers. The convolutional and pooling layers are discussed in more detail in the following sections.

1) CONVOLUTIONAL LAYERS

This type of the layer is an essential building block of CNNs. Convolutional layers expose three important ideas which help improve machine learning engines—parameter sharing, equivariant representations, and sparse interactions [86]. A convolutional layer encompasses n trainable *kernels* of size k which are convolved (with a stride s) with an input signal to generate n *feature maps*. The input signal and the kernels are represented as multi-dimensional data arrays (*tensors*). In this work, we focus on one-dimensional signals—each pixel in an input HSI is a 1D tensor of values (for specific wavelengths), and the kernels are also one-dimensional. Therefore, if an input 1D signal I is convolved with the kernel K (this operation is denoted by the $*$ sign), we have the *i-th* entry in the output tensor $O(i)$ given as (to make it mathematically correct, we *flip* the kernel—note that when the index in the input increases, the corresponding index in the kernel decreases; without kernel flipping, it would be cross-correlation):

$$O(i) = (I * K)(i) = \sum_j I(j) \cdot K(i-j). \tag{1}$$

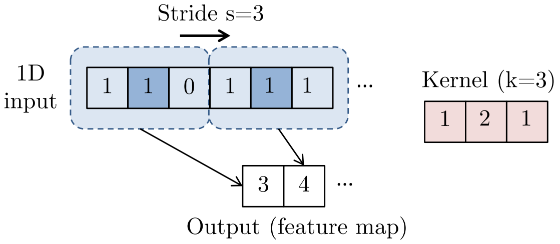

FIGURE 2. Example of an 1D signal (rendered in blue) convolved (stride $s = 3$) with a kernel of size $k = 3$ (in red). The positions in which the kernel is centered during the convolution process are rendered in dark blue.

In Figure 2, we render an illustrative example of convolving a kernel (of size $k = 3$) with an input 1D signal. Here, we restrict the kernel positions only to those in which it is entirely positioned within the input signal (otherwise, we would have to *pad* the "border" of the input). The kernel is sliding with a stride $s = 3$. Hence, it is first centered at the second position in the input tensor, then—at the fifth position, and so forth (these positions are rendered in dark blue in Figure 2). Therefore, the first entry in the output feature map becomes $1 \cdot 1 + 2 \cdot 1 + 1 \cdot 0 = 3$, and the second entry: $1 \cdot 1 + 2 \cdot 1 + 1 \cdot 1 = 4$. The size of the feature map O_s depends on the kernel size, size of the input I_s, padding p and stride s, and can be calculated as $O_s = \lfloor (I_s - k + 2p)/s \rfloor + 1$. In our example (Figure 2), assuming that $I_s = 100$, we would have the 1D output feature map of the $O_s = \lfloor (100 - 3 + 2 \cdot 0)/3 \rfloor + 1 = 33$ size. Convolutions can be applied for inputs of variable size.

2) POOLING LAYERS

The aim of pooling layers is to replace the outputs of certain parts of the deep network by a summary statistics of the neighboring outputs, in order to ensure the *representation invariance* with respect to small translations of the input [86]. This invariance means that if the input is slightly translated, the pooled output does not change. A pooling layer effectively reduces the size of its input (thus performs *downsampling*)—the size of the output is $O_s = (I_s - k)/s + 1$, where I_s is the input size, k denotes the pooling window size, and s denotes the stride. Additionally, pooling can be perceived as a regularizer which helps avoid overfitting.

As in convolutional layers, the pooling layer utilizes a sliding kernel of size k (stride s) which is moved across the input. In Figure 3, we present an example of two most popular pooling operations applied to the same input—the average (yellow) and maximum (orange). We set $k = 3$ and $s = 3$, and the positions in the input tensor where the pooling window is centered are rendered in dark blue.

B. ATTENTION-BASED CNNs

We introduce attention-based CNNs for extracting attention heatmaps from HSI. These CNNs exclusively exploit the spectral information acquired for each pixel in an input HSI (pixels are processed separately—we *do not* utilize any spatial information concerning the pixels' neighborhood in HSI, therefore our deep networks are the spectral CNNs).

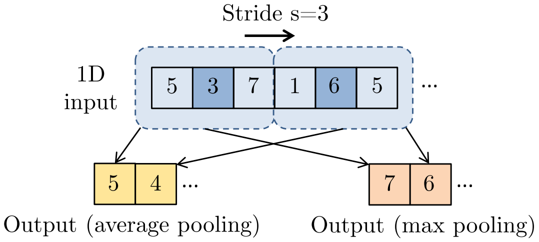

FIGURE 3. Example of the 1D signal (rendered in blue) pooled (stride $s = 3$) with the average and maximum pooling operations (the size of the pooling window is $k = 3$). The positions in which the kernel is centered during the convolution process are rendered in dark blue.

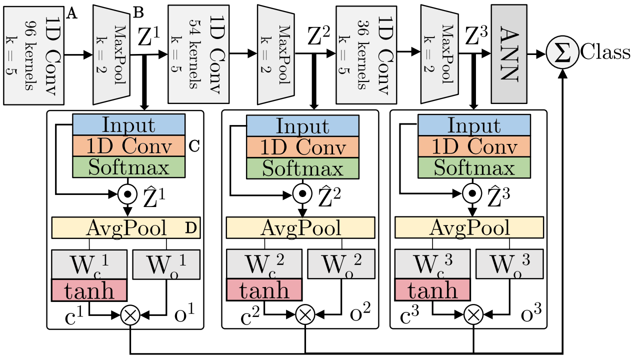

FIGURE 4. In attention-based CNNs, features at different levels Z^l are processed to generate the attention heatmaps, and they are used to output (i) a class hypothesis based on the local information, and (ii) a confidence score c^l. The final output is the softmaxed weighted sum of the attention estimators, and the output of the network's classifier (here, an artificial neural network, ANN containing two hidden layers containing 512 and 128 neurons with ReLU).

However, they could be potentially extended by incorporating the convolutional layers which would operate in the spatial HSI dimension (the network would be a spatial-spectral CNN), because the attention modules are topology-agnostic.

Our attention-based CNNs are inspired by the recent paper by Rodriguez *et al.* [22], where the authors proposed a new attention mechanism that can be seamlessly incorporated into any convolutional architecture (without any extra supervision, as no additional class labels are exploited), and applied it in the context of fine-grained object recognition in digital images. In contrast to [22], we do not modify the network loss during the training process—Rodriguez et al. introduced an additional regularization loss that forces the multiple attention heatmaps to be different from each other. This approach may be useful in image analysis where fine-grained object features might be manifested in various parts of the input image. Here, we do not intend to push these heatmaps towards orthogonality (multiple heatmaps which are "similar" may better highlight the most important frequencies in the spectrum, hence more "confidently" distinguish the most informative bands from all others). Also, we avoid inferring with standard loss functions to make the attention modules straightforwardly applicable in CNNs—modifying a loss requires performing the sensitivity analysis to properly tune the weight of the regularization term which quantifies its impact on the loss function.

1) GENERAL OVERVIEW OF THE DEEP NETWORK ARCHITECTURE

In the attention-based CNNs for HSI (Figure 4), an attention module is inserted after each max-pooled activation of a convolutional layer Z^l (l denotes the depth within the network topology, and $l \geq 1$), in order to reduce the computational burden of the attention mechanism. This module is composed of two elements: an *attention estimator*, extracting the most important regions of a feature map, and a *confidence gate*, producing a confidence score for the prediction (these elements are discussed in detail in the following sections). We can easily modify the number of building blocks (BBs) in our CNNs—each BB encompasses the one-dimensional (1D) convolution followed by the non-linearity, batch normalization, and 1D max pooling layer (we operate only in the spectral dimension, hence both types of the layers are one-dimensional), alongside the attached attention module. We exploit the rectified linear unit (ReLU) as a non-linearity, which outputs zero for any negative input x, and it returns the value of x otherwise. Hence, it can be formally written as $ReLU(x) = \max(0, x)$. In this work, we experimentally analyzed the attention-based CNNs with two, three, and four BBs (Section IV). To the best of our knowledge, such CNN architectures have been used neither for band selection from HSI, nor for classification or segmentation of such imagery.

Each layer in our CNN[2] is parameterized with the corresponding hyper-parameter values: the number of kernels n, together with the size of the kernels in the convolutional layers, and the size of the pooling windows (both sizes are denoted as k in Figure 4 for brevity). The kernel size is kept constant for all convolutional layers ($k = 5$, unit stride, and the padding of 2). The pooling window size was kept unchanged ($k = 2$, stride of 2) in all layers. The number of kernels n in the consecutive convolutional layers decreases with the increase of the number of BBs (Figure 4) in order to reduce the computational complexity of the model, alongside its memory footprint. We expect that the shallower convolutional layers will be able to extract low-level discriminant deep features based on the spectral information, as they should be manifested in specific parts of the spectrum.

In Table 1, we present the dimensionality of the input and output tensors for the selected operations in our deep architecture (annotated as A, B, C, and D in Figure 4; b is the number of bands in the input HSI). The details of the C and D steps (in the attention module) are discussed below.

2) ATTENTION ESTIMATOR

The attention estimator module encompasses the 1D convolution with one kernel applied (therefore, $n = 1$; in [22], the number of kernels is greater). The kernel size is equal to the number of feature maps extracted by the corresponding

[2] The attention hyper-parameters are discussed later.

TABLE 1. Data input and output dimensionality of selected steps in our attention-based CNNs (annotated as A, B, C, and D in Figure 4).

Step	Input	Output
A	$b \times 1$	$b \times 96$
B	$b \times 96$	$(\lfloor (b-2)/2 \rfloor + 1) \times 96$
C	$(\lfloor (b-2)/2 \rfloor + 1) \times 96$	$(\lfloor (b-2)/2 \rfloor + 1) \times 1$
D	$(\lfloor (b-2)/2 \rfloor + 1) \times 96$	$(\lfloor (b-2)/2 \rfloor + 1) \times 1$

BB. This kernel performs the dimensionality reduction (all feature maps are reduced to one using 1D convolution), and it is followed by the ReLU activation and spatial (across all entries within the feature map) softmax to elaborate the final attention heatmap (note that we do not softmax separate confidence scores, as suggested in [22], to decrease the computational burden of attention).

The attention estimator learns the following embedding:

$$\mathcal{F} : \mathbb{R}^{b \times n} \rightarrow \mathbb{R}^{b \times 1}, \tag{2}$$

where b and n denote the number of HSI bands and the number of feature maps, respectively. The attention estimator effectively merges all feature maps (FMs) at depth l into a single one (hence, the dimensionality reduction is performed). The estimator builds an attention heatmap $\hat{Z}^l$—it is used to normalize each activation map in Z^l, which denotes the set of all activation maps at the level l (i.e., the attention heatmap is exploited to highlight the importance of each entry in each activation map). The hypothesis H^l of the output space given its local information is finally produced:

$$H^l = AvgPool(\hat{Z}^l \odot Z^l), \tag{3}$$

where the $\odot$ sign represents the element-wise (Hadamard) product. Note that the number of activation maps Z^l at a given level l is variable (i.e., 96 after the first BB, 54 after the second BB, and 36 after the third BB), and this normalization is executed to each of them—they are average-pooled to produce the hypothesis H^l. It is later exploited by a linear classifier to predict the label of the input sample:

$$o^l = H^l W_o^l. \tag{4}$$

The most computationally intensive part of the attention estimator is the 1D convolution. In general, the time complexity of a 1D convolutional layer amounts to $\mathcal{O}(k \cdot b \cdot n)$, assuming a single input channel to this layer, and k denotes the kernel size, b is the size of the input tensor, and n is the number of kernels in this layer [87], [88]. Importantly, we exploit just a single kernel in the C step (Figure 4). Then, the computational complexity of the Hadamard product alongside the pooling layer is of $\mathcal{O}(b' \cdot n)$, where n is the number of feature maps in this context, and b' is the input size.

3) CONFIDENCE GATE

The local features are very often not enough to output a high-quality class hypothesis. Thus, we couple each attention module with the network's output to predict the confidence score c by the means of an inner product with the gate weight matrix W_c (at the l-th level):

$$c^l = \tanh(H^l W_c^l). \tag{5}$$

The final output of the network is the softmaxed weighted sum of the attention estimators and the output of the classifier o^{net} multiplied by its confidence score c^{net}:

$$output = softmax(o^{net} \cdot c^{net} + \sum_{l=1}^{||BB||} c^l \cdot o^l). \tag{6}$$

The softmax function (which can be calculated in linear time) converts a real-valued score x (e.g., the network output) into a probability value p in the multi-class classification. Thus, a vector of such scores $\boldsymbol{x} \in \mathbb{R}^C$ is converted into a vector of probabilities $\boldsymbol{p} \in [0, 1]^C$, where p_i is the probability of an input pixel HSI belonging to the i-th (out of C) class, and it is given as:

$$p_i = \frac{e^{x_i}}{\sum_{k=0}^{C-1} e^{x_k}}. \tag{7}$$

The output vector $\boldsymbol{p}$ is the probability vector, therefore it is non-negative, and $\sum_{c=0}^{C-1} p_c = 1$, and can be used to predict the final class label for each incoming HSI pixel.

