



Support Vector Machines for Vegetation Classification – A Revision

ANDREAS CHRISTIAN BRAUN, UWE WEIDNER & STEFAN HINZ, Karlsruhe

Keywords: Support vector machines, vegetation classification, kernel learning, pattern recognition

Summary: Within the last decade, a new classification method from the field of machine learning, has found increased application in remote sensing. So called support vector machines (SVM) are being used for the classification of virtually all kinds of data used in remote sensing. They achieve special attention in the field of hyperspectral datasets. For these, SVMs are especially well suited, as they cope very well with their high number of dimensions. This contribution gives a thorough introduction into the topic, outlines fields of applications – thereby focussing on classifying vegetation – and presents a comparison of results with other methods such as maximum likelihood and spectral angle mapper.

Zusammenfassung: Über den Einsatz von Support-Vektor-Maschinen (SVM) für Vegetationsklassifizierung. Im letzten Jahrzehnt hat ein neues Klassifizierungsverfahren aus dem Bereich des maschinellen Lernens vermehrt Anwendung im Bereich der Fernerkundung gefunden. Die so genannten Support-Vektor-Maschinen (SVM) werden für die Klassifizierung praktisch aller Datentypen der Fernerkundung eingesetzt. Sie erhalten eine besondere Aufmerksamkeit im Bereich hyperspektraler Datensätze, für die SVMs besonders geeignet sind, da sie sehr gut mit der hohen Anzahl an Dimensionen zurechtkommen. Dieser Beitrag gibt eine ausführliche Einführung in das Thema, stellt Anwendungsgebiete – vor allem aus dem Bereich der Vegetationsklassifizierung – dar und präsentiert einen Vergleich anhand von Klassifizierungsergebnissen mit anderen Verfahren (Maximum-Likelihood und Spectral Angle Mapper).

1 Introduction

In recent years, a new technique of pattern recognition has found broad acceptance in the remote sensing community. Support vector machines (SVM), have been developed by VAPNIK et al. (1992) as a method of supervised classification. Since then, the potential of different types of SVM classifiers has been improved and adapted to a large number of tasks. Classification procedures based on SVM have been applied to multispectral (MITRA et al. 2004), hyperspectral (GUALTIERI & CROMP 1998, CAMPS-VALLS et al. 2004) data, synthetic aperture radar (SAR) data (TAN et al. 2007), and light detection and ranging (LiDAR) data

(LODHA et al. 2006). In a large number of publications, SVMs outperformed other classifiers (like Maximum-Likelihood, Minimum distance, Spectral angle mapper, but also neural networks, FOODY & MATHUR 2004) in terms of accuracy and robustness. The main advantage of SVM is given by the fact that they abstain from a statistical description of the data, but find optimal separation (hyper-)planes learnt from the spatial distribution of training data in the feature space.

The paper is organized as follows. At first, we will give a non-mathematical introduction to SVM, then we will give some examples of applications in the field of vegetation and forestry. After that, we will present results from

our own work. Finally, we will give a short summary and outlook.

2 Brief Revision of SVM

The basic drawback of common classifiers that led to the development of SVM was to find a classifier that generalizes well and is less susceptible to non-Gaussian distributed and deviant patterns than statistically motivated methods (VAPNIK et al. 1992, CORTES & VAPNIK 1995). Consider the situation given in Figs. 1 and 2. The set of points for the class marked by crosses, contains outliers. Thus, a classifier based on the class centres (like Minimum Distance) is strongly affected by the location of these outliers, resulting in a separation plane that cuts off some of the points which are close to the other class. Effectively, points that lie behind a front of other points from the same class, are the ones that should least affect the position of the separating plane. It is evident, that the only points that should affect the position of the plane, are the ones that lie opposed to the closest points from the other class, (marked by small squares in Fig. 2). These are called support vectors. If the two classes can be separated linearly, it is possible to find an infinite set of lines to separate the classes (imagine rotating the dashed line in Fig. 2 by a few degrees). As we cannot be sure, that the training data describe their class perfectly, it should be aimed to find a line which leaves a broad area (called margin in SVM) between classes to leave space for the test data. According to the statistical learning theory (for an overview cf., e. g., VAPNIK 1999), a classifier maximizing the free space between point

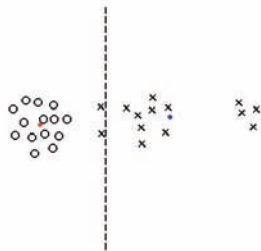


Fig. 1: K-means Separation.

