MINIMAL LEARNING MACHINE IN ANOMALY DETECTION FROM HYPERSPECTRAL IMAGES

Ilkka Pölönen[1]*, Kimmo Riihiaho[1], Anna-Maria Hakola[1], Leevi Annala[1]

[1] Faculty of Information Technology, University of Jyväskylä, 40100, Jyväskylä, Finland
(ilkka.polonen, kimmo.a.riihiaho, anna.m.hakola, leevi.a.annala)@jyu.fi

Commission III, WG III/4

KEY WORDS: Minimal Learning Machine, Hyperspectral Imaging, Anomaly Detection, Remote Sensing

ABSTRACT:

Anomaly detection from hyperspectral data needs computationally efficient methods to process the data when the data gathering platform is a drone or a cube satellite. In this study, we introduce a minimal learning machine for hyperspectral anomaly detection. Minimal learning machine is a novel distance-based classification algorithm, which is now modified to detect anomalies. Besides being computationally efficient, minimal learning machine is also easy to implement. Based on the results, we show that minimal learning machine is efficient in detecting global anomalies from the hyperspectral data with low false alarm rate.

1. INTRODUCTION

Anomaly detection, often referred also as outlier detection, is a vital application when we are looking for something abnormal. To detect these anomalies, we have also to understand what is normal. We might have a dataset, which we know with certainty that there is nothing unusual in it. Thus, we can use this dataset to teach normal behaviour to the anomaly detection method. This is an example of semi-supervised learning (Chandola et al., 2009). Anomalies can also be detected in un-supervised manner, but then we have to make some assumptions about the data. These assumptions can be, for example that the majority of the data points are presenting normal behaviour or the anomalies has a sparse neighbourhood. In this study, we introduce a new semi-supervised anomaly detection method for the hyperspectral data, which is based on the minimal learning machine (de Souza Junior et al., 2015).

Our objective is to decrease the complexity of the training process and actual anomaly detection. In the era of deep learning complexity, of machine learning models has been increasing all the time. Simultaneously popularity of hyperspectral imaging has increased, because of the new smaller and cheaper imagers. Many potential applications of the hyperspectral imaging use complex machine learning models to predict or estimate parameters. If an imager is mounted on some autonomous platform such as a drone, it will need to either storage a huge amount of data or run machine learning models in real-time, while there are restrictions with payload and available energy. Anomaly detection is one application where model training and detection should be done in real-time.

With hyperspectral images there exists large variability of anomaly detection algorithms from Mahalanobis distance-based RX -method (Reed, Yu, 1990) and support vector approaches (Banerjee et al., 2006) to deep autoencoders (Zhao et al., 2017). The hyperspectral image itself can include several different kinds of anomalies. Point anomalies are single data points, which somehow stand out from the normal behaviour of the data (Chandola et al., 2009). In Figure 1, anomalies are point anomalies.

* Corresponding author

In this study, we use our proposed method to detect point anomalies. Other types of anomalies that can be found from a hyperspectral image, are spatial and spatiotemporal anomalies.

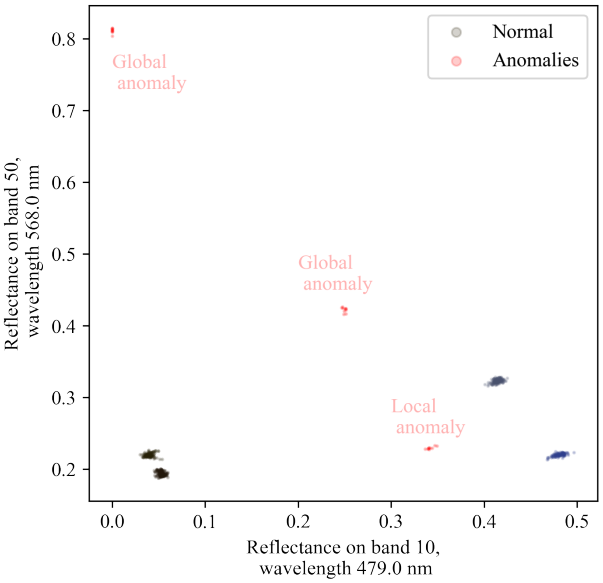

Figure 1. Example of two bands from hyperspectral test data. Majority of data points are in normal behaviour and minority are anomalies. Dataset contains both global and local anomalies.