4) ILLUSTRATIVE ATTENTION-BASED CNN EXAMPLE

To better understand the influence of the data dimensionality reduction in the deeper parts of our attention-based CNNs (with three BBs), we render (Figure 5) an example process of extracting attention heatmaps from an input HSI (with $b = 103$ bands). Let us assume that the fifth entry in each example feature map is informative, and should be selected by the attention mechanism. The final attention heatmap should return the attention score for each band in the input HSI, hence the attention heatmaps extracted in the deeper parts of the CNN architecture are interpolated (i.e., their dimensionality is increased, as presented in Figure 5, where we render the heatmaps with the size corresponding to b). We can appreciate that the attention scores gradually decrease in the larger (deeper) neighborhoods of the fifth entry in each heatmap (the attention mechanism is applied over the feature maps, and the receptive field within the CNN is larger in the deeper parts of the architecture). Eventually, we average all the heatmaps to elaborate the final attention heatmap which is subjected to the anomaly detection (Section III-C). The anomaly detection approach can be counter-intuitive (we are often interested in removing outliers from the data in machine learning applications), however these "outliers" represent the most informative HSI bands in this context—the remaining bands are annotated with low attention scores, hence are considered less useful during the classification.

C. SELECTION OF HSI BANDS AS ANOMALY DETECTION

In this work, we exploit an Elliptical Envelope (EE) algorithm to extract the most important (discriminative) bands from the input (full) HSI based on the final attention heatmap

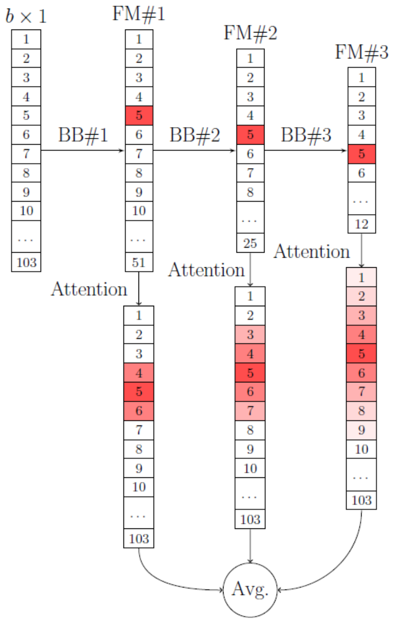

FIGURE 5. Extracting the attention heatmaps from an example HSI (with $b = 103$ bands) using our attention-based CNN (with three BBs). We visualize the influence of the dimensionality reduction in the feature maps (FMs) extracted in the deeper parts of our attention-based CNN on the attention heatmaps: the fifth entry in each FM is considered *informative*, and the corresponding attention heatmap is rendered below—the darker red the cell is, the higher attention value is obtained. The final attention heatmap is obtained by averaging the heatmaps extracted at all levels in the deep architecture.

(Section III-B.2). Since the number of such important bands should be low, they can be understood as an *anomaly* in the input (full) set (see an example of this phenomenon rendered in Figure 5; also we present the attention heatmaps extracted for real benchmark datasets which manifest the same feature of the attention heatmaps later in the paper, see the experimental Section IV-C, more specifically Figure 7). In EE, the data is modeled as a high-dimensional Gaussian distribution with covariances between feature dimensions (here, the entries of the final attention heatmap extracted using an attention-based CNN, as presented in Figure 5; thus, the input tensor to the EE algorithm is of the $b \times 1$ size), and an ellipse which covers the majority of the data is determined. These samples which lay outside of this ellipse are classified as *anomalous* [89]. EE utilizes a fast algorithm[3] for the minimum covariance determinant estimator [90], where the data is divided into non-overlapping sub-samples for which the mean (μ) and the covariance matrix in each feature dimension ($\boldsymbol{C}$) are calculated. Finally, the Mahalanobis distance $\boldsymbol{D}$ is extracted for each sample $\boldsymbol{x}$:

$$\boldsymbol{D} = \sqrt{(\boldsymbol{x} - \mu)^T \boldsymbol{C}^{-1} (\boldsymbol{x} - \mu)}, \tag{8}$$

and the samples with the smallest values of $\boldsymbol{D}$ are retained. In EE, the fractional contamination rate (λ) defines how much data in the analyzed dataset should be selected as anomalies (hence, should not lay within the final ellipse). These data samples (i.e., spectral bands) are selected as *important* in our technique—they are assigned significantly larger attention values in the heatmap compared with all other bands.

IV. EXPERIMENTS

A. EXPERIMENTAL SETUP

In all experiments, we perform Monte-Carlo cross-validation and divide each HSI dataset (Section IV-B) 30 times into balanced training ($\boldsymbol{T}$) and validation ($\boldsymbol{V}$) sets (we perform under-sampling of the majority classes and ignore background pixels), and the unseen test sets (Ψ). We randomly sample pixels (*not* patches, as it would be done in the case of spatial-spectral segmentation algorithms) from the input HSI, and put them into $\boldsymbol{T}$, $\boldsymbol{V}$ or Ψ. These sets encompass 80%, 10%, and 10% of all pixels in the HSI, respectively, and they never overlap—since we analyze only spectral segmentation algorithms here (i.e., utilizing exclusively the spectral information while classifying a pixel), this training-validation-test division does not cause a training-test information leak (which would have appeared if we had analyzed spatial-spectral approaches over this data split, and it would ultimately lead to over-optimistic conclusions on the classification performance of such methods that exploit the spatial neighborhood information about a pixel being classified, as shown in our recent paper [6]). The $\boldsymbol{T}$ and $\boldsymbol{V}$ sets are used during the CNN training, whereas Ψ is utilized to quantify the generalization of the trained models. We report per-class and average (AA) accuracy, and the values of the Cohen's kappa coefficient[4] given as $\kappa = 1 - \frac{1-p_o}{1-p_e}$, where p_o and p_e are the observed and expected agreement (assigned vs. correct class label), respectively, and $-1 \leq \kappa \leq 1$. Interestingly, there is no single "correct" interpretation of kappa values, and even a score as low as 0.4 might be acceptable in certain applications [91]. However, the larger the kappa score becomes, the better is the performance of the classifier. All the measures reported in this paper are averaged across all 30 runs for each investigated setup.

Our CNNs were implemented in `Python 3.6` with `PyTorch 0.4`—we made the implementation publicly available at https://github.com/ESA-PhiLab/hypernet/tree/master/python_research/experiments/hsi_attention. The CNN training (ADAM optimizer [92] with the default parametrization: learning rate of 0.001, $\beta_1 = 0.9$, and $\beta_2 = 0.999$) terminates if after 25 consecutive epochs the accuracy over $\boldsymbol{V}$ does not increase.

B. DATASETS

We focused on two multi-class HSI benchmarks: Salinas Valley (acquired using the NASA Airborne Visible/Infrared Imaging Spectrometer AVIRIS sensor), and Pavia University (Reflective Optics System Imaging Spectrometer ROSIS sensor). AVIRIS registers 224 contiguous bands with

[3] For the detailed analysis of its time complexity, see [90].

[4] The kappa coefficient shows us how much better is the analyzed classifier than a random one which guesses the label based on the data distribution.

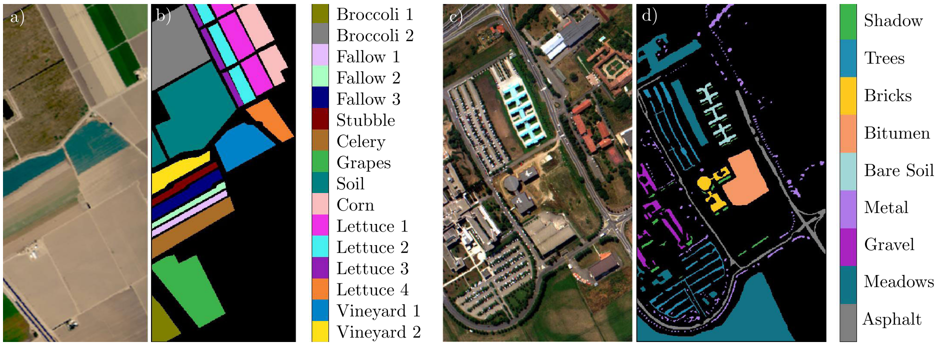

FIGURE 6. Benchmark HSI: a) Salinas Valley scene, b) Salinas Valley ground-truth, c) Pavia University scene, and d) Pavia University ground-truth.

TABLE 2. The number of examples from each Salinas-Valley class.

Class	Description	Examples#
1	Broccoli green weeds 1	2,009
2	Broccoli green weeds 2	3,726
3	Fallow	1,976
4	Fallow rough plow	1,394
5	Fallow smooth	2,678
6	Stubble	3,959
7	Celery	3,579
8	Grapes untrained	11,271
9	Soil vineyard green weeds	6,203
10	Corn senescent green weeds	3,278
11	Lettuce romaine 4 week	1,068
12	Lettuce romaine 5 week	1,927
13	Lettuce romaine 6 week	916
14	Lettuce romaine 7 week	1,070
15	Vineyard untrained	7,268
16	Vineyard vertical trellis	1,807
—	Total	54,129

TABLE 3. The number of examples from each Pavia-University class.

Class	Description	Examples#
1	Asphalt	6,631
2	Meadows	18,649
3	Gravel	2,099
4	Trees	3,064
5	Painted metal sheets	1,345
6	Bare soil	5,029
7	Bitumen	1,330
8	Self-blocking bricks	3,682
9	Shadows	947
—	Total	42,776

wavelengths in a 400 to 2450 nm range (visible to near-infrared), with 10 nm bandwidth, and it is calibrated to within 1 nm. ROSIS collects the spectral radiance data in 115 bands in a 430 to 850 nm range (4 nm nominal bandwidth). Both sets are imbalanced (Tables 2–3), and contain under-represented classes, e.g., class 13 (C13) in Salinas Valley or class 9 (C9) in Pavia University.

1) SALINAS VALLEY

This set (217 × 512 pixels) was captured over Salinas Valley in California, USA, with a spatial resolution of 3.7 m. The image shows different sorts of vegetation, corresponding to 16 classes (Figure 6a–b). The original data contains 224 bands, however 20 bands were removed by the authors of this set due to either atmospheric absorption or noise contamination [93] (204 bands remained[5]).

2) PAVIA UNIVERSITY

This set (340×610 pixels) was captured over Pavia University in Lombardy, Italy, with a spatial resolution of 1.3 m. It shows an urban scenery with nine classes (Figure 6c–d). The set contains 103 bands, as 12 most noisy bands (out of 115) were removed by its authors.

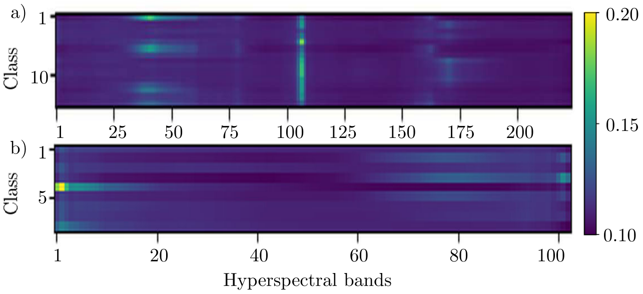

FIGURE 7. Example average attention-score heatmaps for a) Salinas Valley and b) Pavia University show that certain bands convey more information than the others (the brighter the regions are, the higher attention scores were obtained).

C. SELECTION OF BANDS USING THE PROPOSED ALGORITHM

In this experiment, we extracted bands from the benchmark HSI using our attention-based CNNs followed by the anomaly detection. For each dataset, we ran CNNs equipped with two, three, and four BBs (referred to as CNN-2A, CNN-3A, and CNN-4A) 30 times using Monte-Carlo cross-validation, and the attention scores (which were fairly consistent for all runs; $p < 0.01$, two-tailed Wilcoxon tests) were averaged across all executions and CNN architectures (example scores are visualized as heatmaps in Figure 7). Given the average attention scores, the Elliptic Envelope algorithm with different values of the contamination rate

[5] See details at: http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes; last access: January 8, 2020.

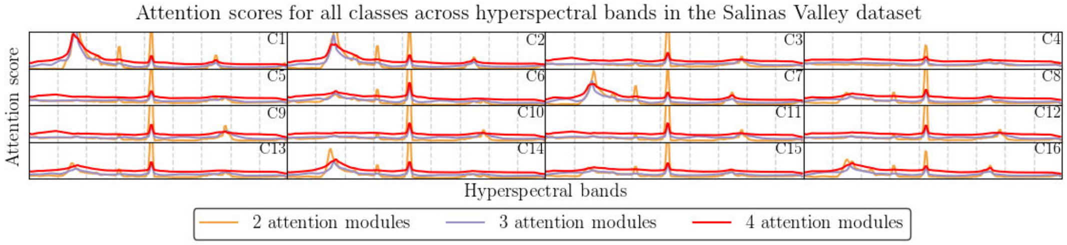

FIGURE 8. Averaged attention scores for the Salinas Valley dataset.