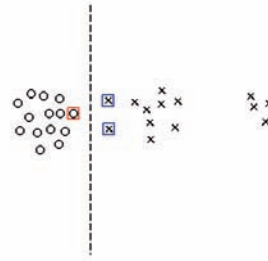


Fig. 2: SVM Separation.

clouds generalizes well, i. e., runs a low risk of classifying unknown patterns falsely.

These considerations led to the development of SVM (VAPNIK et al. 1992). Given an p -dimensional feature space, we choose a couple of m training samples x_i with p features $x = (x_1, x_2, \dots, x_p)$ and a class label $y_i \in [-1, +1]$ to form a dataset $T = \{x_1, y_1, x_2, y_2, \dots, x_m, y_m\}$. Considering Fig. 2, it becomes clear that there exists a separation plane, given in the normal form $w \cdot x + b = 0$, which correctly assigns all of the training data in T into the classes defined by their labels y_i , leaving a broad margin for data with unknown label. Here, w is the normal vector on the plane and the bias b is the distance from the closest point on the hyperplane to the origin. The task is now to solve an optimisation problem, thus, finding an optimal separation hyperplane (OSH). Mathematical foundations can be found in (BURGES 1998, IVANCIUC 2007). A very comprehensive, yet more intuitive description is given in (BENNETT & CAMPBELL 2000). Numerical solutions are described in (BOTTOU & LIN 2007). Within the optimisation problem, patterns from T appear in dot products – an important feature to keep in mind. After w and b have been found, deciding the class membership of new data is done by simply checking, on which side of the hyperplane they are situated.

2.1 Relieving the Constraints by Soft Margin SVM

However, in many cases it may not be possible, or wanted, to find a plane which separates all the training samples into the right classes.

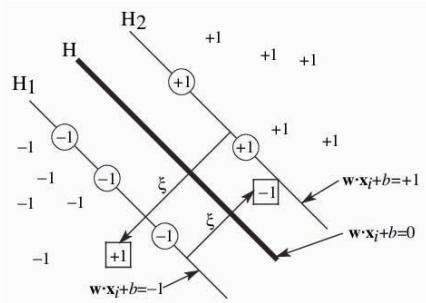


Fig. 3: Soft-margin SVM by using slack variables ξ , w represents the normal vector of the plane, b is the distance of the closest point to the origin.

As shown in Fig. 3, for each class there exists one pattern (marked by squares) to be on the opposite site of the separating plane. These might be known exceptions, like pixels from falsely assigned training areas. In this case it could be allowed for these pixels to be on the wrong side of the plane. This is realized by introducing slack variables $\xi_i \geq 0$ to each data point, which equal zeros if the points are located correctly and has a positive value if the point is on the wrong side. These slack variables are summed up, penalized by a weighting factor and added to the optimisation problem. Various concepts of this so called *Soft Margin* approach have been proposed in literature, the most common ones being *C* (Cost) SVM (CORTES & VAPNIK 1995). However, in recent publications, a promising new approach named ν -SVM (SCHÖLKOPF et al. 2000) is used. It can be applied to achieve separation of one class regardless of the other classes (One-class SVM).

2.2 Finding a Solution in High-dimensional Spaces by Kernels

Fig. 4 gives an example, where using a statistical description can lead to linearly infeasible separation problems. Although it is clear that the two classes can perfectly be separated, using a criterion like the class centres is not adequate to solve the problem. As in this case linear SVM, even using slack variables, also fails to separate the data linearly, this situation leads us to the next stage.

