A Machine Learning Approach for Material Detection in Hyperspectral Images

Raphaël Marée
GIGA, University of Liège, Belgium

Benjamin Stévens
PEPITe S.A., Belgium

Pierre Geurts
EE & CS, University of Liège, Belgium

Yves Guern
ATIS, France

Philippe Mack
PEPITe S.A., Belgium

Abstract

In this paper we propose a machine learning approach for the detection of gaseous traces in thermal infra red hyperspectral images. It exploits both spectral and spatial information by extracting subcubes and by using extremely randomized trees with multiple outputs as a classifier. Promising results are shown on a dataset of more than 60 hypercubes.

1. Introduction

In this paper we propose a supervised machine learning approach for the detection of specific gaseous traces in thermal infra red (TIR) hyperspectral images. Although invisible to the human eye, different materials (gases, liquids, solids) exhibit unique spectral signatures. Automatic detection of these traces is of great importance, for example for environmental monitoring where possible applications include the detection of oil slicks in the sea after a shipwreck [7] or the detection of industrial gaseous pollutants [5]. In the biomedical field, hyperspectral image analysis approaches could help to detect abnormalities such as skin tumours [2, 6] or other malignancies in tissues [10].

Our proposed approach uses both spatial and spectral information. It takes its inspiration from the methods of Marée et al. [9, 8] and Dumont et al. [3] respectively proposed for the automatic classification and annotation (semantic segmentation) of conventional 2D images. Their method was based on the extraction of local subwindows at random positions in images, their description by raw pixel values, and the use of an ensemble of decision trees technique, namely the extremely randomized trees [4].

In this work, we extend the method of [3] to work with hyperspectral data cubes where a given pixel has tens of wavelengths instead of only the three red, green, and blue components in conventional color images. We also highlight one interesting built-in feature of tree-based algorithms, namely the possibility to rank variables according to their importance for discriminating classes, which allows implicitely to perform spectral band selection without the need of explicit pre-processing or dimensionality reduction techniques. We illustrate the potentials of the approach for the detection of simulated gaseous traces incrusted in real-world scenes under several scenarios.

2. Method

We address the problem of known material detection as a supervised segmentation task: given a sample LS of hypercubes with every pixel (in the x-y position plane) labelled with one class among a finite set of predefined classes (gas types or non-gas), build a classifier able to predict as accurately as possible the class of every pixel of a new, previously unseen, hypercube.

Contrary to many other approaches dedicated to hyperspectral image segmentation, we do not explicitely perform dimensionality reduction or band selection (typically using ICA, PCA, or MNF like in [12]) neither apply any transforms to the hyperspectral images (such as wavelet packet transform in [11]). Instead, we work with raw data and the task of selecting the relevant spectral bands is left to the learning algorithm. Another important aspect in the analysis of hyperspectral images is the exploitation of spatial information for pre-processing or learning [1].

In our work, we propose to adapt the image annotation approach of [3] for the segmentation of hyperspectral images. The main idea of the training stage of this method, transposed to hyperspectral images, is to extract smaller local hypercubes of fixed sizes (called subcubes in what follows) from the learning sample of hypercubes and to use a multiple output learning algorithm to learn a subcube annotation model. Predictions for a new test hypercube is then obtained by aggregating the predictions given by this annotation model for all possible subcubes that the test hypercube contains. Like in the conventional 2D image context, we expect that the use of a multiple output method and the averaging effect at the prediction stage should contribute to provide spatially smooth predictions.

978-1-4244-3993-5/09/$25.00 ©2009 IEEE

The description of the method below has been adapted from [3]. Although rather straightforward, the extension to hyperspectral images proposed here is novel.

2.1. Generation of the learning sample of subcubes

The learning sample is obtained by extracting N subcubes of size $w \times h$ (in the $x-y$ position plane) at random positions in the (bigger) training hypercubes. Each of these subcubes is described, at the input, by $w \times h \times l$ numerical values (where l is the number of spectral bands), and, at the output, by the class of each of its $w \times h$ pixels. By randomly sampling subcubes, we are able to cover large area proportions of hypercubes at a lower cost than exhaustive sampling.