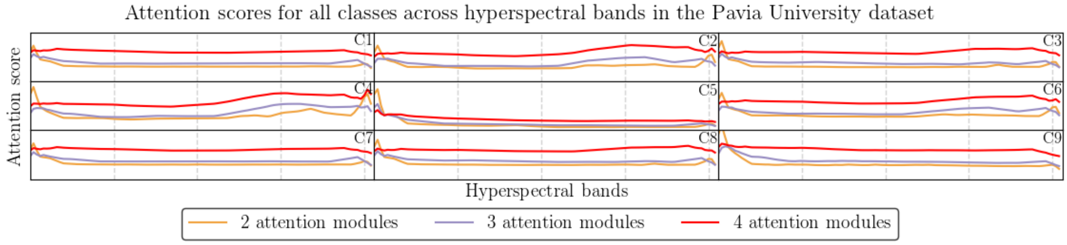

FIGURE 9. Averaged attention scores for the Pavia University dataset.

TABLE 4. Number of bands selected using the proposed algorithm for the a) Salinas Valley and b) Pavia University datasets.

	Contamination rate (λ) →	0.01	0.02	0.03	0.04	0.05
a)	Number of selected bands	28	28	29	33	38
	Percentage of all bands	13.73	13.73	14.22	16.18	18.63
b)	Number of selected bands	9	12	14	20	28
	Percentage of all bands	8.74	11.65	13.59	19.42	27.18

$\lambda = \{0.01, 0.02, \ldots, 0.05\}$ (the lower λ is, the smaller number of bands will not be encompassed by an elliptical envelope and will be annotated as "anomalous", hence carrying important information) was used to extract the final subset of HSI bands. The band selection results are gathered in Table 4. Although the contamination rate is a hyper-parameter of our method and it should be determined *a priori*, the differences (in terms of the number of selected bands) across different λ values are not very large (note that for Salinas Valley, the number of bands extracted for $\lambda = 0.01$ and $\lambda = 0.02$ were equal). Very small λ values can be used to further lower the number of HSI bands if necessary (e.g., in hardware-constrained environments and/or to compress HSI before transferring it back to Earth from the satellite). Our technique drastically decreased the number of HSI bands for all datasets, and for all λ's: less than 14% and 9% of bands were selected as important for $\lambda = 0.01$ for Salinas and Pavia, which amounts to 28 and only 9 bands, respectively.

The average attention scores for the Salinas Valley and Pavia University datasets are visualized in detail (for each class and for each CNN separately) in Figures 8–9. There exist several attention peaks for Salinas Valley indicating the most meaningful part of the spectrum that is used to distinguish between pixels of all classes (see the highest peak in the middle of the spectrum). Although for Pavia University there are less such clearly selected bands, some parts of the spectrum are definitely more distinctive than the others (see both ends of the spectrum in the second row of Figure 7). This experiment showed that our CNNs (with various numbers of BBs) followed by the anomaly detection retrieve very consistent attention scores annotating the most important bands, and that our approach is data-driven (it can be easily applied to any new HSI dataset).

D. INFLUENCE OF ATTENTION MODULES ON CLASSIFICATION

This experiment verifies if applying attention modules in a CNN has any (positive or negative) impact on its performance and convergence of the training process. For each set, we trained the deep networks with and without attention using original HSI data (without band selection). The CNNs with the attention modules are referred to as CNN-2A, CNN-3A, and CNN-4A (for two, three, and four BBs, respectively), whereas those which are not accompanied with them include CNN-2, CNN-3, and CNN-4 (two, three, and four convolutional-pooling blocks, as depicted in Figure 4).

The average per-class accuracy scores (averaged across 30 executions) for Salinas and Pavia are gathered in Tables 5 and 6. The differences between the architectures are not statistically important (i.e., CNN-2 compared with CNN-2A, CNN-3 with CNN-3A, and CNN-4 with CNN-4A), according to the Wilcoxon tests at $p < 0.01$. Therefore, attention modules did not adversely impact the performance of the CNNs—they allow for building a high-quality model and selecting the most important bands *at once*. Deeper CNNs delivered more stable results (std. dev. of the accuracy over Ψ decreased from 0.007 to 0.005 for Salinas, and from 0.03 to 0.01 for Pavia).

TABLE 5. Classification accuracy (in %) of various models obtained for the full and reduced Salinas Valley dataset (we report the number of bands and the contamination rate in parentheses; "Full" for no reduction). The best results in each column and for each classifier (collectively for CNNs with and without attention) are boldfaced, whereas the worst results are grayed.

Algorithm	Bands	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	AA
CNN-2	204 (Full)	**99.16**	**99.34**	**97.91**	99.34	96.12	99.63	**99.67**	**73.22**	**98.97**	**92.05**	**95.82**	99.71	98.10	95.27	70.88	98.79	**94.62**
CNN-2A	204 (Full)	98.97	98.72	96.85	99.60	**97.51**	99.52	**99.67**	72.89	98.90	91.03	94.98	99.49	97.95	95.20	69.60	**99.01**	94.37
CNN-2	38 (0.05)	98.68	98.94	91.65	99.60	91.79	99.52	99.60	63.81	97.36	88.68	90.37	99.78	97.95	95.97	68.06	98.10	92.49
CNN-2A	38 (0.05)	98.53	98.17	92.27	99.56	91.83	99.41	99.38	65.79	97.40	86.19	91.50	99.78	98.06	96.45	66.85	98.06	92.45
CNN-2	33 (0.04)	98.24	98.57	92.38	99.49	92.45	99.41	99.38	61.72	97.80	87.73	91.50	99.74	97.99	95.71	69.12	98.42	92.48
CNN-2A	33 (0.04)	98.50	98.86	91.03	**99.63**	94.95	**99.85**	99.41	66.26	97.33	87.29	93.92	99.78	**98.13**	**96.52**	67.77	98.72	93.00
CNN-2	29 (0.03)	98.83	98.46	91.14	99.52	93.08	99.45	99.41	66.08	97.95	88.13	90.33	99.71	97.80	95.90	68.57	98.90	92.70
CNN-2A	29 (0.03)	98.86	98.86	93.44	99.45	92.12	99.63	99.63	66.52	97.77	87.88	92.86	99.89	97.95	96.34	67.18	98.57	92.93
CNN-2	28 (0.02/1)	98.24	98.61	92.01	99.45	94.21	**99.85**	99.41	61.76	97.36	86.67	92.16	99.78	97.80	95.82	**71.72**	98.46	92.71
CNN-2A	28 (0.02/1)	98.86	98.46	91.14	**99.63**	91.90	99.56	99.41	69.27	97.66	85.82	92.16	**99.93**	97.95	96.12	65.20	98.79	92.62
CNN-3	204 (Full)	**99.05**	99.27	96.74	99.16	**97.14**	**99.74**	99.67	70.95	98.68	92.67	**96.56**	**99.85**	**98.68**	96.23	**73.99**	**99.16**	94.85
CNN-3A	204 (Full)	98.64	**99.49**	**97.84**	99.23	96.12	99.56	99.45	**76.34**	**98.79**	**93.33**	96.34	99.82	97.77	96.74	69.85	98.75	**94.88**
CNN-3	38 (0.05)	98.42	98.75	92.09	**99.71**	93.41	**99.74**	99.56	64.54	96.81	88.83	92.60	99.74	98.28	**96.92**	69.85	98.39	92.98
CNN-3A	38 (0.05)	98.10	97.44	88.17	99.41	91.21	99.63	99.41	65.13	97.80	89.12	92.20	99.41	97.66	96.26	66.78	98.02	92.23
CNN-3	33 (0.04)	97.69	98.79	93.19	99.45	91.68	99.52	**99.78**	65.53	97.36	89.56	92.34	99.82	97.77	96.08	68.39	98.97	92.87
CNN-3A	33 (0.04)	97.77	97.84	89.93	99.30	91.98	99.56	99.27	66.45	96.70	89.89	91.54	99.78	97.95	96.23	66.63	98.94	92.48
CNN-3	29 (0.03)	98.46	98.02	93.63	99.56	91.21	99.60	99.52	65.68	97.22	90.59	89.45	99.78	97.91	95.86	67.73	98.61	92.68
CNN-3A	29 (0.03)	98.13	98.35	90.40	99.56	93.22	99.56	99.34	69.12	97.22	88.75	91.87	99.71	98.24	96.26	63.37	98.97	92.63
CNN-3	28 (0.02/1)	98.10	98.13	91.90	99.56	92.49	99.63	99.63	64.95	97.73	87.69	91.50	99.82	97.25	96.04	67.73	98.86	92.56
CNN-3A	28 (0.02/1)	97.66	98.06	92.31	99.41	90.55	99.41	99.60	66.56	96.78	88.50	91.72	99.82	97.95	96.08	66.85	98.83	92.50
CNN-4	204 (Full)	98.75	**99.05**	**98.06**	99.45	94.51	**99.74**	99.52	68.39	**98.79**	92.38	96.26	99.78	98.17	93.99	**73.63**	98.79	94.33
CNN-4A	204 (Full)	**99.45**	99.01	97.99	99.30	**96.56**	99.67	**99.82**	**72.67**	**98.79**	**92.53**	**97.51**	**99.93**	98.39	97.03	70.95	98.53	**94.88**
CNN-4	38 (0.05)	98.39	97.40	92.09	**99.60**	91.61	99.30	99.52	66.92	97.47	88.83	91.98	99.78	97.40	96.92	66.04	98.72	92.62
CNN-4A	38 (0.05)	97.25	98.83	91.54	99.41	92.09	99.34	99.34	64.76	97.22	87.07	90.62	99.82	97.99	96.08	66.45	**99.01**	92.30
CNN-4	33 (0.04)	97.84	97.25	92.38	99.49	92.38	99.45	98.97	65.53	97.33	87.47	92.97	99.78	98.35	96.56	64.95	98.86	92.47
CNN-4A	33 (0.04)	97.47	98.32	91.58	99.38	92.60	99.49	99.34	65.02	97.29	88.21	91.32	99.71	98.17	96.78	66.48	98.64	92.49
CNN-4	29 (0.03)	98.61	97.88	90.29	99.45	92.12	99.34	99.23	68.39	96.19	87.47	90.81	99.74	98.24	96.23	64.87	98.50	92.34
CNN-4A	29 (0.03)	97.55	97.95	88.68	99.45	92.86	99.56	99.38	61.65	97.66	88.68	90.51	99.78	97.99	**97.14**	68.79	98.32	92.25
CNN-4	28 (0.02/1)	98.57	97.77	92.38	99.41	90.44	99.71	99.41	64.03	96.19	87.84	89.74	99.74	**98.50**	97.11	65.68	98.97	92.22
CNN-4A	28 (0.02/1)	98.75	97.80	89.30	99.56	92.01	99.63	99.16	64.25	97.11	87.55	91.06	99.12	98.13	95.90	65.53	98.28	92.07
SVM	204 (Full)	**99.93**	**99.89**	**99.93**	99.41	**99.74**	99.82	**99.74**	**84.21**	**99.89**	**97.25**	**99.93**	99.96	**99.74**	**99.38**	**81.03**	99.38	**97.45**
SVM	38 (0.05)	99.52	99.63	98.57	99.67	97.66	99.67	99.34	79.41	99.67	94.18	98.90	**100.0**	99.41	99.30	77.51	**99.56**	96.38
SVM	33 (0.04)	99.67	99.63	98.28	99.63	97.55	**99.89**	99.71	76.19	99.49	93.30	99.05	99.89	99.71	**99.38**	75.27	99.34	96.00
SVM	29 (0.03)	99.67	99.49	98.57	**99.71**	96.85	99.67	99.52	73.33	99.45	94.84	99.23	99.82	99.52	99.27	75.38	99.27	95.85
SVM	28 (0.02/1)	99.41	99.63	98.13	99.63	97.69	99.71	99.67	75.24	99.56	94.21	98.94	99.85	99.63	98.75	74.62	99.30	95.87
RF	204 (Full)	**99.67**	**99.96**	**99.63**	99.63	**98.79**	99.71	**99.52**	**76.41**	**99.30**	**94.54**	**98.79**	99.74	**99.30**	**98.64**	**75.20**	**99.12**	**96.12**
RF	38 (0.05)	99.52	99.16	98.32	**99.78**	97.36	**99.85**	**99.52**	73.00	98.32	91.72	97.47	99.71	98.72	97.77	71.72	99.01	95.06
RF	33 (0.04)	99.60	99.12	98.42	99.34	96.67	99.74	99.05	70.26	98.68	91.50	96.74	99.82	98.46	97.03	70.18	98.94	94.60
RF	29 (0.03)	99.30	99.19	97.25	99.38	96.41	**99.85**	99.49	71.68	98.53	91.98	96.12	99.67	98.28	97.22	70.26	99.08	94.61
RF	28 (0.02/1)	99.41	99.34	97.91	99.52	96.96	99.45	99.49	71.61	98.53	91.65	95.97	**99.85**	98.10	97.47	70.15	98.90	94.65
DT	204 (Full)	**99.63**	**99.01**	**97.80**	99.30	**97.51**	**99.38**	**99.52**	**66.70**	**98.02**	**91.03**	**96.34**	97.69	97.95	**96.15**	**65.49**	98.10	**93.73**
DT	38 (0.05)	99.23	98.57	95.27	98.79	96.12	99.34	99.05	64.21	96.19	85.71	93.37	97.40	97.80	95.64	64.58	**98.28**	92.47
DT	33 (0.04)	99.16	98.64	94.69	**99.38**	96.81	**99.38**	98.86	63.63	96.78	86.56	93.30	98.06	**98.02**	95.49	64.03	96.85	92.48
DT	29 (0.03)	99.01	98.46	94.69	99.34	96.12	99.12	99.12	64.03	96.74	85.02	92.56	**98.17**	97.99	95.75	64.14	97.66	92.37
DT	28 (0.02/1)	99.01	98.53	94.32	99.23	96.34	99.27	99.12	62.89	96.89	85.20	92.60	98.06	97.84	94.84	63.92	97.77	92.24