In Figure 1, we can also see two different types of point anomalies. These are global and local anomalies. Global anomalies are clearly distinguishable from the normal dataset. Local anomalies are geometrically inside of the normal dataset, or normal dataset somehow shadows it. In general, global anomalies are easier to detect than local ones.

In the following sections, we will explain how minimal learning machine works in the hyperspectral anomaly detection. We will study its performance and compare it to the other well-known methods.

2. METHODS AND MATERIAL

2.1 Minimal learning machine

Minimal learning machine (MLM) is distance-based classification method, which offers tools to create computationally cheap training and classification (de Souza Junior et al., 2015). MLM utilises linear mapping between input and output distances. In case of the hyperspectral images, these distances would be $d(\mathbf{x}_i, \mathbf{m}_k)$ and $\delta(\mathbf{y}_i, \mathbf{t}_k)$, where $\mathbf{x}_i \in X \subset \mathbb{R}^D$ are training set of spectra with D wavebands and $\mathbf{m}_k \in R$ are randomly sampled subset of X and correspondingly $\mathbf{y}_i \in Y \subset \mathbb{R}$ are labels of training set and $\mathbf{t}_k \in T$ are subset of Y. Training set X consist of N samples, and subset R has K samples. Now, we define two matrices based on these distances $\mathbf{\Delta}_y \in \mathbb{R}^{N\times K}$ and $\mathbf{D}_x \in \mathbb{R}^{N\times K}$. By assuming the linear mapping between these two distance matrices, we have a linear model

$$\mathbf{\Delta}_y = \mathbf{D}_x\mathbf{B} + \mathbf{E}, \tag{1}$$

where $\mathbf{B}$ is coefficients and $\mathbf{E}$ is residual. Coefficients $\mathbf{B}$ can be approximated using ordinary least squares estimator

$$\widehat{\mathbf{B}} = (\mathbf{D}_x^T\mathbf{D}_x)^{-1}\mathbf{D}_x^T\mathbf{\Delta}_y. \tag{2}$$

Now $\widehat{\mathbf{B}}$ is linear model between distances $\delta(\mathbf{y}_i, \mathbf{t}_k)$ and $d(\mathbf{x}_i, \mathbf{m}_k)$. distances between new spectrum $\mathbf{x}_n$ and its label $\mathbf{y}_n$ is

$$\delta(\mathbf{y}_n, T) = d(\mathbf{x}_n, R)\widehat{\mathbf{B}}. \tag{3}$$

Outputs $\mathbf{y}_n$ can be estimated by solving optimisation problem

$$\min_{\mathbf{y}_n} \sum_{k=1}^{K} \Big((\mathbf{y}_n - \mathbf{t}_k)^T(\mathbf{y}_n - \mathbf{t}_k) - \delta^2(\mathbf{y}_n, T) \Big)^2. \tag{4}$$

For anomaly detection, we do not have to estimate $\mathbf{y}_n$, which is computationally most expensive part of the classification. If $\mathbf{x}_n$ is inside of the training set, it means that $\mathbf{y}_n$ is nearby points in subset T. Thus, the distribution of estimated distances $\delta(\mathbf{y}_n, T)$ should be relatively similar to training phases distances in $\mathbf{\Delta}_y$. If $\mathbf{x}_n$ is anomaly or outlier, it should be detected already in $\delta(\mathbf{y}_n, T)$. Now, it is enough to study the behaviour of $\delta(\mathbf{y}_n, T)$. Here we use L2 -norm

$$\|\delta(\mathbf{y}_n, T)\|^2 \tag{5}$$

and variance

$$\text{Var}\left(\delta(\mathbf{y}_n, T)\right) \tag{6}$$

to detect anomalies. For these two values, we set threshold values that reveal anomalous spectra from the dataset. Because the computationally heaviest part, calculating $\mathbf{y}_n$, is not done, the computational complexity of anomaly detection is near $\mathcal{O}(NK)$, which is in relation with the number of samples in R.

Because we are using anomaly detection for the hyperspectral data, both cosine and euclidean distances are studied when we calculate distances $d(\mathbf{x}_i, R)$ and $d(\mathbf{x}_n, R)$ and compare the performance of the method. As an angle based distance, cosine distance is more robust for intensity changes in the spectral data.

Implementation of the MLM for anomaly detection is done using python programming language and libraries: numpy and scipy.