2.2. Learning the subcube annotation model

An annotation model for subcubes is then obtained by using multiple output extremely randomized trees [3].

In (single output) classification, the basic Extra-Trees method [4] grows an ensemble of M (typically $M \in [10; 100]$) unpruned trees, each one being created in a top-down fashion. With respect to other tree-based ensemble methods such as Tree Bagging or Random Forests, Extra-Trees select cut-points at random and use the whole learning sample rather than a bootstrap replica. Their node splitting algorithm depends on two parameters, namely the size K of the random subset of attributes considered at each split, and the minimal (sub)sample size to split a node, $n_{\min}$. The default value recommended in [4] is K equal to the square root of the total number of attributes and $n_{\min} = 2$ (corresponding to fully developed trees).

This algorithm can be adapted to handle multiple classification outputs simply by changing the score measure used to evaluate splits and by modifying the way predictions are computed at tree leaves. The score measure used in our experiments is the average of the standard Gini entropy based score for each output pixel. More precisely, we define:

$$Score(S,T) = \frac{1}{w.h}\sum_{k=1}^{w*h} score^k(S,T) \quad (1)$$

$$Score^k(S,T) = G_C^k(S) - G_{C|T}^k(S) \quad (2)$$

$$G_C^k(S) = \sum_{i=1}^{m} \frac{n_{i.k}}{n_{...}}(1 - \frac{n_{i.k}}{n_{...}}) \quad (3)$$

$$G_{C|T}^k = \sum_{j=1}^{2}\sum_{i=1}^{m} \frac{n_{ijk}}{n_{...}}(1 - \frac{n_{ijk}}{n_{.j.}}), \quad (4)$$

where T is the split to evaluate, S the subsample of size $n_{...}$ associated to the node to split, m is the number of pixel classes, $n_{i.k} (i = 1, ..., m, k = 1, \ldots, wh)$ is the number of subcubes in S of class i at pixel k, n_{i1k} (resp. n_{i2k}) the number of subcubes in S of class i at pixel k and which satisfy (resp. do not satisfy) test T, and $n_{.1.}$ (resp. $n_{.2.}$) the total number of subcubes in S which satisfy (resp. do not satisfy) test T.

Once the tree is built, class probability estimates at each subcubes position could then be obtained at each leaf simply by computing class frequencies over all training subcubes that reach the leaf. However, the size of such a probability vector is equal to the number of pixels times the number of classes and thus the storage of these vectors at each leaf node could require a lot of memory space. To reduce this requirement, we follow [3] and keep track only of the majority class of each pixel together with its confidence as measured by its frequency.

2.3. Prediction of a new test hypercube

To annotate a new test hypercubes, all of its constituting (overlapping) subcubes[1] are extracted and annotated using the subcube annotation model. This annotation assigns a set of tuples *(class number,rate of confidence)* to each pixel of the tested hypercubes (one for each tree and for each subcube that covers this pixel). From this set of predictions, a vector of size m (number of different classes) is computed for each pixel: its i-th element ($i = 1, ..., m$) is the sum of the confidence numbers which are associated to the i-th class among all predictions. The predicted class for the considered pixel is then the one that receives the highest overall confidence.

2.4. Computational complexity

The complexity of the learning stage is $\mathcal{O}(Mwhl KN \log_2 N)$ to grow M trees from N subcubes with the Extra-Trees filtering parameter K. The computation of a prediction for a subcube with M trees requires $\mathcal{O}(M \log_2 N)$ tests. The number of subcubes to test in a hypercube of size $w_i \times h_i$ is equal to $(w_i - w + 1)(h_i - h + 1)$.