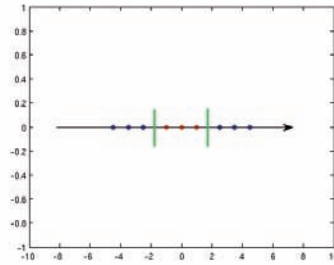


Fig. 4: Original feature space.

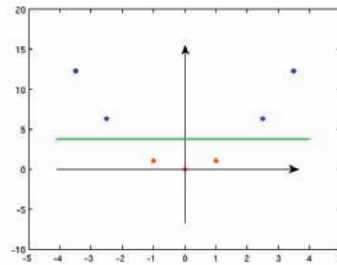


Fig. 5: Higher dimensional space.

For exactly this kind of cases, the SVM approach has been enhanced. Separating the two classes is merely a question of increasing the dimensionality of the feature space. This is done by introducing a non-linear feature space mapping $\Phi(x)$ (like the square distance to the origin in this case, see Fig. 5) that makes the data linearly separable in the higher-dimensional space. $\Phi(x)$ is not required to be calculated explicitly, though. Instead the so called „kernel-trick“ (IVANCIUC 2007) is applied, which reformulates the mapping and evaluating of two data points x_i and x_j by $K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$. Kernels operate in the original feature space but act like dot products in the high-dimensional space (BURGES 1998). Thus, they compute the solution implicitly, making the problem resolvable in a much easier way. There exists a large variety of kernels, yet the most frequently used ones – apart from the linear kernel, which equals to not performing any transformation – are the following:

$$K(x_i, x_j) = \exp\left(\frac{x_i \cdot x_j}{\sigma^2}\right)$$

Radial Basis Function (RBF) Kernel (1)

$$K(x_i, x_j) = (x_i \cdot x_j + 1)^d$$

Polynomial Kernel of d^{th} degree (2)

2.3 Methods for More than Two Classes by Multi-class SVM Strategies

Up to now we have only discussed two class problems. Within the field of remote sensing, it is merely the case, to have only two classes of interest. What is needed is a method to adapt SVMs to the classification of problems with more than two classes. Some research has been made using specialized algorithms to treat more than two classes at once (WESTON & WATKINS 1999, ASHARAF et al. 2007, SOMAN et al. 2007). However, until now these methods are still on an experimental level. The approach widely used is to subdivide the n class problem into several 2 class problems. A number of specific strategies have been developed, and the three most important ones are presented here. The first adaptation of SVM to multiple classes was the One-against-all approach. For each of the n given training classes, a OSH is optimised, labelling the recent class as +1 and all the pixels from the other classes as -1. During the classification phase, each pixel is passed on to each of the n SVM classifiers. It is assigned as member of a class, if labelled as +1 by the respective SVM classifier and left without class membership if labelled as -1. As this approach has a couple of inconveniences (e. g., the existence of a not meaningful rejection class), One-against-one has been invented. It refuses to consider all n classes at the same time, but finds for all the $n(n-1)/2$ combinations of classes a suitable hyperplane. This means, that a certain hyperplane is only meaningful for two classes, ignoring all the other classes. In the classification phase, each pixel is checked by all the $n(n-1)/2$ classifiers, recording the label attributed by each classifier. After finishing this process, the max wins strategy assigns the pixels to the classes, they have received most votes from. There has been a number of publications using hierarchy-based approaches. The most common one is the DAG (directed acyclic graph) by PLATT et al. (2002). It can be

considered as a special case of One-against-one, due to the identical training procedure. In contrast to One-against-one, the data set is split up during the classification phase by following a graph. Thus, the data set which is passed on to a certain SVM model gets ever smaller on each layer of the graph, shortening classification times significantly.