2.2 Reference methods

To test MLM anomaly detection capabilities, we compare it to existing and well-known methods: one class support vector machine (OC-SVM), isolation forest (IsF), global RX algorithm and local RX algorithm. Both OC-SVM and IsF can be used in unsupervised way, but here we use those in semi-supervised manner.

In the OC-SVM, the algorithm tries to find one decision boundary for the whole dataset (Manevitz, Yousef, 2001). In the most simple case, we could fit a hypersphere to the training data by alternating its radius and center. When we consider hyperspectral data, such as shown in Figure 1, we notice that one hypersphere is too coarse for this kind of dataset. Thus, here we utilise gaussian kernel, which allows us to define more complex decision boundaries. Because we are using the OC-SVM in a semisupervised manner, during the training, we are leaving a margin of the decision boundary ν to relatively small. This means that boundary is following quite tightly the outlines of the training set.

The IsF is derivative from the random forest (Liu et al., 2008). It isolates observations by first selecting a feature randomly and then splitting the data based on a random value between the maximum and minimum values of the selected feature. This is continued in a recursive manner until some selected depth of the tree, or the maximum depth of the implementation is achieved. Here anomalies have noticeably shorter path lengths. Again, because of the semi-supervised learning, we are setting relatively low contamination rate, which is regulating the proportion of anomalies in the training dataset.

Both RX algorithms are using the Mahalanobis distance to detect anomalies (Reed, Yu, 1990). The Mahalanobis distance measures distance between a single data point x and distribution of the dataset Y by

$$d(x, Y) = \sqrt{(x-\mu)^T C^{-1}(x-\mu)}, \tag{7}$$

where μ is mean of data points in Y and C is the covariance matrix. In the global RX dataset Y is the whole spectral image or in semisupervised case, the training data. The local RX can be used only in an unsupervised manner by taking pixel's surrounding neighbourhood to the Y. In the local RX, spectral image is gone trough by sliding window manner. Anomalies in both cases are detected by setting up a threshold value. When the distance is great, it is more likely that data point is anomalous.

For OC-SVM and IsF we are using existing implementations from Scikit-learn Python library (Pedregosa et al., 2011). RX algorithms are implemented by using the numpy and scipy libraries.

2.3 Artificial data

Methods are tested with two datasets. The first dataset is a sub set from X-Rite's ColorChecker containing four colours. Data is captured using a visible and near-infrared hyperspectral camera, which is manufactured by VTT (Saari et al., 2013). The dataset has 100 wavebands from 450 nm to 750 nm. There are separate training set and test set. In figure 1, test dataset is illustrated based on two wavebands. In Figure 2, there is a visualization of the training set and corresponding spectra. From Figures 1 and 2 we can see that there is some fluctuation even

between the data points of the same colour. In the training set we have 2500 data points and each class has 625 data points.

The training set contains only normal behaviour, while in the test set, there are 30 anomalous data points. Otherwise test set has similar dimensions as the training set — containing 2500 data points. These are also subset from the same colour of the colorchecker including same colours as the training set, but they are from a different spatial location. Anomalous pixels are randomly located in the test set. These are taken from different parts of the colorchekcer. Figure 3 shows spectra of all anomalous pixels. There are three subsets of anomalies: two of them are behaving like global anomalies, and one is a local anomaly, as Figure 1 illustrates.

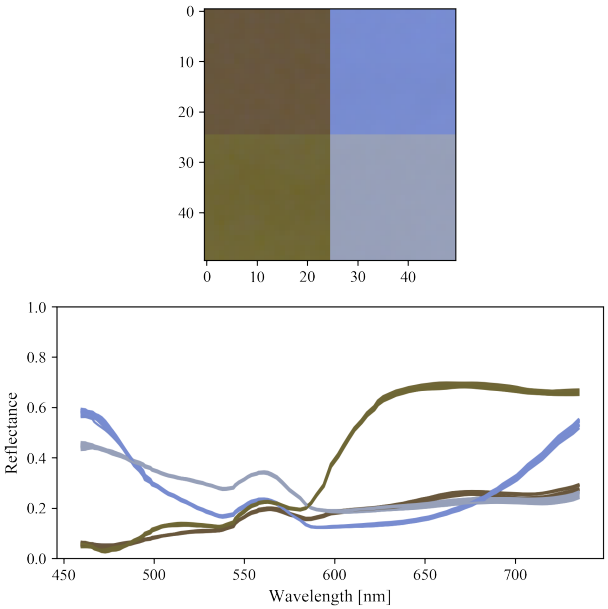

Figure 2. Illustration of the training dataset. Above there is "RGB" presentation of the dataset in the spatial dimensions. Below there are sample reflectance spectra from each class.