2.5. Variable importance ranking

Several importance measures have been proposed in the literature to compute from a tree model a ranking of the input features according to their relevance for predicting the output. In a multiple output context, these measures can be extended to obtain a ranking for each output pixel. However, in the context of our application, we are mainly interested in computing a ranking of the features (ie. spectral bands) for each class (ie. type of gas), independently of the output pixel being predicted. To derive such importance for

[1] Note that the prediction stage can be fasten, at the expense of accuracy, by subsampling the subcubes. To get a prediction for the whole hypercubes, the only constrain is that every pixel must be covered by at least one subcube.

a given class i, we therefore propose to compute at each test node T the following expression:

$$V_i(T) = \frac{1}{w.h} \sum_{k=1}^{w*h} \{n_{i.k}(1 - \frac{n_{i.k}}{n_{...}}) - \sum_{j=1}^{2} n_{ijk}(1 - \frac{n_{ijk}}{n_{.j.}})\}, \quad (5)$$

where the notations are as defined in Section 2.2. $V_i(T)$ is such that $\sum_{i=1}^{m} V_i(T) = n_{...} Score(S,T)$ with $Score(S,T)$ as defined in (2). A split at a tree node is thus considered as more important for class i if it concerns many cases ($n_{...}$ is large) and at the same time discriminates well this class versus the others ($n_{ijk}/n.jk \simeq 0$ or $n_{ijk}/n_{.jk} \simeq 1, \forall j,k$). The overall importance of a feature for class i is then computed by summing the $V_i(T)$ values for all test nodes T of the ensemble of trees where this feature is used to split. For the sake of presentation, one usually normalizes the importances obtained in this way for the different features so that they sum up to 100%.

3. Experiments

3.1. Training data

In our work, the goal is to detect different types of gaseous traces. Such an approach can be useful in many applications eg. chemical accident detection, surveillance of industries, long term pollution follow-up, etc. The sensor technology used in this project focuses on thermal infrared domain, ie. a specific segment of the electromagnetic spectrum (8 to 12 microns). It presents specificities which cannot be exploited in the visible or near infra red domain: traces of gaseous, liquid, or solid chemical materials can be seen as long as their presence is associated to temperature gradients in the surroundings and their light absorption bands is adapted to the wavelength range of observation (which enhances the detection of their presence based on thermal contrasts).

In our case, the imaging sensors measure the spectral signatures in real-world scenes. 160×100 images are obtained where each pixel spectral response is described by 94 spectral bands with wavelength ranging from 7.76 m to 12.98 m. Various gases are simulated under various conditions and incrusted in these cubes, as illustrated in Figure 1. Our Java implementation works with hypercubes in the ENVI format and of dimensions $width \times height \times wavelengths$ = $160 \times 100 \times 94$.

In our experiments, we work with three gases: SF6 (Sulfur hexafluoride), Acetone, and Methanol. For each of them, we have in the training set 21 hypercubes where the gas was encrusted. Ten of these cubes depict a ground scene (rock and soil) and were acquired during the middle of afternoon, with an ambient temperature of 28°C, the sensor positioned at a height of 12 m compared to the ground, and a distance between the sensor and the observed ground of

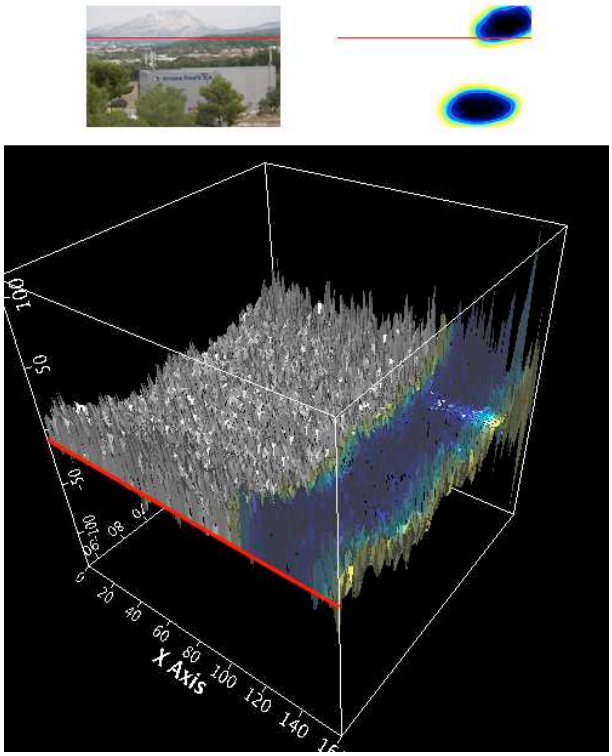

Figure 1. An original scene (top left), the ground truth of two SF6 gas clouds incrusted (top right), and the resulting hypercube (bottom), more precisely the x-position axis and l-wavelength axis (y axis is fixed at the red line position).