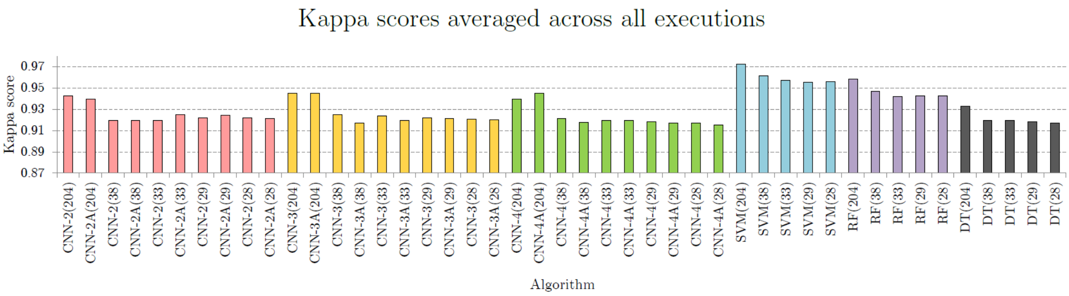

FIGURE 10. Average kappa scores obtained for the full and reduced Salinas Valley set (we report the number of bands in parentheses).

On the other hand, we can observe only minor improvements in the performance when more BBs are appended. It shows that the shallower models can extract high-quality features using just two convolutional-pooling blocks. The same observation can be drawn from Figures 10–11, where we render the kappa scores for Salinas and Pavia. There are classes (C8 and C15 for Salinas, and C1, C2 and C8 for Pavia) which are "difficult" for all classifiers (Tables 5–6). In both cases, it is observed for the most numerous classes, and it can be attributed to the fact that they are under-sampled while creating the balanced training sets. Therefore, the sampled examples are not representative.

TABLE 6. Classification accuracy (in %) of various models obtained for the full and reduced Pavia University dataset (we report the number of bands and the contamination rate in parentheses; "Full" for no reduction). The best results in each column and for each classifier (collectively for CNNs with and without attention) are boldfaced, whereas the worst results are grayed.

Algorithm	Bands	C1	C2	C3	C4	C5	C6	C7	C8	C9	AA
CNN-2	103 (Full)	85.82	**91.31**	**86.28**	95.89	99.65	93.12	**95.35**	**85.11**	99.75	**92.47**
CNN-2A	103 (Full)	**86.77**	90.21	84.68	96.28	99.65	**93.76**	95.32	84.65	99.86	92.35
CNN-2	28 (0.05)	77.80	87.09	79.79	96.10	99.61	86.95	91.84	81.60	99.96	88.97
CNN-2A	28 (0.05)	80.82	85.67	80.46	**96.63**	**99.86**	88.51	91.88	82.84	**100.0**	89.63
CNN-2	20 (0.04)	76.60	77.98	77.77	90.92	99.57	85.74	89.04	76.42	99.96	86.00
CNN-2A	20 (0.04)	72.98	77.66	76.31	91.42	99.47	83.37	89.89	76.81	**100.0**	85.32
CNN-2	14 (0.03)	69.43	60.00	55.21	91.88	99.68	82.20	90.21	76.81	**100.0**	80.60
CNN-2A	14 (0.03)	67.41	64.04	63.16	92.34	99.50	81.77	88.69	70.35	99.93	80.80
CNN-2	12 (0.02)	66.31	50.11	51.10	91.35	98.69	76.31	86.63	73.87	99.96	77.15
CNN-2A	12 (0.02)	66.21	49.08	51.67	90.60	98.79	75.25	87.70	71.95	99.86	76.79
CNN-2	9 (0.01)	62.91	44.82	40.04	89.54	98.72	76.13	85.35	71.45	99.89	74.32
CNN-2A	9 (0.01)	63.69	46.63	48.87	88.87	98.69	74.11	82.73	68.69	99.89	74.68
CNN-3	103 (Full)	**87.62**	**90.74**	85.85	**97.87**	99.65	**94.40**	95.71	**85.32**	**100.0**	**93.02**
CNN-3A	103 (Full)	86.17	89.08	**86.10**	97.45	**99.82**	93.83	**95.92**	83.62	99.86	92.43
CNN-3	28 (0.05)	81.52	80.53	83.83	95.85	99.75	89.18	91.77	79.01	**100.0**	89.05
CNN-3A	28 (0.05)	81.81	85.89	79.50	95.50	99.54	86.60	92.09	80.60	99.93	89.05
CNN-3	20 (0.04)	74.29	74.65	75.25	91.24	99.54	83.65	89.43	74.89	99.89	84.76
CNN-3A	20 (0.04)	72.77	79.82	73.62	90.53	99.61	78.79	89.40	76.38	**100.0**	84.55
CNN-3	14 (0.03)	71.38	62.20	64.82	91.45	99.57	78.97	87.87	70.89	99.89	80.78
CNN-3A	14 (0.03)	68.94	62.52	61.84	92.13	99.40	80.53	87.84	70.71	99.89	80.42
CNN-3	12 (0.02)	68.65	50.53	57.06	91.99	98.97	73.90	85.50	69.11	99.86	77.29
CNN-3A	12 (0.02)	68.01	49.75	60.00	90.82	99.04	76.77	86.60	65.96	99.79	77.42
CNN-3	9 (0.01)	65.71	43.69	46.13	88.94	98.48	78.12	82.20	68.26	99.86	74.60
CNN-3A	9 (0.01)	64.22	46.17	43.44	89.47	98.58	76.06	83.90	72.13	99.89	74.87
CNN-4	103 (Full)	85.71	89.01	**84.96**	96.45	99.65	**94.47**	93.90	84.29	99.93	92.04
CNN-4A	103 (Full)	**87.02**	**89.89**	84.65	**97.30**	99.68	93.83	**94.43**	**85.67**	**99.96**	**92.49**
CNN-4	28 (0.05)	81.13	86.45	83.97	95.64	99.47	87.52	91.84	76.63	99.89	89.17
CNN-4A	28 (0.05)	78.62	84.57	82.48	94.33	99.47	85.53	92.02	78.05	99.93	88.33
CNN-4	20 (0.04)	74.33	76.60	70.28	88.97	**99.75**	80.21	89.36	76.56	99.86	83.99
CNN-4A	20 (0.04)	75.82	79.26	76.56	91.10	99.54	81.31	88.97	77.38	99.89	85.54
SVM	103 (Full)	**88.90**	**93.97**	**87.94**	**97.91**	99.86	**94.65**	**94.86**	**88.90**	**99.93**	**94.10**
SVM	28 (0.05)	81.91	87.84	84.29	96.99	99.86	89.36	93.09	82.34	99.89	90.62
SVM	20 (0.04)	76.74	85.89	81.84	93.65	**99.89**	85.92	92.77	79.68	99.89	88.48
SVM	14 (0.03)	71.35	71.95	72.59	92.20	99.86	82.98	90.82	75.67	**99.93**	84.15
SVM	12 (0.02)	67.48	53.97	63.90	93.05	98.87	74.72	88.44	73.05	99.79	79.25
SVM	9 (0.01)	64.65	49.86	50.57	92.34	98.90	76.95	85.71	71.81	99.79	76.73
RF	103 (Full)	**82.55**	**83.40**	**85.71**	**96.28**	**99.50**	**88.09**	**91.49**	**83.83**	**100.0**	**90.09**
RF	28 (0.05)	80.25	80.64	79.86	95.35	99.47	86.99	90.60	82.27	99.96	88.38
RF	20 (0.04)	77.20	73.05	74.26	94.04	99.33	83.83	89.47	80.04	99.96	85.69
RF	14 (0.03)	74.54	62.84	64.11	92.77	99.72	79.15	88.26	73.19	**100.0**	81.62
RF	12 (0.02)	69.57	53.83	60.14	92.87	99.08	73.83	86.77	70.96	**100.0**	78.56
RF	9 (0.01)	65.25	49.36	53.48	91.31	98.76	73.01	83.90	68.33	**100.0**	75.93
DT	103 (Full)	**80.67**	**77.77**	**74.57**	**92.94**	**99.22**	80.57	**86.70**	**74.96**	99.96	**85.26**
DT	28 (0.05)	78.90	75.43	71.88	91.28	98.79	**80.71**	84.33	71.17	**100.0**	83.61
DT	20 (0.04)	75.96	69.26	67.02	86.03	**99.22**	76.35	83.76	69.18	99.96	80.75
DT	14 (0.03)	72.02	62.80	57.38	82.73	98.79	72.30	81.84	64.01	99.96	76.87
DT	12 (0.02)	68.97	54.96	54.72	82.34	98.58	65.14	78.76	61.03	99.96	73.83
DT	9 (0.01)	66.17	53.90	49.33	81.84	98.83	61.31	72.59	60.67	**100.0**	71.63

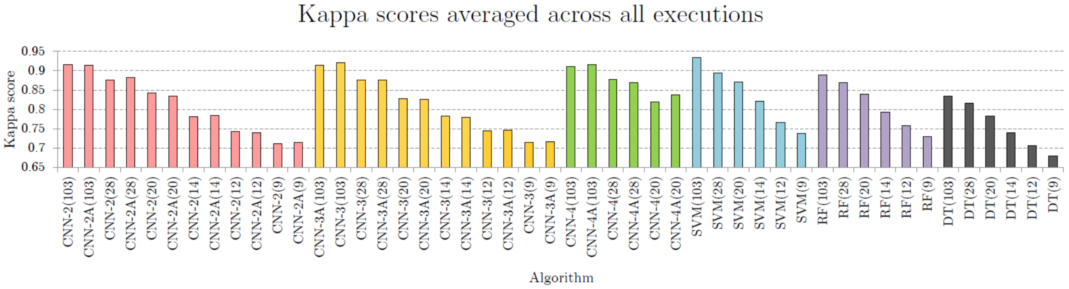

FIGURE 11. Average kappa scores obtained for the full and reduced Pavia University set (we report the number of bands in parentheses).

The average number of training epochs before reaching convergence, and the average processing time[6] of a single epoch are presented in Figs. 12–13, respectively. Appending attention modules increases neither the number of epochs nor the processing time (see CNN-2 vs. CNN-2A, CNN-3 vs. CNN-3A, and CNN-4 vs. CNN-4A in both figures; standard deviations remain the same too), hence they can be considered as a seamless CNN extension to enhance its

[6] Using NVIDIA Titan X Ultimate Pascal GPU 12 GB GDDR5X.

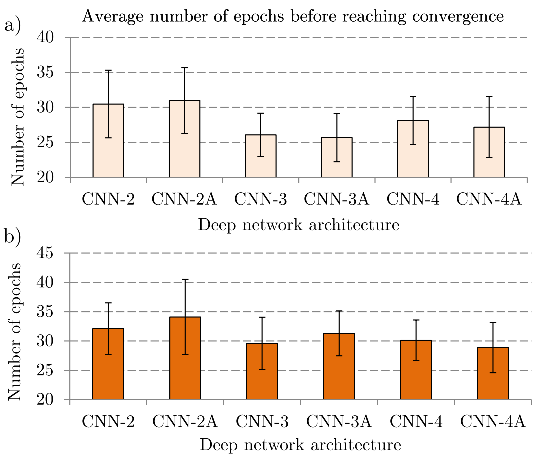

FIGURE 12. Average number of epochs before reaching convergence for the a) Salinas Valley and b) Pavia University datasets.