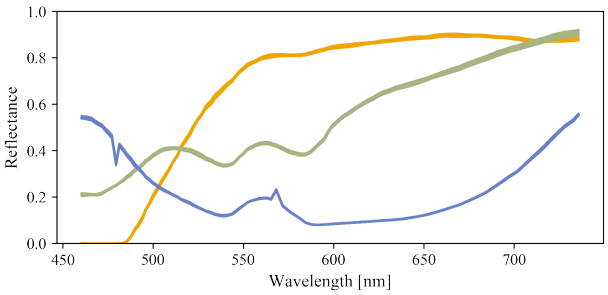

Figure 3. Spectra of anomalous pixels. These are randomly distributed to the test dataset, which has same dimensions as the training set has ($50 \times 50 \times 100$).

2.4 Forest data

The second dataset is from the Finnish forests, where the main tree species are pines, spruces and birches. Training dataset ($1500 \times 1400 \times 38$) contains mainly forest, some grass area and forest road. This dataset is a subset of a larger dataset, which has been previously used in tree species classification (Nevalainen et al., 2017, Pölönen et al., 2018, Nezami et al., 2020). Description of the dataset in details can be found from (Nevalainen et al., 2017). The dataset has high spatial resolution 9 cm ground sampling distance (GSD) and 38 spectral bands from 507 nm to 820 nm. A narrowband RGB image of the training dataset is shown in Figure 4. For actual training, we randomly selected 100000 samples from the image. From this dataset, we selected randomly 100 spectra to be a subset R.

The test set is from the same remotely sensed dataset. It includes similar features as the training set, but there are some anomalous objects. There are three reflectance panels (size 1 m 1 m with a nominal reflectivity of 0.03, 0.1 and 0.5), one black panel, one blue van, and cross-shaped georeferencing signal with an arm length of 3 m and width of 30 cm. These are circulated with red in Figure 5. Spectra of van, reflectance panel and the forest are shown in Figure 6.

Because the training dataset does not have labels for the whole image, we performed k-means clustering (k=3) to produce needed labels. This has not been included in training time, because there are alternative ways to produce these labels.

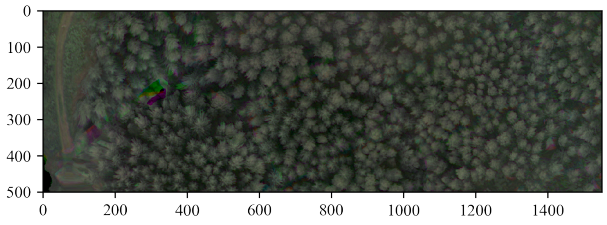

Figure 4. Training data for the anomaly detection from the forest. Data has been captured from the UAV with 9 cm GSD.

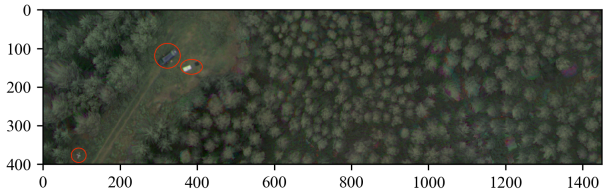

Figure 5. Test dataset for the anomaly detection from the forest. Inside of red circulated areas are anomalous objects (three reflectance panels, one black panel, one blue van, and cross-shaped georeferencing signal).

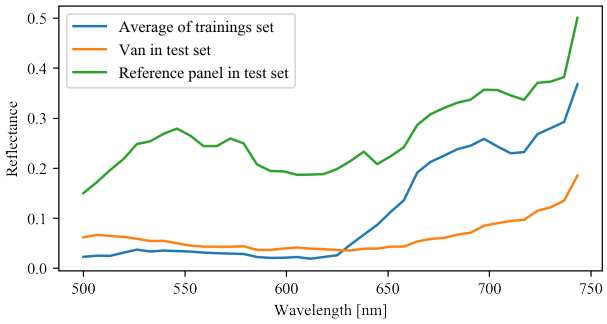

Figure 6. Mean spectra of training dataset and samples from the van and the reflectance panel (nominal reflectivitiy of 0.5).