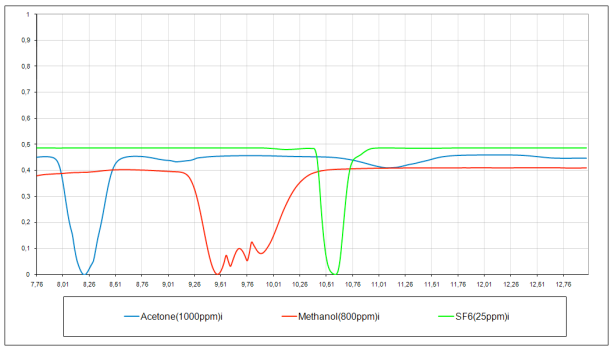

Figure 2. Acetone, Methanol, and SF6 theoretical spectral responses between 7.76 m and 12.98 m.

50 meters. In these cubes, a gas cloud was simulated with various concentrations. We also include in the training set 10 other cubes from a landscape scene with various thermal contrasts. Each of these cubes contains one gas cloud and was acquired at the end of afternoon. The visibility was 20 km and the sky is cloudy at horizon. The temperature at the horizon was 19°C. The first plan is situated at 100 meters. The second plan is situated at 10 km and the sky, by definition, is modelled at 100 km. Finally, in the last street scene, a very big cloud was incrusted. The cube was acquired in

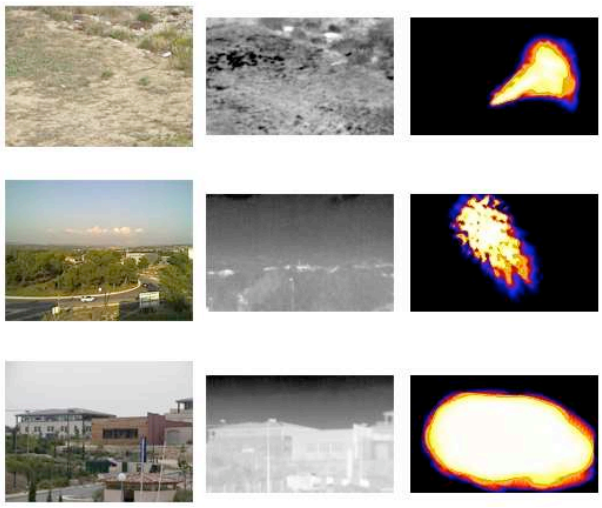

Figure 3. Original pictures of the three training scenes (left), infrared images at 7.76 m (middle), and ground truths (right). In the training set, for each gas, 10 cubes are generated from the ground scene (top), 10 from the landscape scene (middle), 1 cube from the street scene (bottom).

the middle of afternoon. The visibility was 23 km. The temperature at the horizon was 27°C. The first plan (building) is situated at 200 m (meters) and the sky, by definition, is modelled at 100 km. The observed scenes and simulated gas clouds are illustrated in Figure 3. Before applying the learning method, the ground truths are binarized and pixels are annotated as non-gas (black), Acetone (blue), Methanol (red) or SF6 (white). A multi-class model is then built.

3.2. Ranking of relevant spectral bands

In this first experiment, we illustrate the implicit spectral band selection of the learning algorithm. As the gases we are working on are simulated, we know in advance which spectral bands are specific for each gas (see Figure 2). The purpose of this experiment is thus to assess if these bands are ranked first by the learning algorithm (see Section 2.5) applied to discriminate gas types and non-gas in the training set of 63 hypercubes.

Despite the various concentrations, temperatures, and background scenes in the training dataset resulting in rather different hypercubes, the algorithm succeeds to detect spectral bands specific to these three gaseous traces, as shown by results in Table 1 in agreement with Figure 2. These results were obtained using 5000 training subcubes of size $3 \times 3 \times 94$ randomly sampled in each hypercube (thus a total of 315000 training subcubes), $M = 10$ trees, and the Extra-Trees parameter K equal to the total number of input attributes ($3 \times 3 \times 94 = 846$). The model has a total complexity of 438030 nodes and takes 20Mo (3.15Mo if compressed) on disk.