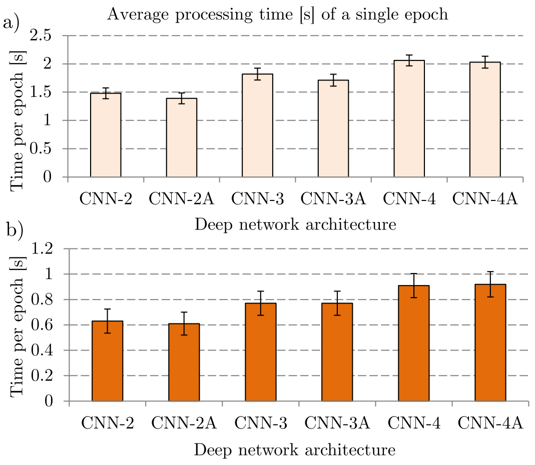

FIGURE 13. Average processing time [s] of a single epoch for the a) Salinas Valley and b) Pavia University datasets.

operational ability (it not only does learn how to effectively classify HSI pixels but also selects important HSI bands). Similarly, more BBs minimally affected the overall execution time of our band selection algorithm (reported in Table 8).

E. CLASSIFICATION ACCURACY OVER REDUCED DATASETS

In this experiment, we evaluated the classification performance of well-established state-of-the-art models trained using full and reduced HSI datasets. These classifiers included Support Vector Machines (SVMs), Random Forests (RFs), and Decision Trees (DTs). We followed the same experimental scenario, however we also executed grid search to optimize the hyper-parameters of all models: C and γ of the radial-basis kernel function in SVMs, number of trees in RFs, minimum samples per leaf in DTs, and minimum samples in a split in both RFs and DTs. The training with grid search was repeated 30 times (Monte-Carlo cross-validation). We report the grid-search characteristics in Table 7. The results show that decreasing the HSI datasets (the lower λ values are, the higher reduction rates are obtained, as given in Table 4) helps shorten the grid-search time which can easily become very large for full datasets (e.g., SVM for Salinas). Such hyper-parameter optimizations are not necessary in our CNNs.

TABLE 7. Average grid-search time [s] for the a) Salinas Valley and b) Pavia University datasets for all contamination rates λ.

	Algorithm	$\lambda \rightarrow 0.01$	0.02	0.03	0.04	0.05	Full
a)	SVM	316.2	330.0	370.8	394.8	469.2	3,466.8
	RF	42.6	42.6	42.6	43.8	46.8	78.6
	DT	10.8	10.8	10.8	12.6	13.8	65.4
b)	SVM	84.0	93.0	99.6	120.0	143.4	431.4
	RF	27.6	27.0	27.0	31.2	34.8	54.6
	DT	3.6	4.2	4.8	6.0	7.8	28.8

The average-accuracy results gathered in Tables 5–6 show that for most of the classes, the performance of the investigated classifiers is not diminished by our band selection technique. Although there exist classes for which the accuracy decreased (e.g., C2 and C3 in Pavia), the differences for other classes are rather negligible, especially for CNNs for $\lambda \geq 0.03$ (note that CNN-4A could not be trained for very small number of bands because of the dimensionality reduction performed in the pooling layers). This observation is proved by the Wilcoxon tests (across both Salinas and Pavia sets) executed to analyze the differences between models trained with different datasets (with and without reduction). Although the differences in AA of the classifiers trained with the reduced numbers of bands are statistically important (at $p < 0.01$), they are not as dramatic as in other state-of-the-art band selection algorithms [60].

The inference time of all investigated learners was very short, and allowed for real-time processing. Reducing the number of bands decreased the *total* inference time (of all examples in the test sets Ψ which amounted to approx. 1,500 examples in Salinas Valley, and to approx. 850 examples in Pavia University, averaged across all runs) for both datasets: 0.06 down to 0.03 s (CNN-2A), 0.07 to 0.04 s (CNN-3A), 0.09 to 0.04 s (CNN-4A), and 0.16 to 0.12 (SVM) for Salinas Valley ($\lambda = 0.01$; the inference time for RF and DT remained unchanged and it amounted to 0.1 s and less than 0.01 s, respectively). The decrease in the inference time was analogous for the Pavia University dataset. Also, we do not report the times for CNNs without attention as they were practically the same as for the attention-based CNNs. This inference time is independent from the utilized band selection algorithm—it depends on the number of selected bands.

F. COMPARISON WITH THE STATE OF THE ART

We compare our band selection algorithm with other state-of-the-art techniques. For the sake of thoroughness, we took into consideration both filter and wrapper approaches. As a filter (unsupervised) algorithm, we implemented the method by Guo *et al.* [70] which exploits the mutual information across the bands in the band selection process (we refer to this algorithm as MI). In [70], the authors used the estimated reference maps (rather than the ground-truth segmentation) to calculate

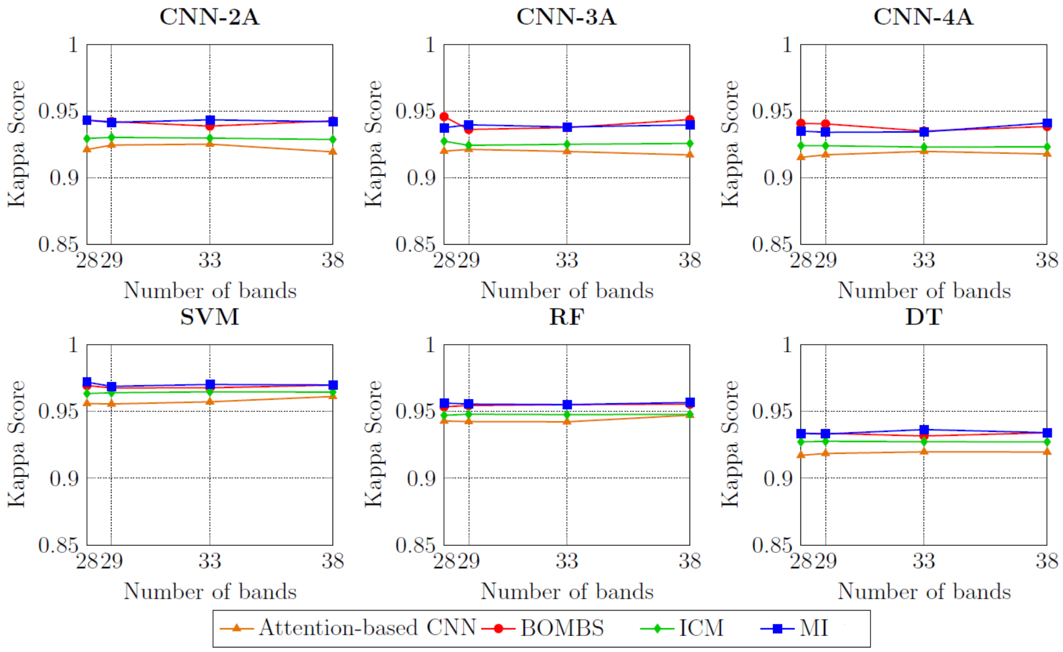

FIGURE 14. Kappa coefficient scores elaborated over the unseen test sets Ψ for Salinas Valley.

the mutual information. Since this map should be estimated using available knowledge about the spectral signatures of the materials encountered within the scene, the lack of such signatures may easily lead to inaccurate (or even incorrect) reference maps. For fair comparison, we used the original ground-truth information instead of such estimated reference maps—it, in turn, can render over-optimistic results for this method (i.e., our MI implementation is "handicapped" by the availability of the ground truth).

As the wrapper approaches, we selected two modern algorithms by Zhang *et al.* [83] and by Cao *et al.* [82]. Zhang *et al.* [83] introduced a multi-objective immune algorithm (BOMBS) for band selection from HSI. Two objectives incorporated in this multi-objective optimization include the amount of conveyed information and the redundancy within the selected subset of bands. In [82], the authors assess the quality of selected subsets of HSI bands using the pixel-wised classification map (obtained using support vector machines with the radial-basis kernel function) enhanced by the edge preserved filtering (they use the information about the unlabeled and labeled pixels simultaneously)—we refer to this method as ICM (**I**mproved **C**lassification **M**ap). The hyper-parameters of all state-of-the-art methods were set as reported in the corresponding papers, however in all methods we need to specify the desired number of bands in the reduced HSI. For fair comparison, we extracted the same number of bands as presented in Table 4. All state-of-the-art methods were implemented in `Python 3.6`, and they are available in the same Git repository (see Section IV-A).

TABLE 8. Average time [s] of the investigated algorithms for the a) Salinas Valley and b) Pavia University datasets for all contamination rates λ.

	Algorithm	$\lambda \to 0.01$	0.02	0.03	0.04	0.05
a)	**CNN-2A**	—	62.0	60.0	62.6	55.6
	CNN-3A	—	64.3	66.4	64.6	65.1
	CNN-4A	—	67.2	73.0	67.4	69.9
	BOMBS [83]	—	9,364.3	9,516.6	9,868.3	10,455.1
	ICM [82]	—	5,607.1	5,901.0	6,803.8	7,746.8
	MI [70]	—	3.4	3.4	3.4	3.4
b)	**CNN-2A**	29.4	30.2	29.1	29.4	28.4
	CNN-3A	33.4	40.0	34.3	33.9	34.0
	CNN-4A	41.0	43.0	42.0	42.3	39.9
	BOMBS [83]	8,329.9	8,717.4	8,830.3	9,286.0	9,975.8
	ICM [82]	180.1	204.0	222.0	277.6	358.4
	MI [70]	3.1	3.1	3.1	3.1	3.1

The kappa scores are presented in Figures 14 and 15 for Salinas Valley and Pavia University, respectively. They consistently grow for all techniques with the increase of the number of extracted bands, and ultimately converge to the same values. As already mentioned, the results for MI may be over-optimistic, as we utilize the entire ground-truth information to extract the important bands. Hence, we "leak" the information across the training and test sets because the training-validation-test splits are created *after* the band selection step, and before training a supervised learner. The execution times (Table 8) show that our technique is orders of magnitude faster when compared with the wrapper algorithms (BOMBS and ICM) while delivering competitive classification results. Also, the number of bands selected in all methods was set according to our contamination factors—if we did not know the desired number of bands, we would have to execute each method in a grid search-like manner, and it would drastically increase their running time.

In Figures 16–17, we render RGB color-composite visualizations of three bands randomly sampled from the subsets of bands obtained using all investigated band selection

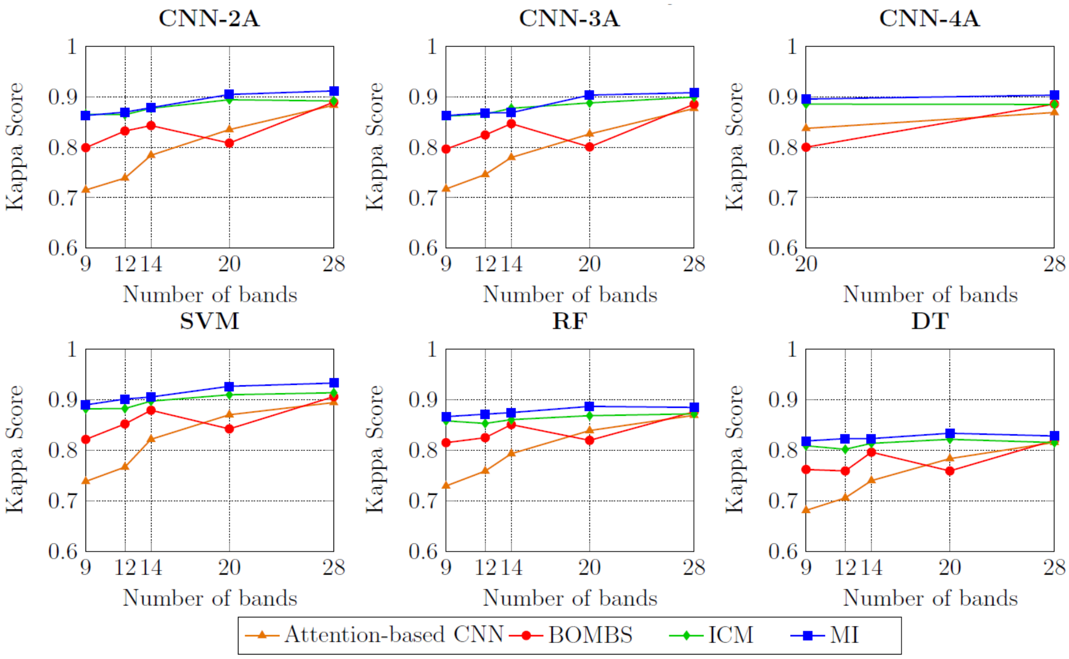

FIGURE 15. Kappa coefficient scores elaborated over the unseen test sets Ψ for Pavia University.