2.4 One-class SVM

A major drawback of these approaches is the lack of methods to train and classify images, using only a subset of the classes present in the image. If, for example, out of n classes in a rural area, only two (e. g., forests and meadows) are of interest, one none the less has to train SVM models, considering all the classes. This drawback can be solved using One-class SVM. This strategy is effectively not a multi-class strategy, but a different type of error penalization (ν -SVM) which can be exploited to overcome the drawbacks of the *Cost-SVM* based multi-class strategies given in Section 2.3. Instead of finding a hyperplane, that separates a class from the other classes, a hypersphere is found, which optimally hulls in the points from one class. This enables the user, to find a separating criterion for a given class *without* explicitly assigning training areas for other classes.

3 Performance of SVM for Forest and Vegetation Classification

In this section, we will show the applicability of SVM in the field of vegetation classification and forestry. At first, an overview on the work of other authors from the field of forestry will be given. In the next step, we will present some results from our own work that was achieved using the classical multi-class approaches, presented in Section 2.3. We will conclude by comparing a One-against-all classification, using the traditional *Cost-SVM* approach, with the newer One-class SVM. The reason for this is, that One-class SVM is a promising approach, which has not been used widely in remote sensing so far.

3.1 Related Work

In this section, we present some related work on the application of SVMs in the field of forestry. WIJAYA & GLOAGUEN (2007) classify two different Landsat ETM7+ scenes of tropical rainforests (South Ecuador, Central Indonesia) into five classes. Using SVM, they achieve 73 % of overall accuracy for Ecuador and 81 % for Indonesia. In their study, ZAMMIT et al. (2006) present classifications of burnt forest areas using after-fire scene in France from SPOT5. In their comparison, SVM outperforms K-Means, and K-Nearest-Neighbours regarding overall accuracy (96.5 % for SVM) and false positives (3.1 % for SVM). LAFARGE et al. (2004) apply a specialized kernel to detect forest fires with SPOT5. Applying SVM, the authors are able to detect fire through smoke formation using only the panchromatic channel. Through a combination of Landsat classification with climate data and other ecological data, GUO et al. (2005) develop a prediction map for areas, threatened by sudden oak death (*Phytophthora ramorum*) in California. The authors report that, due to the high true-positive rates yielded (0.91), SVM can be recommended for ecological niche classification. YANG et al. (2006) use SVMs to predict gross primary production of forested and non-forested areas at a continental-scale using data from the MODIS sensor. The results of SVM were competitive to the standard MODIS GPP/NPP algorithm.

3.2 Performance Analysis for Different Data, Kernels and Strategies

The capability of SVM shall be shown by comparing the classification accuracy of SVM to those achieved by other methods. In this section, we will present the classification results for different data types. For some results, we will only give the Overall accuracy (OAA) and cappa statistics in Tab. 1. Three results, however, will be discussed in detail. The first scene was taken by the QuickBird satellite in 2005. Secondly, a dataset, taken by the HyMap sensor in 2003 was worked on. The scenes show the urbanized area of Karlsruhe, Ger-

many. The last scene was a subset of the RapidEye Demonstration scene, showing the urbanized area of Griffith, New South Wales, Australia. For each scene, the same training areas were used to perform both an SVM classification, and another classification approach (Maximum likelihood for QuickBird and RapidEye, Spectral angle mapper for HyMap). Maximum likelihood classification (MLC) was performed using a commercial software package, Spectral angle mapper (SAM) was carried out, using Hypertools (PACLİK et al. 2005) for Matlab. The SVM classification was performed, using an adapted version of LibSVM (CHANG & LIN 2008). Afterwards, control areas were defined by field survey, using our knowledge of the scenes and aerial imagery available at the institute. For each land cover class, various control areas from different locations in the scene where assigned. We used at least two times more pixels for control areas, than for training areas and made sure, that no class had significantly more control area pixels than others. These control areas where used to calculate confidence matrices for both the SVM classification as well as for the comparison (MLC or SAM).