3. RESULTS

An artificial dataset was used to evaluate the performance of MLM by varying the size of the actual training set and the portion of set R from the training set. In a computational sense, we measured both training and detection time, and compared those to the reference methods. As a measure of accuracy, we calculated the area under curve (AUC) and draw the receiver operating characteristic curve (ROC) for best performing MLM setup and reference methods. Threshold for anomaly detection was set for the variance to > 70 % and the L2-norm to > 50 % of maximum value. For the global RX threshold was set to > 3.5 and for the local RX to > 5.

MLM description in section 2.1 shows that in the training and the detection, the most influential factor is the size of subset R. This is shown in Figures 7–10, where we compare training and detection time against the size of the training dataset and the subset R. Results for the euclidean and cosine distances are reported separately. From Figures 7–10 we can see that selected distance metric has an effect on training time: using cosine distance is slower than using Euclidean distance.

In the case of AUC, cosine distance seems to perform better. From Figures 11 and 12, we can see that for anomaly detection, AUC of L2-norm and variance seems to produce quite similar results. With L2-norm, AUC has less fluctuation than with variance. The variance of cosine distance is giving higher AUC than L2-norm, but in the case of Euclidean distance, L2-norm is less affected by the size of training sets and subset R.

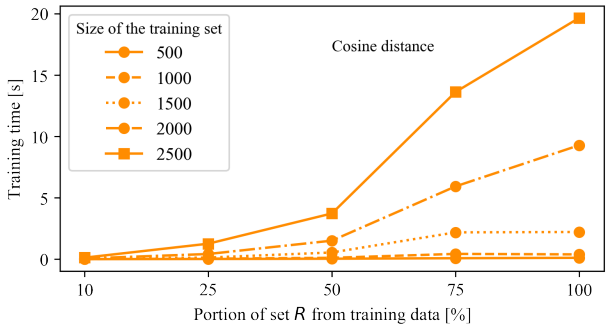

Figure 7. Training time of the MLM using cosine distance varying size of the training set and proportion subset R.

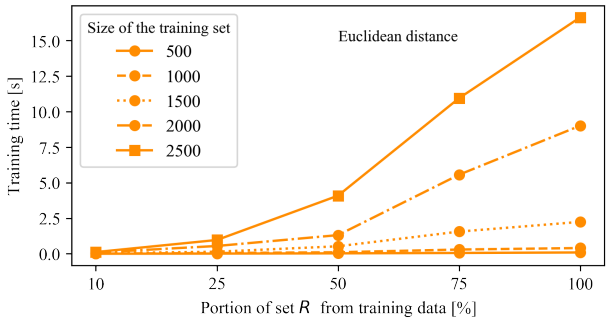

Figure 8. Training time of the MLM using Euclidean distance varying size of the training set and proportion subset R.

In the Table 1 we compare results of MLM (cosine, variance, training set size 2500, R size 250) to the reference methods.

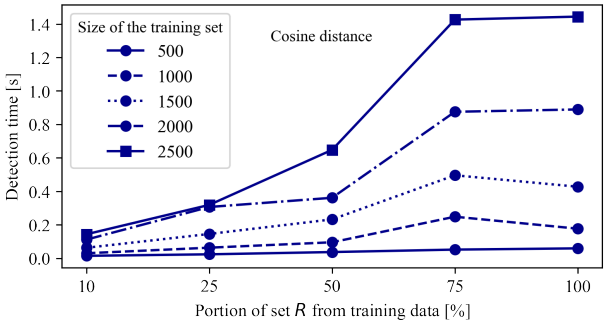

Figure 9. Anomaly detection time of the MLM using cosine distance varying size of the training set and proportion subset R.

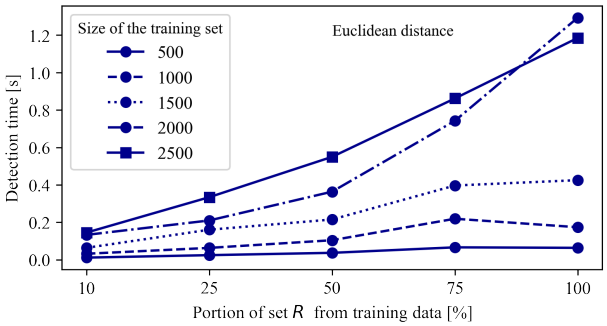

Figure 10. Anomaly detection time of the MLM using Euclidean distance varying size of the training set and proportion subset R.