TABLE 9. A rough comparison (the kappa values) of our band selection with other state-of-the-art techniques over a) Salinas Valley and b) Pavia University. For Salinas Valley, we selected 28 bands using attention-based CNNs followed by anomaly detection (the contamination rate was λ = 0.01, as presented in Table 4), whereas the results of other techniques are reported for 24 bands. On the other hand, we selected 20 bands for Pavia University (λ = 0.04) using our algorithm, to keep the number of bands (and contamination rates) consistent across the experiments for this set, whereas the other techniques extracted 21 bands [9].

	Bands selected using other methods									Bands selected using attention-based CNNs								
	MVPCA	CFSFDP	SC	SSR	ISSC	LRSC	L2/3-LRSC	TV-LRSC	L2-LRSC	CNN-2	CNN-2A	CNN-3	CNN-3A	CNN-4	CNN-4A	SVM	RF	DT
a)	0.8356	0.9200	0.9201	0.8851	0.9230	0.9228	0.9255	0.9258	0.9285	0.9222	0.9212	0.9207	0.9200	0.9170	0.9154	0.956	0.9429	0.9172
b)	0.5951	0.8485	0.8927	0.8828	0.8944	0.9016	0.9120	0.9171	0.9235	0.8425	0.8349	0.8285	0.8262	0.8199	0.8373	0.8703	0.8390	0.7834

algorithms (the reflectance values captured within the bands mentioned in the parentheses are mapped onto the RGB space, hence the values for each band are normalized to the [0, 255] range [94]). These examples show that all band selection algorithms can effectively extract informative bands which are indeed useful in distinguishing different materials in the scene (see e.g., different shades of green for the attention-based CNN in Figure 16, or different shades and colors for all methods in Figure 17). However, we can also appreciate examples in which three randomly selected bands would not be enough to accurately segment the image, e.g., the sixth column, (83, 89, 103) for BOMBS in Figure 16 (they are likely useful when combined with other bands in the corresponding reduced HSI).

Finally, to further confront our attention-based CNNs with the state of the art, we performed a rough comparison of the results obtained using maximum-variance PCA (MVPCA) [11], clustering by fast search and find of density peaks (CFSFDP) [62], [95], spectral clustering (SC) [96], sparse self-representation (SSR) [97], improved sparse subspace clustering (ISSC) [62], low-rank subspace clustering (LRSC) [9], and its several variants: $\ell_{2/3}$-norm regularized LRSC (L2/3-LRSC), total variation-regularized LRSC (TV-LRSC), and ℓ_2-norm regularized LRSC (L2-LRSC). The aforementioned techniques have been applied to select 24 and 21 bands for the Salinas Valley and Pavia University datasets, respectively [9], whereas we selected a similar (but not exactly the same) number of bands using attention-based CNNs—28 for Salinas Valley, and 20 bands for Pavia University, in order to keep the contamination rates consistent across all experiments (see Table 4). The kappa values (Table 9) indicate that the attention-based CNNs followed by anomaly detection can consistently deliver high-quality subsets of all bands, and they are competitive with the state of the art. All of the methods ran in well below 2 minutes for both datasets [9].

G. DISCUSSION

The experimental results reported in Sections IV-C–IV-F shed more light on the abilities of our attention-based CNNs followed by anomaly detection exploited for hyperspectral band selection. Our technique was not only compared with other methods from the literature, but we also investigated the impact of applying the attention modules in various convolutional architectures. The experiments showed that the attention-based CNNs are able to effectively select

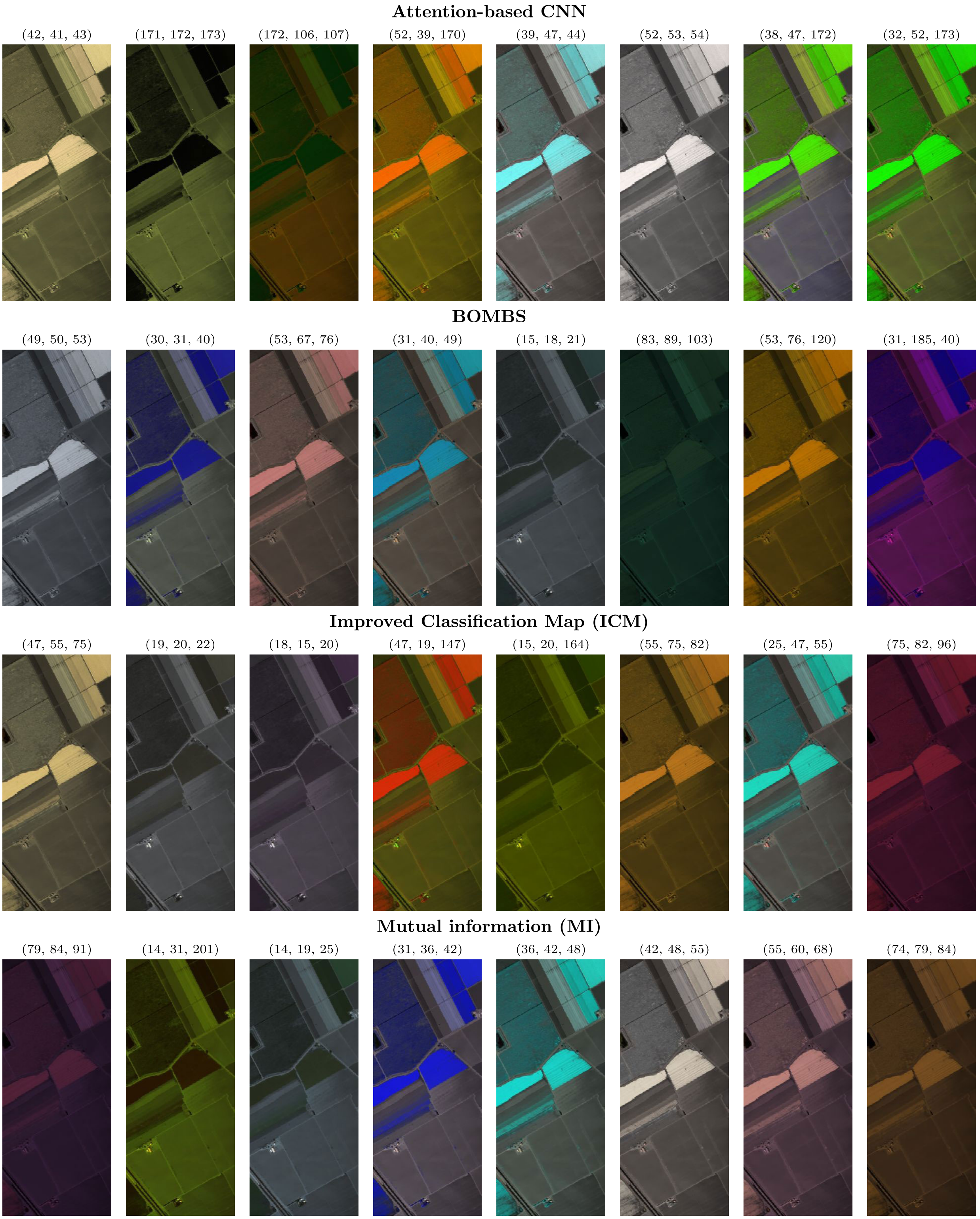

FIGURE 16. The RGB visualization of three randomly selected bands for Salinas Valley.

important bands from an input HSI without deteriorating the classification performance of supervised learners (both deep learning-powered and conventional classifiers), the attention modules adversely affect neither the training/inference times nor the classification accuracy of the underlying models (according to the non-parametric Wilcoxon tests), and they are competitive with the state-of-the-art band selection approaches.

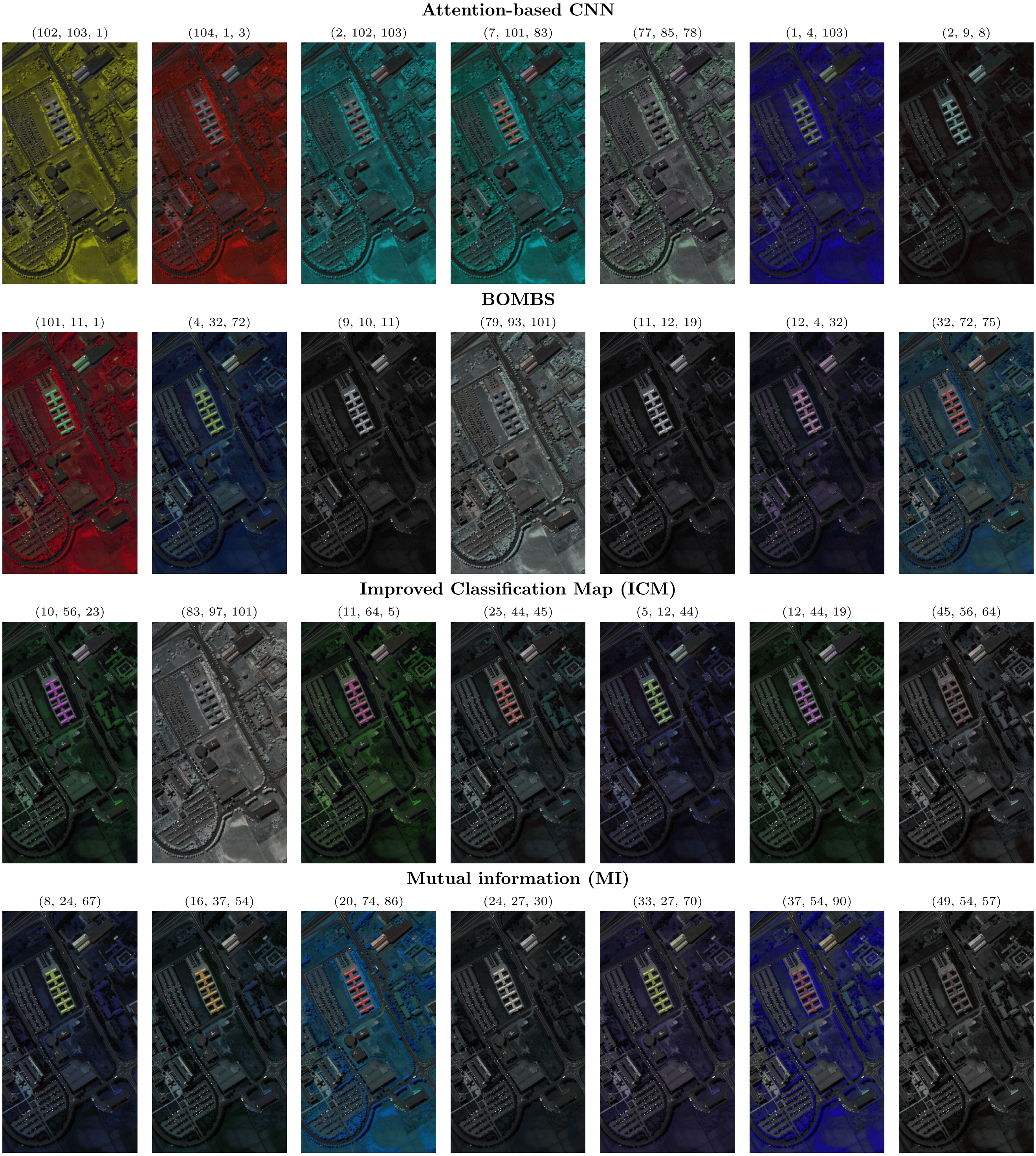

FIGURE 17. The RGB visualization of three randomly selected bands for Pavia University.

Importantly, the training process is designed in such a way that it outputs the weighted bands (according to the attention scores), alongside a ready-to-use deep model that can be exploited to classify the incoming hyperspectral samples. The computational investigation showed that our algorithm can run orders of magnitude faster than the wrapper techniques while delivering very competitive results. Additionally, the reduced datasets can significantly speed up the grid-search process that is required while deploying practically all popular supervised learners (note that the only hyperparameter of our method is the contamination rate), together with the inference time of the trained models. On the other

TABLE 10. The advantages and disadvantages of the proposed attention-based CNNs for hyperspectral band selection.

Advantages	Disadvantages
• Selects a subset of bands and trains a fully-functional deep model for hyperspectral classification in one pass • Attention modules deteriorate neither the classification performance nor the training/inference time of the CNNs • Attention modules are seamlessly applicable to any convolutional neural network architecture, and the capacity of the deep learner can reflect the difficulty of the classification problem • Can select any number of bands • Data-driven, straightforwardfly applicable to new multi/hyperspectral data • Easy to implement and understand • Works orders of magnitude faster than the wrapper approaches • Any anomaly detection can be applied to select important bands after the attention-based weighting	• Number of bands for selection (the contamination rate λ) is a hyperparameter that needs to be fine-tuned • Delivers worse results for very small contamination rates compared with the state-of-the-art techniques

hand, the attention-based CNNs elaborated slightly worse reduced band subsets for the smallest contamination rates. It may indicate that the selected bands were neighboring in the spectral dimension (due to the peaks in the attention scores), and more diversely selected bands could have been more appropriate to represent the characteristics of objects captured in the analyzed scenes. Also, as the deep networks are high-capacity learners, they may be vulnerable to the lack of ground-truth samples. These issues, however, require further investigation and constitute our current research focus.