Comparison with Maximum likelihood – QuickBird

From the entire QuickBird scene, a 1.000 per 1.000 pixel subset, showing the forested area north west of Karlsruhe castle, was taken. 13 different land cover classes, covering a wide range of vegetation, as well as urban classes, were chosen. Concerning the forested areas, two different kinds of mixed beech groves (*Fagetalia*), two different kinds of meadows (one in good conditions, one slightly degraded by trespassing) and an alley of lime-tree (*Tilia europaea*) on gravel were taken into account. Training was performed for both MLC and Cost-SVM (with RBF kernel, applying the One-against-one strategy). Hence, for each of the $n(n-1)/2$ training steps, a different SVM model was obtained. Considering overall accuracy and cappa coefficient, the SVM outperformed MLC (see Tab. 1). Comparing results (given in Figs. 11 to 13) visually, it can be noted, that SVM delivered a more consistent image. Many surfaces (like, e. g., red roofs, or

the area around the castle) appear more coherent in SVM than in MLC. The same accounts for the lake. The MLC result seems too much dominated by the cyan class for alleys. The reason for this might be, that alleys are composed of trees and the sand beyond. This results in a class with two class centres, which deviates strongly from the normal distribution recommended for MLC classification. Furthermore, SVM seems to be better suited to distinguish between dark-grey roofs and light-grey streets.

Comparison with Maximum Likelihood – RapidEye

As for QuickBird, a *Cost-SVM* approach with RBF Kernel and One-against-one was compared to MLC. The subset size was 1.000 per 1.000 pixels and 13 landuse classes have been assigned. (Figs. 14 to 16) show the visual results. As can be seen, MLC assigned much more pixels to the dark-green fields class, which seems only partly justified. There are obvious differences in the urbanized areas. The SVM assigned less pixels in the city to vegetation classes. Industry and other urban classes are much more dominant, reinforcing the impression of higher coherence, that the SVM result for QuickBird gave. Just as for QuickBird, Overall Accuracy and kappa statistics of SVM exceeded those of MLC (see Tab. 1).

Comparison with spectral angle mapper – HyMap

To compare SVM with a traditional method for classification of hyperspectral data, a 281 per 169 pixel subset from a HyMap scene was chosen and processed with SAM and SVM. For the latter, a *Cost-SVM* approach with the One-against-one strategy was chosen. We

used a linear kernel, as due to the high number of dimensions (126) in the input space, we hoped to be able to find a linear OSH without transforming the data to higher dimensional spaces. As the comparison of overall accuracy and kappa (Tab. 1) reveals, this assumption was justified. As well as for the QuickBird classification, the visual results (Figs. 17 to 19) seem more coherent for SVM, than for SAM. Within the forests, SVM could distinguish better between the different tree classes. Shrublands marked in yellow have been identified more clearly and also buildings are not so much mixed up with pixels from other classes.

Comparison One-Against-all versus One-Class SVM

As described in Section 2.4, a major advantage of One-class SVM is, that training and classification can be performed for each class independently of the other classes. Despite this obvious advantage, One-class SVM has found few applications in remote sensing so far, none of the work presented in Section 3.1 has used it. To illustrate the difference between One-class SVM, we performed a short comparison. For a very small (338 per 291 pixels) subset of the QuickBird scene, different classifications were performed. The goal was to achieve a good separation for three *vegetation classes only*: dark trees like beech (*Fagus sylvatica*) and hornbeam (*Carpinus betulus*), light trees like lime-tree (*Tilia europaea*) and birch (*Betula spec.*) and meadows, see Fig. 6). At first, a One-against-all classification was performed, assigning trainings areas for the three vegetation classes only. Fig. 7 shows that the outcome was quite poor. This is exactly what we had expected, as the One-against-all strategy is unable to perform well if not trained for all classes. Next, we assigned training ar-

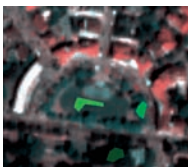


Fig. 6: Training areas.



Fig. 7: O-a-a 3 classes.



Fig. 8: O-a-a all classes.