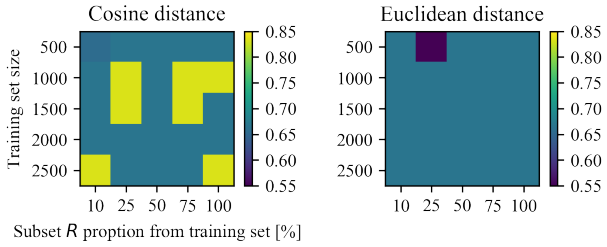

Figure 11. Area under curve value of the MLM using L2-norm with cosine distance (left) and Euclidean distance (right) varying size of the training set and proportion subset R.

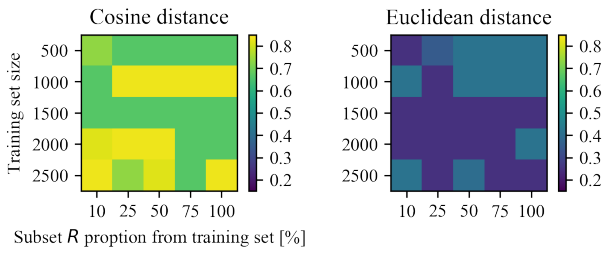

Figure 12. Area under curve value of the MLM using variance with cosine distance (left) and Euclidean distance (right) varying size of the training set and proportion subset R.

With a small dataset, OC-SVM is outperforming MLM in training and anomaly detection. IsF is slower than MLM in both categories. Both global and local XR are slower in the detection

of anomalies. Training of the global XR is faster, but it only calculates covariance matrix and means of the training set.

In Figure 13 ROC curves and AUC are compared with MLM and reference methods. It reveals us that OC-SVM and IsF have higher AUC than MLM. It seems that MLM is not capable of detecting local anomalies, while OC-SVM and IsF are. Both RX algorithms give moderate results. The main difference between MLM and other methods presented in this study is that with MLM, there are not any false alarms. This can also be seen from Figure 14, which shows anomaly detection maps of the test set.

	A	B	C	D	E
Training time [s]	0.112	0.005	0.492	0.007	-
Testing time [s]	0.059	0.003	0.171	0.060	0.435

Table 1. Comparison of the computation times for the artificial dataset. A: MLM (cosine, variance, training set size 2500, R size 250), B: One-Class-SVM, C:Isolation forest, D: Global RX, E: Local RX.

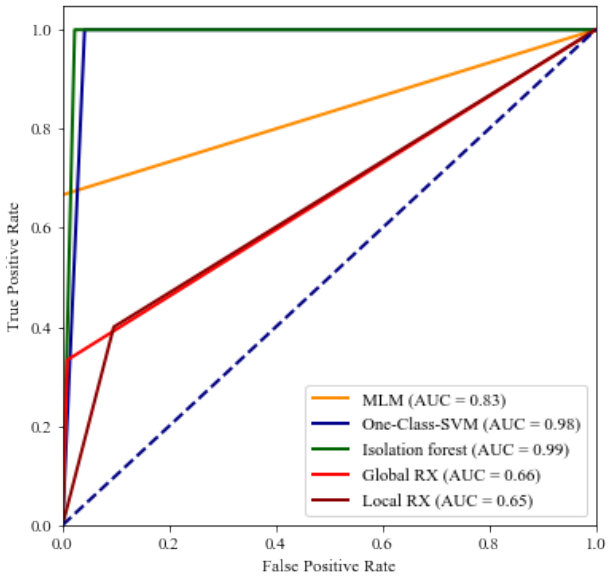

Figure 13. Comparison of the ROC-curves of the MLM and reference methods. ROC curve of the MLM shows that it does not have any false positives. MLM cannot find local anomalies.

We made only visual comparison to the results of the forest dataset. For the MLM, we used variance with the threshold > 0.4. Global and local RX threshold values were respectively > 0.8 and > 0.1.

From Table 2, we can see that in the case of larger datasets, MLM is outperforming all reference methods in testing, and both OC-SVM and IsF in training. Also, visual results in Figure 15 show that MLM is more capable of detecting anomalies from the dataset. MLM is capable of separating all anomalies with relatively low false alarm rate. Both RX methods seem to be useless. OC-SVM finds two reference panels and IsF finds part of the van, georeferencing signal and one reference panel. Both have a higher false alarm rate than MLM has.