In some applications, this functionality might be useful to characterize materials for which the spectral signatures are unknown in advance (e.g. tumors). It might also be used as a feature selection/dimensionality reduction procedure when the total amount of data is too large.

Wavelength	Importance
SF6	
70 (10.56 m)	16.45%
69 (10.52 m)	9.82%
71 (10.60 m)	9.44%
72 (10.64 m)	8.31%
0 (7.76 m)	5.21%
68 (10.48 m)	2.55%
73 (10.68 m)	1.58%
59 (9.72 m)	1.12%
Methanol	
0 (7.76 m)	6.87%
44 (9.52 m)	6.35%
42 (9.44 m)	5.51%
43 (9.48 m)	5.38%
45 (9.56 m)	3.98%
46 (9.60 m)	3.03%
48 (9.68 m)	2.48%
41 (9.40 m)	2.34%
47 (9.64 m)	1.74%
49 (9.72 m)	1.36%
51 (9.80 m)	1.17%
50 (9.76 m)	1.11%
52 (9.84 m)	1.11%
54 (9.92 m)	1.02%
53 (9.88 m)	1.01%
Acetone	
11 (8.20 m)	9.08%
12 (8.24 m)	8.10%
13 (8.28 m)	5.21%
10 (8.16 m)	4.16%
0 (7.76 m)	3.37%
14 (8.32 m)	2.17%
9 (8.12 m)	1.90%
44 (9.52 m)	1.07%
46 (9.60 m)	1.05%
43 (9.48 m)	1.05%
71 (10.60 m)	1.04%
70 (10.56 m)	1.02%

Table 1. Variable ranking for each gas class (only variables with importance $> 1\%$ are shown).

3.3. Automatic segmentation

We present here multi-class prediction results obtained automatically on hypercubes that were not included in the training set.

3.3.1 Known scenes but unknown conditions

In the first batch of experiments, the test cubes are from the same scenes (the ground scene and the background scene), but the gas clouds were simulated under different conditions (namely concentrations or temperatures are different). Figure 4 illustrates the results obtained with subcubes of sizes $w = h = 1$ (left) and $w = h = 3$ (right). Both models give good results, probably because the variations of the test hypercubes compared to the training hypercubes are not drastic. However, we observe that thanks to the use of spatial information through 3×3 subcubes, predictions are smoother compared to the case where only individual pixel spectral signatures are used. Indeed, this latter tends to produce isolated false predictions and erratic gas cloud borders. These results were obtained using 5000 training subcubes randomly sampled in each of the 63 hypercubes, $M = 10$ fully developed trees, a number of random tests equal to the rounded square root of the total number of input attributes ($\sqrt{1 \times 1 \times 94} = 10$ for 1×1 subcubes, $\sqrt{3 \times 3 \times 94} = 29$ for 3×3 subcubes). Let us emphasize here the fact that we are dealing with a training database of $63 \times 5000 \times 3 \times 3 \times 94 = 266,490,000$ floating point values. In our current unoptimized implementation, the Extra-Trees subcube model is built in slightly less than 9 minutes on a 2.4Ghz computer and require less than 2Gb of running memory. The model has a total complexity of 543056 nodes. Loading and predicting a new hypercube is rather fast (less than 5 seconds) as it only involves propagating its subcubes through the ensemble of trees where internal nodes simply test raw spectral responses within subcubes. It is also worth noting that the method is highly parallelizable. Indeed, tree induction, subcube extraction and their propagation in trees are processes that could be run independently and their results subsequently aggregated.