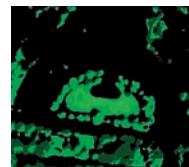


Fig. 9: O-a-a set to black.

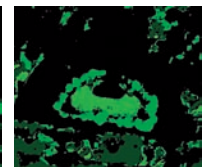


Fig. 10: One-class-SVM.

	Dark Trees (<i>Fagus</i> , <i>Carpinus</i> etc.)	Light Trees (<i>Tilia</i> , <i>Betula</i> etc.)	Meadows
One-class SVM	28.3 %	51.8 %	65.1 %
One-against-all	33.5 %	75.8 %	76.3 %

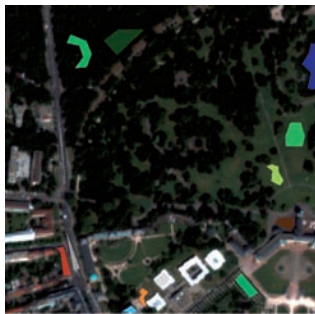


Fig. 11: QuickBird training areas.



Fig. 12: SVM result.



Fig. 13: MLC result.



Fig. 14: RapidEye training areas.



Fig. 15: SVM result.



Fig. 16: MLC result.



Fig. 17: HyMap training areas.

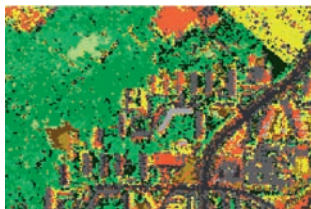


Fig. 18: SVM result.



Fig. 19: SAM result.

reas to *all the classes* present in the scene and classified with One-against-all again. Fig. 8 shows the outcome for all classes, in Fig. 9 all classes apart from the vegetation classes have been set to black to ensure visual comparability with the next result. At last, we trained a One-class SVM for each of the three vegetation classes, ignoring of course all the other

classes. In contrast to Fig. 7, Fig. 10 shows that this strategy *can indeed* be followed, using One-class SVM.

However, comparing results quantitatively, One-class SVM failed to outperform One-against-all. By assigning user-defined control areas, the rates of true positives were calculated. One-class SVM achieved poorer accu-

Tab. 1: Comparison of different classifications (results in bold letters are described in this report).

Scene	SVM		MLC		SAM	
	OAA	Cappa	OAA	Cappa	OAA	Cappa
QuickBird	87.5 %	0.86	80.8 %	0.78	---	---
RapidEye	79.4 %	0.77	62.7 %	0.59	---	---
Landsat TM	80.1 %	0.8	65.4 %	0.6	---	---
HyMap subset 1	72.0 %	0.7	---	---	48.5 %	0.4
HyMap subset 2	82.6 %	0.8	---	---	48.6 %	0.4

racy for all of the three vegetation classes. One-class SVM found in total fewer pixels for each class, this might have led to the worse accuracy rates. A possible reason for this might be overfitting. As the training areas for each class were very small, classes could probably not been described well enough. Another reason might be, that the differences between dark and light trees have not been well distinguished by One-class SVM. The rows of dark trees in the south of the scene are too much mixed up with pixels classified as light trees in the result of One-class SVM.

4 Summary and Outlook

The results presented here confirm the statements of other authors (FOODY & MATHUR 2004, MELGANI & BRUZZONE 2004, LODHA et al. 2006, TAN et al. 2007) that SVMs are promising and capable classifiers for remotely sensed data of different kinds. The method delivers reliably well results for data of different dimensionality and origin (LiDAR, SAR, multi- and hyperspectral). It outperformed MLC and SAM in a direct comparison. Especially in the case of hyperspectral data, a linear kernel is worth trying. Due to the high dimensionality of the hyperspectral input space, there might be no need to chose a non-linear functions to achieve separation. According to (VAPNIK et al. 1992) the more complex the kernel function is, the SVM classifier produces higher generalization errors.