All computations were done using MacBook Pro (Mid-2014) with 3 GHz Intel Core i7 processor and 16 GB memory.

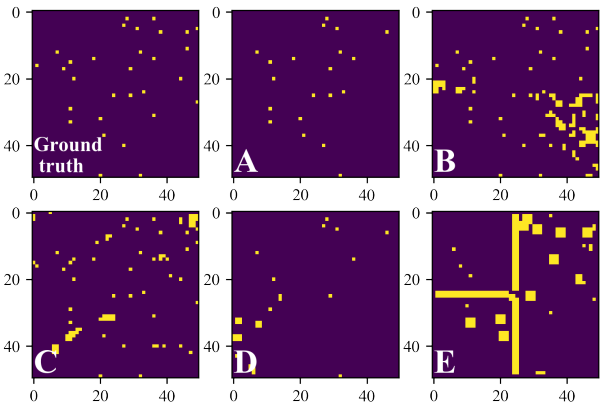

Figure 14. Anomaly detection maps of the MLM and reference methods for the artificial dataset. A: MLM (cosine, variance, training set size 2500, R size 250), B: One-Class-SVM, C: Isolation forest, D: Global RX, E: Local RX

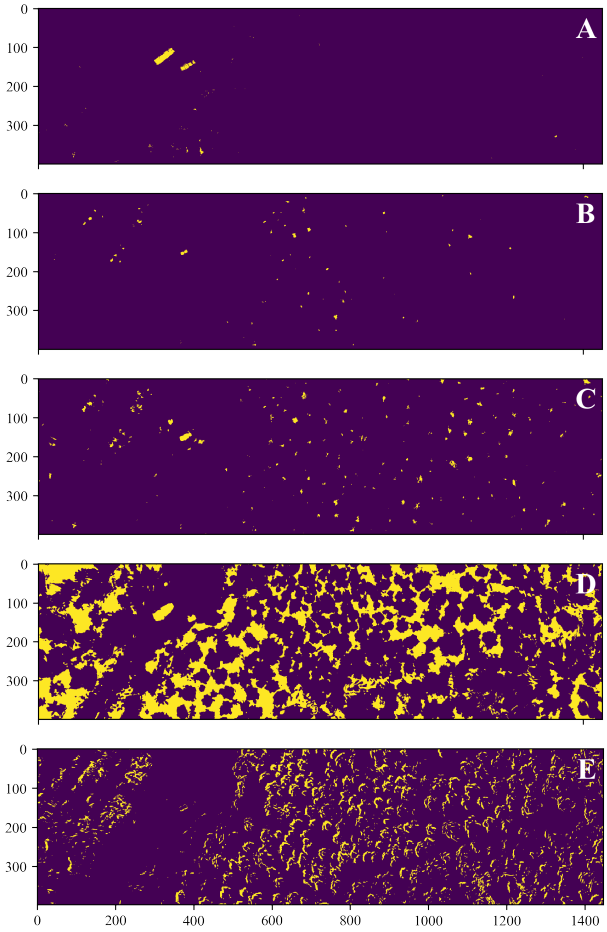

Figure 15. Anomaly detection maps of the MLM and reference methods for the forest dataset. A: MLM (cosine, variance, training set size 2500, R size 250), B: One-Class-SVM, C: Isolation forest, D: Global RX, E: Local RX.

4. DISCUSSION

Overall, the results show that the main benefits of the MLM are the low false alarm rate, the computational efficiency, and the extremely easy implementation using distance functions and

	A	B	C	D	E
Training time [s]	0.546	2.589	9.445	0.0681	-
Testing time [s]	2.953	5.727	25.18	8.684	60.17

Table 2. Comparison of the computation times for the forest dataset. A: MLM (cosine, variance, training set size 2500, R size 250), B: One-Class-SVM, C: Isolation forest, D: Global RX, E: Local RX.

standard linear algebra. Based on the results, MLM seems to work in anomaly detection. It is effectively capturing global anomalies with low false alarm rate. As Figures 7–10 showed, the most influencing factor affecting computational cost and time is the size of subset R. This affects both training and testing. MLM reveals its efficiency when datasets are large.

Variance as anomaly detection value from cosine distance is working well with spectral data. This might be related to that spectral data is more angle-dependent than for example intensity related. Another interesting reason might be that variance of data points' cosine angles to other data points is actually an anomaly detection method called angle-based outlier degree.