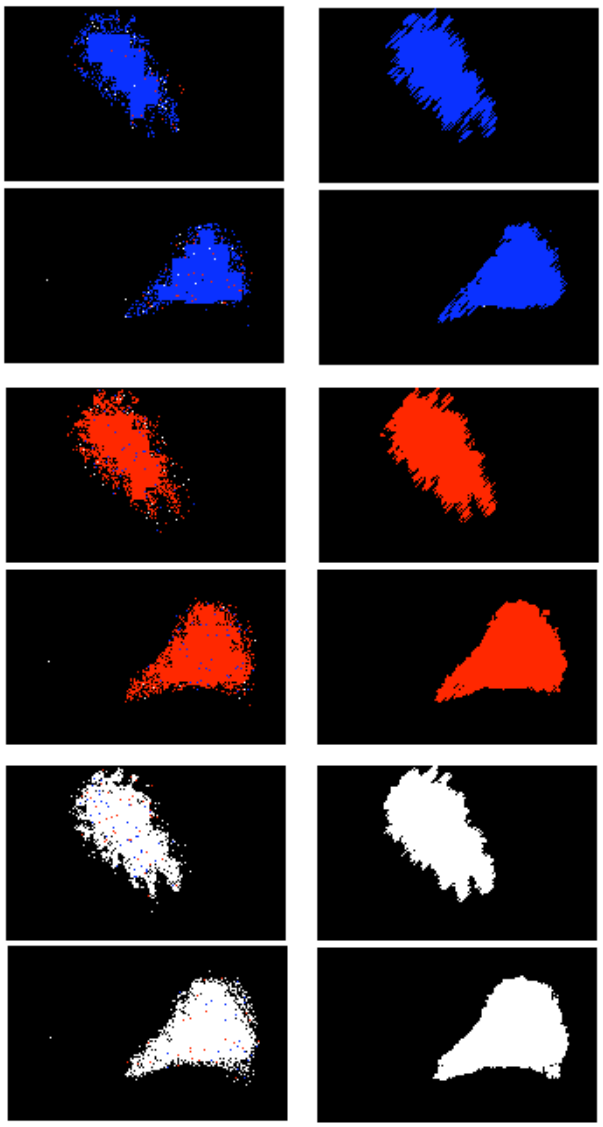

Figure 4. Predictions for Acetone, Methanol, SF6 test cubes of known scenes but unknown conditions. Left: 1×1 subcubes, Right: 3×3 subcubes.

3.3.2 Unknown scene and unknown conditions

In the last experiment, we use the model built on the 63 training cubes and try to detect two simulated gaseous traces in an hypercube of an unknown, grass, scene. More precisely, for this new scene, for each of the three gas we have one test hypercube where two gas clouds were incrusted to different concentrations: a cloud with a low concentration, and a cloud with saturated absorption, as illustrated in Figure 5. The saturation of gas warps the principal absorption peak and reveals sometimes secondary absorption peaks. The cube was acquired in the middle of afternoon. The ambient temperature during the acquisition was 23°C. The sensor is positioned at a height of 12 m compared to the ground. The distance between the sensor and the observed grass is 30 m.

Figure 5 show results for the hypercube containing SF6 gaseous traces (similar results are obtained for hypercubes with Methanol and Acetone). Although results are very noisy with the 1×1 annotation model and the second gas cloud is mostly missed, the predictions are getting better with models based on increased subcube sizes. With 10×10 subcubes, the two gas clouds are clearly detected and separated from the non-gas background. Regarding confidence maps, we observe that there is a lot of uncertainty with individual pixel classifier predictions. With larger subcube annotation models, the method is more confident (higher probability estimates) in most of the non-gas regions and for the cloud with higher concentration, but less confident for the lower concentration gas cloud and at the borders of both clouds. Overall, using subcubes allows to reduce

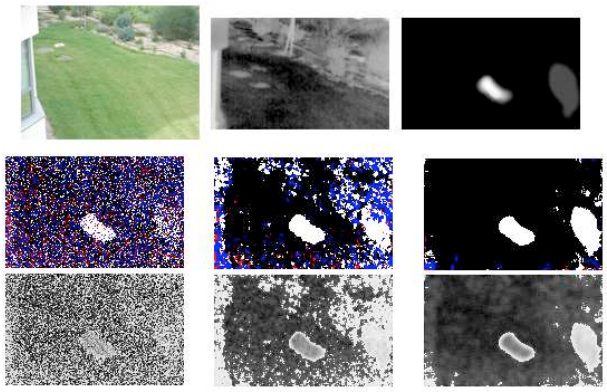

Figure 5. Top: Original image of the unknown scene, infrared image at 7.76 m, and the ground truth. Middle: predictions obtained using models built on subcubes of sizes 1×1, 3×3, or 10×10 (from left to right). Bottom: confidence maps (black is the most confident, white is the less confident).

noise in the predictions and improves spatial coherence, as well as increases the confidences. Post-processing (e.g. using thresholds on confidences, constraints on the connected component sizes, or further spatial smoothing) might be investigated to further reduce false detections.