A lot of work is currently being done in developing specialized kernels (CAMPS-VALLS et al. 2005), better training algorithms (SU 2009, KEERTHI & LIN 2003) and combination of SVM with feature extraction (CAO et al. 2003). How-

ever, we consider, that the usage of One-class SVM is a promising field of work, as it allows the user to classify only the classes of interest to him, without having to keep in mind each of the n occurring classes in the entire scene.

References

- ASHARAF, S., NARASIMHA MURTY, M. & SHEVADE, S.K., 2007: Multiclass Core Vector Machines. – Computer Science and Automation, Indian Institute of Science, Bangalore, India – 560012.
- BENNETT, K. & CAMPBELL, C., 2000: Support Vector Machines: Hype or Hallelujah? – Math Sciences Department Rensselart Polytechnic Institute, Troy, NY 12180.
- BOSER, B., GUYON, I. & VAPNIK, V., 1992: A Training Algorithm for Optimal Margin Classifiers. – AT&T Bell Laboratories, Crawford Corner Road, Holmdel, NJ 07733.
- BOTTOU, L. & LIN, C.J., 2007: Support Vector Machine Solvers. – Large Scale Kernel Machines, MIT Press, Cambridge, MA, USA, 1–28.
- BURGES, C., 1998: A Tutorial on Support Vector Machines for Pattern Recognition. – Kluwer, Boston, USA.
- CAMPS-VALLS, G., GÓMEZ-CHOVA, L., CALPE-MARAVILLA, J., MARTÍN-GUERRERO, J., SORIA-OLIVAS, E., ALONSO-CHORDÁ, L. & MORENO, J., 2004: Robust Support Vector Method for Hyperspectral Data Classification and Knowledge Discovery. – IEEE Transactions on Geoscience and Remote Sensing.
- CAMPS-VALLS, G., GOMEZ-CHOVA, L., MUÑOZ-MARÍ, J., VILA-FRANCÉS, J. & CALPE-MARAVILLA, J., 2005: Composite Kernels for Hyperspectral Image Classification. – IEEE Geoscience and Remote Sensing Letters.
- CAO, L.J., CHUA, K.S., CHONG, W.K., LEE, H.P. & GU, Q.M., 2003: A comparison of PCA, KPCA and ICA for dimensionality reduction in support