The reader should have some caution because, for example, OC-SVM is using LIBSVM, which is native C++ implementation. If MLM were also implemented with C++, it would benefit from that in time comparison.

Because in testing phase $d(\mathbf{x}_n, R)$ is calculated in any case, we tested how $\mathrm{Var}(d(\mathbf{x}_n, R))$ or $||d(\mathbf{x}_n, R)||$ would work in the anomaly detection. Unfortunately, neither of these could produce sensible separation between anomalous and normal data points.

There are some interesting options for future work. Because of subset R is playing a crucial role in the computational efficiency of MLM, it should be paid attention to select these points more carefully. In this study, we used random sampling. We believe that with the intelligent selection we can significantly reduce the subset's size. There is still a need for the labelling of the training data. There might be strategies to avoid this, or we can change the labelling to other measures. It could be possible, for example, to build some regression upon other anomaly detection measures. MLM failed on detection of local anomalies. This could be avoided by building some piecewise training and detection algorithm.

5. CONCLUSION

In this study, we showed how minimal learning machine can be used in anomaly detection in general and in particular in the hyperspectral anomaly detection. Computationally efficient solutions are needed in real-time hyperspectral anomaly detection. Fortunately, MLM meets these criteria. Besides computational efficiency, MLM is easy to implement. It can be used in small single-board computers to utilise it in such applications as drone and cube satellite-based remote sensing.

ACKNOWLEDGEMENTS

This study is partly funded by Academy of Finland (Grant 327862). Eija Honkavaara, Niko Viljanen and Teemu Hakala from Finnish Geospatial Research Institute in National Land Survey of Finland (FGI), are acknowledged for the data capturing, photometric processing and georeferencing of forest dataset.

REFERENCES

Banerjee, A., Burlina, P., Diehl, C., 2006. A support vector method for anomaly detection in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 44(8), 2282–2291.

Chandola, V., Banerjee, A., Kumar, V., 2009. Anomaly Detection: A Survey. *ACM Comput. Surv.*, 41(3). https://doi.org/10.1145/1541880.1541882.

de Souza Junior, A. H., Corona, F., Barreto, G. A., Miche, Y., Lendasse, A., 2015. Minimal learning machine: a novel supervised distance-based approach for regression and classification. *Neurocomputing*, 164, 34–44.

Liu, F. T., Ting, K. M., Zhou, Z.-H., 2008. Isolation forest. *2008 Eighth IEEE International Conference on Data Mining*, IEEE, 413–422.

Manevitz, L. M., Yousef, M., 2001. One-class SVMs for document classification. *Journal of machine Learning research*, 2(Dec), 139–154.

Nevalainen, O., Honkavaara, E., Tuominen, S., Viljanen, N., Hakala, T., Yu, X., Hyyppä, J., Saari, H., Pölönen, I., Imai, N. N. et al., 2017. Individual tree detection and classification with UAV-based photogrammetric point clouds and hyperspectral imaging. *Remote Sensing*, 9(3), 185.

Nezami, S., Khoramshahi, E., Nevalainen, O., Pölönen, I., Honkavaara, E., 2020. Tree Species Classification of Drone Hyperspectral and RGB Imagery with Deep Learning Convolutional Neural Networks. *Remote Sensing*, 12(7), 1070.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. et al., 2011. Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, 12, 2825–2830.

Pölönen, I., Annala, L., Rahkonen, S., Nevalainen, O., Honkavaara, E., Tuominen, S., Viljanen, N., Hakala, T., 2018. Tree species identification using 3d spectral data and 3d convolutional neural network. *2018 9th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, IEEE, 1–5.

Reed, I. S., Yu, X., 1990. Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(10), 1760–1770.

Saari, H., Pölönen, I., Salo, H., Honkavaara, E., Hakala, T., Holmlund, C., Mäkynen, J., Mannila, R., Antila, T., Akujärvi, A., 2013. Miniaturized hyperspectral imager calibration and uav flight campaigns. *Sensors, Systems, and Next-Generation Satellites XVII*, 8889, International Society for Optics and Photonics, 88891O.

Zhao, C., Li, X., Zhu, H., 2017. Hyperspectral anomaly detection based on stacked denoising autoencoders. *Journal of Applied Remote Sensing*, 11(4), 042605.