4. Discussion

In this paper, we proposed a generic machine-learning approach for the detection of known materials in hyperspectral images. It was applied for the detection of simulated gaseous traces in thermal infra red hyperspectral images of real-world scenes. The method works directly on hypercube raw data without signal pre-processing and exploits spectral and spatial information. We have shown promising qualitative results on a database of more than 60 hyperspectral cubes as well as the learning algorithm's capability to perform spectral band selection, and its low computational complexity. Future work is needed to evaluate quantitatively the approach on even more data and compare its performances with other methods. Extension of the model to deal with numerical outputs (to predict gas concentration) instead of discrete outputs is also thinkable.

5. Acknowledgments

This work presents results of the European FP6 project HAWKEYE (contract n° SST4-CT-2005-516168). Raphaël Marée is supported by the GIGA (University of Liège) with the help of the Walloon Region and the European Regional Development Fund. Pierre Geurts is a research associate of the F.R.S.-FNRS, Belgium.

References

[1] G. Camps-Valls, L. Gomez-Chova, J. Muoz-Mar, J. Vila-Francs, and J. Calpe-Maravilla. Composite kernels for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 2006.

[2] Z. Du, M. K. Jeong, and S. G. Kong. Band selection of hyperspectral images for automatic detection of poultry skin tumors. *IEEE Trans. Automation Science and Engineering*, 4(3):332–339, 2007.

[3] M. Dumont, R. Marée, L. Wehenkel, and P. Geurts. Fast multi-class image annotation with random subwindows and multiple output randomized trees. In *Proc. International Conference on Computer Vision Theory and Applications (VISAPP)*, volume 2, pages 196–203. INSTICC, feb 2009.

[4] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine Learning*, 36(1):3–42, 2006.

[5] E. Hirsch and E. Agassi. Detection of gaseous plumes in ir hyperspectral images using hierarchical clustering. *Applied Optics*, 46:6368–6374, 2007.

[6] S. G. Kong and L.-J. Park. *Augmented Vision Perception in Infrared*, volume 3, chapter Hyperspectral Image Analysis for Skin Tumor Detection, pages 155–171. Springer, springer edition, 2009.

[7] M. Lennon, S. Babichenko, N. Thomas, G. Mercier, and V. Mariette. Oil slick detection and characterization by satellite and airbone sensors: experimental results with sar hyperspectral and lidar data. In *IEEE International Geoscience And Remote Sensing Symposium (IGARSS)*, 2005.

[8] R. Marée, P. Geurts, J. Piater, and L. Wehenkel. Random subwindows for robust image classification. In *Proc. IEEE CVPR*, volume 1, pages 34–40. IEEE, 2005.

[9] R. Marée, P. Geurts, G. Visimberga, J. Piater, and L. Wehenkel. An empirical comparison of machine learning algorithms for generic image classification. In F. Coenen, A. Preece, and A. Macintosh, editors, *Proceedings of the 23rd SGAI international conference on innovative techniques and applications of artificial intelligence, Research and development in intelligent systems XX,*, pages 169–182. Springer, 2003.

[10] K. Masood and N. Rajpoot. Classification of colon biopsy samples by spatial analysis of a single spectral band from its hyperspectral cube. In *Proc. Medical Image Understanding and Analysis*, 2007.

[11] M. Z. Salvador, R. G. Resmini, and R. B. Gomez. Hyperspectral trace gas detection using the wavelet packet transform. In *Proc. SPIE Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XIV*, 2008.

[12] J. Wang and C.-I. Chang. Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis. *IEEE transactions on geoscience and remote sensing*, 44(6):1586–1600, 2006.