- vector machine. – *Neurocomputing* **55**: 321–336.
- CHANG, C.C. & LIN, C.J., 2008: LIBSVM: A Library for Support Vector Machines. – Department of Computer Science, National Taiwan University, Taipei 106, Taiwan.
- CORTES, C. & VAPNIK, V., 1995: Support-Vector Networks. – AT&T Labs Research, USA.
- FOODY, G. & MATHUR, A., 2004: A Relative Evaluation of Multiclass Image Classification by Support Vector Machines. – *IEEE Transactions on Geoscience and Remote Sensing* **42** (6).
- GUALTIERI, J. & CROMP, R., 1998: Support Vector Machines for Hyperspectral Remote Sensing Classification. – *SPIE*, **3584**, 27th AIPR Workshop, Advances in Computer Assisted Recognition.
- GUO, Q.H., KELLY, M. & GRAHAM, C., 2005: Support vector machines for predicting distribution of Sudden Oak Death in California. – *Ecological Modelling* **182**: 75–90.
- IVANCIUC, O., 2007: Applications of Support Vector Machines in Chemistry. – *Reviews in Computational Chemistry* **23**, Wiley-VCH.
- KEERTHI, S. & LIN, C.J., 2003: Asymptotic Behaviours of Support Vector Machines with Gaussian Kernels. – *Neural Computation* **15**: 1667–1689.
- LAFARGE, F., DESCOMBES, X. & ZERUBIA, J., 2004: Textural kernel for SVM classification in Remote Sensing: Application to forest fire detection and urban area extraction. – Ariana Project, INRIA, 2004 – route de Lucioles – BP93, 06902 Sophia Antipolis, France.
- LODHA, S., KREPS, E., HELMBOLD, D. & FITZPATRICK, D., 2006: Aerial LiDAR Data Classification using Support Vector Machines (SVM). – Department of Computer Science, University of California, Santa Cruz, CA 95064, USA.
- MELGANI, F. & BRUZZONE, L., 2004: Classification of Hyperspectral Remote Sensing Images With Support Vector Machines. – *IEEE Transactions on Geoscience and Remote Sensing* **42** (8).
- MITRA, P., SHANKAR, B. & PAL, S., 2004: Segmentation of multispectral remote sensing images using active support vector machines. – *Pattern Recognition Letters* **25**: 1067–1074.
- PACLIK, P., VERZAKOV, S. & DUIN, R., 2005: Hyper-tools: the toolbox for spectral image analysis.
- PLATT, J., CRISTIANINI, N. & SHAWE-TAYLOR, J., 2000: Large Margins DAGs for Multiclass Classification. – MIT Press, 547–553.
- SCHÖLKOPF, B., SMOLA, A., WILLIAMSON, R. & BARTLETT, P., 2000: New Support Vector Algorithms. – *Neural Computation* **12**: 1207–1245.
- SOMAN, K.P., LOGANATHAN, R., VIJAYA, M.S., AJAY, V. & SHIVSUBRAMANI, K., 2007: Fast Single-Shot Multiclass Proximal Support Vector Machines and Perceptrons. – International Conference on Computing: Theory and Applications (IC-CTA'07), 2007.
- SU, L., 2009: Optimizing support vector machine learning for semi-arid vegetation mapping by clustering analysis. – *ISPRS Journal of Photogrammetry and Remote Sensing* **64**: 407–413.
- TAN, C., KOAY, K., LIM, K., CHUAH, E. & CHUAH, H., 2007: Classification of Multi-Temporal Images for Rice Crops using Combined Entropy Decomposition and Support Vector Machine Technique. – *Progress In Electromagnetics Research*, PIER **71**, 19–39.
- VAPNIK, V., BOSER, B. & GUYON, I., 1992: A Training Algorithm for Optimal Margin Classifiers. – AT&T Bell Laboratories, 50 Fremont Street, San Francisco, CA 94105, USA.
- VAPNIK, V., 1999: An Overview of Statistical Learning Theory. – *IEEE Transactions on Neural Networks*, **10** (5).
- WESTON, J. & WATKINS, C., 1999: Support Vector Machines for Multi-Class Pattern Recognition. – ESANN'1999 proceedings – European Symposium on Artificial Neural Networks, 219–224.
- WIJAYA, A. & GLOAGUEN, R., 2007: Comparison of Multisource Data Support Vector Machine Classification for Mapping of Forest Cover. – Remote Sensing Department, TU-Bergakademie Freiberg.
- YANG, F.H., ICHII, K., WHITE, M., HASHIMOTO, H., MICHAELIS, A., VOTAVA, P., ZHU A.X., HEUTE, A., RUNNING, S. & NEMANI, R., 2006: Developing a continental-scale measure of gross primary production by combining MODIS and AmeriFlux data through Support Vector Machine approach. – *Remote Sensing of Environment* **110**: 109–122.
- ZAMMIT, O., DESCOMBES, X. & ZERUBIA, J., 2006: Burnt Area Mapping using Support Vector Machines. – V International Conference on Forest Fire Research, 2006.

Address of the Authors:

Dipl.-Geoökol. ANDREAS CHRISTIAN BRAUN, Dr.-Ing. UWE WEIDNER, Prof. Dr.-Ing. habil. STEFAN HINZ, Karlsruhe Institute of Technology (KIT) Karlsruhe, Institut für Photogrammetrie und Fernerkundung, 76128 Karlsruhe, Tel.: +49-721-608-6994, -3945, -2314, Fax: -8450, e-mail: andreas.ch.braun@kit.edu, uwe.weidner@kit.edu, stefan.hinz@kit.edu.

Manuskript eingereicht: Februar 2010
Angenommen: April 2010