Our experiments provided us with the quantitative, qualitative, and statistical evidence on the capabilities of the proposed band-selection framework, and allowed us to gain insights into the most important advantages and disadvantages of our technique—they have been summarized in Table 10.

V. CONCLUSION

In this paper, we proposed new attention-based CNNs coupled with anomaly detection for selecting bands from HSI. The attention modules can be seamlessly incorporated into any CNN architecture and they affect neither classification abilities nor training times of CNNs. Such CNNs are fully-functional after their training, and can be used for classifying new data. Experimental validation executed over two benchmark HSI datasets (Salinas Valley and Pavia University), backed up with statistical tests, showed that the proposed algorithm extracts important bands from HSI, and allows us to obtain state-of-the-art classification accuracy using only a fraction of all bands (14–19% for Salinas, and 9–27% for Pavia). Various visualizations helped understand which parts of the spectrum are important in each dataset (our band selection can also enhance interpretability of HSI), and showed that our approach is data-driven and can be easily applied to any HSI dataset. Overall, our rigorous experiments revealed that:

- Attention-based CNNs deliver high-quality classification, and adding attention modules does not impact classification abilities and training time of an underlying CNN.
- Attention-based CNNs followed by the anomaly detection extract the most informative bands in a HSI dataset during the training process (hence, it is an embedded algorithm).
- Bands selected by our algorithm can be used to identify relevant and discard unimportant parts of the spectrum, drastically shortening training times of a classifier, and compressing the HSI data without sacrificing the amount of conveyed information. This compression is especially useful in hardware- and cost-constrained real-life scenarios.
- Our technique is data-driven and can be easily applied to any HSI dataset and any CNN architecture.
- Our technique is competitive with the state-of-the-art filter and wrapper band selection approaches, and works orders of magnitude faster than the latter algorithms.

Our approach can be used to reduce HSI datasets on board of a satellite before transferring HSI to Earth without sacrificing the amount of important information being transferred. It would ultimately decrease the transfer time (alongside its enormous cost), and make the hyperspectral imaging even more affordable in real-life Earth Observation scenarios.

Generating high-quality ground-truth hyperspectral scenes that may be used for training supervised learners is a time-consuming, user-dependent and cumbersome task. Hence, the size of annotated hyperspectral datasets is very limited in practice, and data augmentation [98] and transfer learning [99] techniques are being researched to deal with this issue in the context of their deep learning-powered supervised classification and segmentation. Since our attention-based CNNs require ground-truth hyperspectral data to extract useful bands from the original imagery, our current research focus is put on understanding the impact of the training set size on their performance. Also, we aim at verifying the influence of synthetic training samples on the abilities of our CNNs, especially when generated using noise-injection techniques and generative adversarial nets [100]. Finally, we work on the quantized versions of our attention-based CNNs which will be deployed on board of an imaging satellite, in a very hardware-constrained execution environment [101].

ACKNOWLEDGMENT

The authors are grateful to the anonymous reviewers for their constructive and valuable comments that helped improve the article.

REFERENCES

[1] L. Jiao, W. Sun, G. Yang, G. Ren, and Y. Liu, "A hierarchical classification framework of satellite Multispectral/Hyperspectral images for mapping coastal wetlands," *Remote Sens.*, vol. 11, no. 19, p. 2238, Sep. 2019, doi: 10.3390/rs11192238.

[2] T. Dundar and T. Ince, "Sparse representation-based hyperspectral image classification using multiscale superpixels and guided filter," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 2, pp. 246–250, Feb. 2019.

[3] B. Du and L. Zhang, "Random-selection-based anomaly detector for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 5, pp. 1578–1589, May 2011.

[4] Q. Zhang, Q. Yuan, J. Li, X. Liu, H. Shen, and L. Zhang, "Hybrid noise removal in hyperspectral imagery with a spatial–spectral gradient network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7317–7329, Oct. 2019.

[5] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 4, no. 2, pp. 22–40, Jun. 2016.

[6] J. Nalepa, M. Myller, and M. Kawulok, "Validating hyperspectral image segmentation," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 8, pp. 1264–1268, Aug. 2019.

[7] F. Luo, B. Du, L. Zhang, L. Zhang, and D. Tao, "Feature learning using spatial-spectral hypergraph discriminant analysis for hyperspectral image," *IEEE Trans. Cybern.*, vol. 49, no. 7, pp. 2406–2419, Jul. 2019.

[8] F. Luo, L. Zhang, B. Du, and L. Zhang, "Dimensionality reduction with enhanced hybrid-graph discriminant learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, pp. 1–18, 2020.

[9] H. Zhai, H. Zhang, L. Zhang, and P. Li, "Laplacian-regularized low-rank subspace clustering for hyperspectral image band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1723–1740, Mar. 2019, doi: 10.1109/TGRS.2018.2868796.

[10] R. Yang, L. Su, X. Zhao, H. Wan, and J. Sun, "Representative band selection for hyperspectral image classification," *J. Vis. Commun. Image Represent.*, vol. 48, pp. 396–403, 2017.

[11] C.-I. Chang, Q. Du, T.-L. Sun, and M. L. G. Althouse, "A joint band prioritization and band-decorrelation approach to band selection for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 6, pp. 2631–2641, Nov. 1999.

[12] M. Gong, M. Zhang, and Y. Yuan, "Unsupervised band selection based on evolutionary multiobjective optimization for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 544–557, Jan. 2016.

[13] A. MartÌnez-UsÓMartinez-Uso, F. Pla, J. M. Sotoca, and P. GarcÌa-Sevilla, "Clustering-based hyperspectral band selection using information measures," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4158–4171, Dec. 2007.

[14] X. Cao, T. Xiong, and L. Jiao, "Supervised band selection using local spatial information for hyperspectral image," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 3, pp. 329–333, Mar. 2016.

[15] L. Ma, M. Li, Y. Gao, T. Chen, X. Ma, and L. Qu, "A novel wrapper approach for feature selection in object-based image classification using polygon-based cross-validation," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 3, pp. 409–413, Mar. 2017.

[16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, no. 521, pp. 436–555, 2016.

[17] F. Li, D. A. Clausi, L. Xu, and A. Wong, "ST-IRGS: A region-based self-training algorithm applied to hyperspectral image classification and segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 3–16, Jan. 2018.

[18] Q. Gao, S. Lim, and X. Jia, "Hyperspectral image classification using convolutional neural networks and multiple feature learning," *Remote Sens.*, vol. 10, no. 2, p. 299, Feb. 2018.

[19] P. Zhong, Z. Gong, S. Li, and C.-B. Schonlieb, "Learning to diversify deep belief networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 6, pp. 3516–3530, Jun. 2017.

[20] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, New Orleans, LA, USA, vol. 2, Jun. 2018, pp. 464–468, doi: 10.18653/v1/n18-2074.

[21] N. Parmar, P. Ramachandran, A. Vaswani, I. Bello, A. Levskaya, and J. Shlens, "Stand-alone self-attention in vision models," in *Proc. Annu. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Vancouver, BC, Canada, Dec. 2019, pp. 68–80. [Online]. Available: http://papers.nips.cc/paper/8302-stand-alone-self-attention-in-vision-models

[22] P. Rodriguez, G. Cucurull, J. Gonzalez, J. M. Gonfaus, and X. Roca, "A painless attention mechanism for convolutional neural networks," in *Proc. OpenReview*, 2018, pp. 1–14.

[23] P. Ribalta Lorenzo, L. Tulczyjew, M. Marcinkiewicz, and J. Nalepa, "Band selection from hyperspectral images using attention-based convolutional neural networks," 2018, *arXiv:1811.02667*. [Online]. Available: http://arxiv.org/abs/1811.02667

[24] W. Sun and Q. Du, "Hyperspectral band selection: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 2, pp. 118–139, Jun. 2019.

[25] J. Wang, J. Zhou, W. Huang, and J. F. Chen, "Attention networks for band weighting and selection in hyperspectral remote sensing image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2019, pp. 3820–3823.

[26] X. Mei, E. Pan, Y. Ma, X. Dai, J. Huang, F. Fan, Q. Du, H. Zheng, and J. Ma, "Spectral-spatial attention networks for hyperspectral image classification," *Remote Sens.*, vol. 11, no. 8, p. 963, Apr. 2019, doi: 10.3390/rs11080963.

[27] K.-H. Liu, S.-Y. Chen, H.-C. Chien, and M.-H. Lu, "Progressive sample processing of band selection for hyperspectral image transmission," *Remote Sens.*, vol. 10, no. 3, p. 367, Feb. 2018.

[28] H. Xu, H. Zhang, W. He, and L. Zhang, "Superpixel-based spatial-spectral dimension reduction for hyperspectral imagery classification," *Neurocomputing*, vol. 360, pp. 138–150, Sep. 2019, doi: 10.1016/j.neucom.2019.06.023.

[29] J. Zabalza, J. Ren, Z. Wang, S. Marshall, and J. Wang, "Singular spectrum analysis for effective feature extraction in hyperspectral imaging," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 11, pp. 1886–1890, Nov. 2014.

[30] Z. Chen, J. Jiang, X. Jiang, X. Fang, and Z. Cai, "Spectral-spatial feature extraction of hyperspectral images based on propagation filter," *Sensors*, vol. 18, no. 6, p. 1978, Jun. 2018.

[31] A. Agarwal, T. El-Ghazawi, H. El-Askary, and J. Le-Moigne, "Efficient hierarchical-PCA dimension reduction for hyperspectral imagery," in *Proc. IEEE SSPIT*, Dec. 2007, pp. 353–356.

[32] L. Wang, X. Xie, W. Li, Q. Du, and G. Li, "Sparse feature extraction for hyperspectral image classification," in *Proc. IEEE ChinaSIP*, Jul. 2015, pp. 1067–1070.

[33] J. Jiang, J. Ma, C. Chen, Z. Wang, Z. Cai, and L. Wang, "SuperPCA: A superpixelwise PCA approach for unsupervised feature extraction of hyperspectral imagery," *CoRR*, vol. abs/1806.09807, Aug. 2018. [Online]. Available: http://arxiv.org/abs/1806.09807

[34] M. Ye, C. Ji, H. Chen, L. Lei, H. Lu, and Y. Qian, "Residual deep PCA-based feature extraction for hyperspectral image classification," *Neural Comput. Appl.*, Oct. 2019, doi: 10.1007/s00521-019-04503-3.

[35] M. P. Uddin, M. A. Mamun, and M. A. Hossain, "Feature extraction for hyperspectral image classification," in *Proc. IEEE RHTC*, Dec. 2017, pp. 379–382.

[36] W. Li, S. Prasad, J. E. Fowler, and L. M. Bruce, "Locality-preserving dimensionality reduction and classification for hyperspectral image analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 4, pp. 1185–1198, Apr. 2012.

[37] F. Luo, H. Huang, Y. Duan, J. Liu, and Y. Liao, "Local geometric structure feature for dimensionality reduction of hyperspectral imagery," *Remote Sens.*, vol. 9, no. 8, p. 790, Aug. 2017, doi: 10.3390/rs9080790.

[38] F. Luo, L. Zhang, X. Zhou, T. Guo, Y. Cheng, and T. Yin, "Sparse-adaptive hypergraph discriminant analysis for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, to be published.

[39] J. Wang and C.-I. Chang, "Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 6, pp. 1586–1600, Jun. 2006.

[40] Z. Yawen, D. Guangjun, and X. Zhixiang, "Hyperspectral image tensor feature extraction based on fusion of multiple spectral-spatial features," in *Proc. ACM ICIIP*. New York, NY, USA: ACM, 2016, pp. 43:1–43:8.

[41] W. Sun, A. Halevy, J. J. Benedetto, W. Czaja, W. Li, C. Liu, B. Shi, and R. Wang, "Nonlinear dimensionality reduction via the ENH-LTSA method for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 2, pp. 375–388, Feb. 2014.

[42] W. Sun, A. Halevy, J. J. Benedetto, W. Czaja, C. Liu, H. Wu, B. Shi, and W. Li, "UL-isomap based nonlinear dimensionality reduction for hyperspectral imagery classification," *ISPRS J. Photogram. Remote Sens.*, vol. 89, pp. 25–36, Mar. 2014.

[43] L. Zhang, L. Zhang, D. Tao, X. Huang, and G. Xia, "Nonnegative discriminative manifold learning for hyperspectral data dimension reduction," in *Proc. 5th Workshop Hyperspectral Image Signal Process., Evol. Remote Sens. (WHISPERS)*, Jun. 2013, pp. 1–4.

[44] W. Sun, L. Zhang, and B. Du, "Feature extraction using near-isometric linear embeddings for hyperspectral imagery classification," in *Proc. 8th Workshop Hyperspectral Image Signal Process., Evol. Remote Sens. (WHISPERS)*, Aug. 2016, pp. 1–4.

[45] W. Sun, G. Yang, B. Du, L. Zhang, and L. Zhang, "A sparse and low-rank near-isometric linear embedding method for feature extraction in hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 4032–4046, Jul. 2017, doi: 10.1109/TGRS.2017.2686842.

[46] P.-H. Hsu, "Feature extraction of hyperspectral images using wavelet and matching pursuit," *ISPRS J. Photogram. Remote Sens.*, vol. 62, no. 2, pp. 78–92, Jun. 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0924271607000020

[47] X. Guo, X. Huang, and L. Zhang, "Three-dimensional wavelet texture feature extraction and classification for multi/hyperspectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 12, pp. 2183–2187, Dec. 2014, doi: 10.1109/LGRS.2014.2323963.

[48] L. Zhang, L. Zhang, D. Tao, and X. Huang, "Tensor discriminative locality alignment for hyperspectral image spectral–spatial feature extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 1, pp. 242–256, Jan. 2013, doi: 10.1109/TGRS.2012.2197860.

[49] X. Huang, L. Zhang, and P. Li, "An adaptive multiscale information fusion approach for feature extraction and classification of IKONOS multispectral imagery over urban areas," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 4, pp. 654–658, Oct. 2007, doi: 10.1109/LGRS.2007.905121.

[50] F. Zhang, B. Du, L. Zhang, and L. Zhang, "Hierarchical feature learning with dropout k-means for hyperspectral image classification," *Neurocomputing*, vol. 187, pp. 75–82, Apr. 2016, doi: 10.1016/j.neucom.2015.07.132.

[51] J. Li, H. Zhang, and L. Zhang, "A nonlinear multiple feature learning classifier for hyperspectral images with limited training samples," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2728–2738, Jun. 2015, doi: 10.1109/JSTARS.2015.2400634.

[52] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2175–2184, Apr. 2015, doi: 10.1109/TGRS.2014.2357078.

[53] B.-C. Kuo and D. A. Landgrebe, "Nonparametric weighted feature extraction for classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 5, pp. 1096–1105, May 2004.

[54] M. Marcinkiewicz, M. Kawulok, and J. Nalepa, "Segmentation of multispectral data simulated from hyperspectral imagery," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Yokohama, Japan, Jul./Aug. 2019, pp. 3336–3339, doi: 10.1109/IGARSS.2019.8900502.

[55] J. Li, X. Huang, P. Gamba, J. M. Bioucas-Dias, L. Zhang, J. A. Benediktsson, and A. Plaza, "Multiple feature learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1592–1606, Mar. 2015.

[56] J. Xia, N. Falco, J. A. Benediktsson, J. Chanussot, and P. Du, "Class-separation-based rotation forest for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 4, pp. 584–588, Apr. 2016.

[57] A. Setiyoko, I. G. W. S. Dharma, and T. Haryanto, "Recent development of feature extraction and classification multispectral/hyperspectral images: A systematic literature review," *J. Physics: Conf. Ser.*, vol. 801, Mar. 2017, Art. no. 012045.

[58] H. Yang, Q. Du, and G. Chen, "Unsupervised hyperspectral band selection using graphics processing units," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 3, pp. 660–668, Sep. 2011.

[59] S. Li, J. Qiu, X. Yang, H. Liu, D. Wan, and Y. Zhu, "A novel approach to hyperspectral band selection based on spectral shape similarity analysis and fast branch and bound search," *Eng. Appl. Artif. Intell.*, vol. 27, pp. 241–250, Jan. 2014.

[60] F. Li, P. Zhang, and L. Huchuan, "Unsupervised band selection of hyperspectral images via multi-dictionary sparse representation," *IEEE Access*, vol. 6, pp. 71632–71643, 2018.

[61] S. Jia, Z. Ji, Y. Qian, and L. Shen, "Unsupervised band selection for hyperspectral imagery classification without manual band removal," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 531–543, Apr. 2012.

[62] W. Sun, L. Zhang, B. Du, W. Li, and Y. Mark Lai, "Band selection using improved sparse subspace clustering for hyperspectral imagery classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2784–2797, Jun. 2015.

[63] C. Cariou, K. Chehdi, and S. Le Moan, "BandClust: An unsupervised band reduction method for hyperspectral remote sensing," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 3, pp. 565–569, May 2011.

[64] Z. Ren and L. Wu, "Affinity propagation for hyperspectral band selection," in *Proc. IOP Conf. Ser., Earth Environ. Sci.*, vol. 170, Jul. 2018, Art. no. 022061.

[65] I. Delibasoglu and M. Cetin, "Hyperspectral band selection using structural information via hierarchical clustering," *J. Appl. Remote Sens.*, vol. 13, no. 1, pp. 1–10, Mar. 2019.

[66] W. Sun, G. Yang, and J. Li, "Robust multi-feature spectral clusteirng for hyperspectral band selection," in *Proc. IGARSS*, Jul. 2019, pp. 3800–3803.

[67] W. Sun, J. Peng, G. Yang, and Q. Du, "Correntropy-based sparse spectral clustering for hyperspectral band selection," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 3, pp. 484–488, Mar. 2020.

[68] W. Sun, J. Peng, G. Yang, and Q. Du, "Fast and latent low-rank subspace clustering for hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, to be published.

[69] Y. He, D. Liu, and S. Yi, "Recursive spectral similarity measure-based band selection for anomaly detection in hyperspectral imagery," *J. Opt.*, vol. 13, no. 1, Nov. 2010, Art. no. 015401.

[70] B. Guo, S. R. Gunn, R. I. Damper, and J. D. B. Nelson, "Band selection for hyperspectral image classification using mutual information," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 4, pp. 522–526, Oct. 2006.

[71] Q. Du and H. Yang, "Similarity-based unsupervised band selection for hyperspectral image analysis," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 4, pp. 564–568, Oct. 2008.

[72] X. Cao, X. Li, Z. Li, and L. Jiao, "Hyperspectral band selection with objective image quality assessment," *Int. J. Remote Sens.*, vol. 38, no. 12, pp. 3656–3668, Mar. 2017.

[73] A. Datta, S. Ghosh, and A. Ghosh, "Combination of clustering and ranking techniques for unsupervised band selection of hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2814–2823, Jun. 2015.

[74] W. Sun and Q. Du, "Graph-regularized fast and robust principal component analysis for hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3185–3195, Jun. 2018.

[75] W. Sun, G. Yang, J. Peng, and Q. Du, "Hyperspectral band selection using weighted kernel regularization," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 9, pp. 3665–3676, Sep. 2019.

[76] H. Wu, J. Zhu, S. Li, D. Wan, and L. Lin, "A hybrid evolutionary approach to band selection for hyperspectral image classification," in *Advances in Neural Network Research and Applications*, Z. Zeng and J. Wang, Eds. Berlin, Germany: Springer, 2010, pp. 329–336, doi: 10.1007/978-3-642-12990-2_37.

[77] H. Su, Q. Du, G. Chen, and P. Du, "Optimized hyperspectral band selection using particle swarm optimization," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2659–2670, Jun. 2014.

[78] Y. Yuan, G. Zhu, and Q. Wang, "Hyperspectral band selection by multitask sparsity pursuit," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 631–644, Feb. 2015.

[79] H. Su, B. Yong, and Q. Du, "Hyperspectral band selection using improved firefly algorithm," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 1, pp. 68–72, Jan. 2016.

[80] M. Wang, Y. Wan, Z. Ye, X. Gao, and X. Lai, "A band selection method for airborne hyperspectral image based on chaotic binary coded gravitational search algorithm," *Neurocomputing*, vol. 273, pp. 57–67, Jan. 2018.

[81] A. R. Chowdhury, J. Hazra, and P. Dutta, "A hybrid approach for band selection of hyperspectral images," in *Hybrid Intelligence for Image Analysis and Understanding*. Hoboken, NJ, USA: Wiley, 2017, ch. 11, pp. 263–281.

[82] X. Cao, C. Wei, J. Han, and L. Jiao, "Hyperspectral band selection using improved classification map," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 11, pp. 2147–2151, Nov. 2017.

[83] M. Zhang, M. Gong, and Y. Chan, "Hyperspectral band selection based on multi-objective optimization with high information and low redundancy," *Appl. Soft Comput.*, vol. 70, pp. 604–621, Sep. 2018.

[84] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[85] Y. Zhan, D. Hu, H. Xing, and X. Yu, "Hyperspectral band selection based on deep convolutional neural network and distance density," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 12, pp. 2365–2369, Dec. 2017.

[86] I. J. Goodfellow, Y. Bengio, and A. C. Courville, *Deep Learning Adaptive Computation and Machine Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org/

[87] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proc. IEEE CVPR*, Jun. 2015, pp. 5353–5360, doi: 10.1109/CVPR.2015.7299173.

[88] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, Long Beach, CA, USA, Dec. 2017, pp. 5998–6008. [Online]. Available: http://papers.nips.cc/paper/7181-attention-is-all-you-need

[89] B. Hoyle, M. M. Rau, K. Paech, C. Bonnett, S. Seitz, and J. Weller, "Anomaly detection for machine learning redshifts applied to SDSS galaxies," *Monthly Notices Roy. Astronomical Soc.*, vol. 452, no. 4, pp. 4183–4194, Aug. 2015.

[90] P. J. Rousseeuw and K. V. Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212–223, Aug. 1999.

[91] M. L. McHugh, "Interrater reliability: The kappa statistic," *Biochemia Medica*, vol. 3, no. 22, pp. 276–282, 2012.

[92] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980

[93] R. Kemker and C. Kanan, "Self-taught feature learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2693–2705, May 2017.

[94] H. Su, Q. Du, and P. Du, "Hyperspectral image visualization using band selection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2647–2658, Jun. 2014.

[95] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, Jun. 2014. [Online]. Available: https://science.sciencemag.org/content/344/6191/1492

[96] A. Mohan, G. Sapiro, and E. Bosch, "Spatially coherent nonlinear dimensionality reduction and segmentation of hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 2, pp. 206–210, Apr. 2007.

[97] W. Sun, L. Zhang, L. Zhang, and Y. M. Lai, "A dissimilarity-weighted sparse self-representation method for band selection in hyperspectral imagery classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 9, pp. 4374–4388, Sep. 2016.

[98] J. Nalepa, M. Myller, and M. Kawulok, "Training- and test-time data augmentation for hyperspectral image segmentation," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 292–296, Feb. 2020.

[99] J. Nalepa, M. Myller, and M. Kawulok, "Transfer learning for segmenting dimensionally reduced hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, to be published.

[100] N. Audebert, B. Le Saux, and S. Lefèvre, "Generative adversarial networks for realistic synthesis of hyperspectral samples," 2018, *arXiv:1806.02583*. [Online]. Available: http://arxiv.org/abs/1806.02583

[101] J. Nalepa, M. Antoniak, M. Myller, P. Ribalta Lorenzo, and M. Marcinkiewicz, "Towards resource-frugal deep convolutional neural networks for hyperspectral image segmentation," *Microprocessors Microsyst.*, vol. 73, Mar. 2020, Art. no. 102994.

PABLO RIBALTA LORENZO (Student Member, IEEE) is currently pursuing the Ph.D. degree in machine learning at the Silesian University of Technology, Poland. He was an experienced Engineer and a Researcher specialized in the design and implementation of state of the art technology, built around machine learning and AI. He also works as a Deep Learning Engineer at NVIDIA.

LUKASZ TULCZYJEW is currently a Research Software Engineer at KP Labs, Poland, where he focuses on hyperspectral image analysis. His research and engineering interests include machine learning (which focal point being deep learning) and high-performance Python programming, especially in the context of deep learning.

MICHAL MARCINKIEWICZ received the master's degree in physics from the University of Warsaw, and the Ph.D. degree in physics at Fundamental Research in Montpellier, France. He currently works at NVIDIA, where he conducts research in the domain of deep learning, mainly on computer vision, representation learning, optimization, and model compression. He worked on automatic segmentation and processing of medical images, systems for efficient analysis of hyperspectral satellite data, models for fast and accurate classification of natural images, efficient image style transfer, and audio processing and classification. He authored and coauthored more than 20 publications and conference papers across both physics and computer science.

JAKUB NALEPA (Member, IEEE) received the M.Sc. and Ph.D. degrees (Hons.) in computer science from the Silesian University of Technology, Gliwice, Poland, in 2011 and 2016, respectively. He is currently an Assistant Professor with the Silesian University of Technology. He is also with Future Processing, Poland, and KP Labs, Poland, where he acts as a Chief Scientist in the field of machine learning and computer vision. His research interests encompass machine learning, deep learning, evolutionary algorithms, pattern recognition, medical and satellite imaging, and interdisciplinary applications of these methods. He has been involved in several projects related to the above-mentioned domains—in both academia and industry. So far, he has published more than 90 articles in these fields and acts as a reviewer for more than 50 international journals.