PREPRINT; FINAL PAPER PUBLISHED @ ISPRS JOURNAL OF PHOTOGRAMMETRY AND REMOTE SENSING,
VOL. 151, PP. 42-58, MAY 2019

VIRTUAL SUPPORT VECTOR MACHINES WITH SELF-LEARNING STRATEGY FOR CLASSIFICATION OF MULTISPECTRAL REMOTE SENSING IMAGERY

Christian Geiß*, Patrick Aravena Pelizari, Lukas Blickensdörfer, and Hannes Taubenböck

German Remote Sensing Data Center (DFD), German Aerospace Center (DLR), 82234 Weßling-Oberpfaffenhofen, Germany; christian.geiss@dlr.de, patrick.aravenapelizari@dlr.de, lukas.blickensdoerfer@dlr.de, hannes.taubenboeck@dlr.de

Abstract: We follow the idea of learning invariant decision functions for remote sensing image classification with Support Vector Machines (SVM). To do so we generate artificially transformed samples (i.e., virtual samples) from available prior knowledge. Labeled samples closest to the separating hyperplane with maximum margin (i.e., the Support Vectors) are identified by learning an initial SVM model. The Support Vectors are used for generating virtual samples by perturbing the features to which the model should be invariant. Subsequently, the model is relearned using the Support Vectors and the virtual samples to eventually alter the hyperplane with maximum margin and enhance generalization capabilities of decisions functions. In contrast to existing approaches, we establish a self-learning procedure to ultimately prune non-informative virtual samples from a possibly arbitrary invariance generation process to allow for robust and sparse model solutions. The self-learning strategy jointly considers a similarity and margin sampling constraint. In addition, we innovatively explore the invariance generation process in the context of an object-based image analysis framework. Image elements (i.e., pixels) are aggregated to image objects (as represented by segments/superpixels) with a segmentation algorithm. From an initial singular segmentation level, invariances are encoded by varying hyperparameters of the segmentation algorithm in terms of scale and shape. Experimental results are obtained from two very high spatial resolution multispectral data sets acquired over the city of Cologne, Germany, and the Hagadera Refugee Camp, Kenya. Comparative model accuracy evaluations underline the favorable performance properties of the proposed methods especially in settings with very few labeled samples.

Keywords: Classification, Support Vector Machines, Self-Learning, Active Learning Heuristics, Very High Spatial Resolution Imagery.

1 Introduction

Developing methods for information extraction from remote sensing imagery has been one of the major tasks of the scientific remote sensing community in the past decades. Thereby, different strategies are followed to derive thematic classes from the image data. Due to their comparatively robust and accurate information extraction properties, supervised methods belong to the most popular group of classification approaches. The idea of such approaches is to infer a decision rule (e.g., a function) from limited but properly encoded prior knowledge (i.e., labeled training samples) to allow for an accurate association of class labels for unseen (i.e., unlabeled) samples. However, it is generally very challenging to choose the best method (most likely in terms of classification accuracy) from dozens of different existing approaches for an individual classification problem (Fernández-Delgado et al., 2014). This can be related to the No Free Lunch Theorem, which states, briefly speaking, that if a strategy is more favorable in a certain subdomain then it must be less favorable in another subdomain (Wolpert, 1996; Duda et al., 2001). Nevertheless, for the subdomain of supervised classification problems jointly dealing with i) a small amount of labeled training samples, ii) a high number of features, and iii) complex, nonlinear class distributions, a number of non-parametric machine learning algorithms can be considered as a viable option in general. Their algorithm properties make them in particular relevant for this situation.

For instance, Random Forests (Breiman, 2001) grow multiple decision trees on random subsets of the training samples. The high variance among individual trees, letting each tree vote for the class assignment and determining the respective class according to the majority of the votes, allows the accurate and robust classification of unseen samples with little need for tuning, even when many noisy variables are existent (Stumpf and Kerle, 2011). Alternatively, Support Vector Machines (SVM) is a very popular approach in this application context since they also offer the capability of effectively handling complex remote sensing classification problems (Melgani and Bruzzone, 2004; Camps-Valls and Bruzzone, 2009). They are based on the structural risk minimization principle, which suggest a tradeoff between the accuracy of an approximation and the complexity of the affiliated approximation function. SVM determine a suitable set of parameters to fit a decision surface, the so-called hyperplane, between different classes of labeled samples. To deal with nonlinear problems, the labeled samples are mapped through a nonlinear transformation $\phi(\cdot)$ from the input space into a space of higher dimensionality. In that space, the optimal separating hyperplane maximizes the margin between the patterns of the different classes and the hyperplane. The maximized margin can be described by two additional, marginal hyperplanes that border the samples closest to the separating surface, the so-called support vectors (SVs) (Burges, 1998; Leinenkugel et al., 2011; Geiß et al., 2016a). *Only those samples are needed to define the model*, what allows for building robust models with a high generalization capability based on a comparatively small number of labeled training samples. Simultaneously, this algorithm property opens the opportunity to encode *further prior knowledge* in the classifier in a very *efficient way*: When learning a classification model from labeled samples it is preferable that the solution is robust with respect to changes in the representation of the objects in the data. Those might occur when objects in the data are transformed, e.g., due to perturbations and variations in size, alignment or noise level of the affiliated signal (Camps-Valls et al., 2014).

The incooperation of such properties into the classification model (i.e., here decision functions) renders an algorithm "invariant" (Izquierdo-Verdiguier et al., 2013). In this sense, a suitable approach should account for a tailored regularization scheme to allow for a high generalization capability of learned decision functions also for unseen data with a considerable share of transformed objects.

From different options to *encode invariances in SVM* as e.g., discussed in (Decoste and Schölkopf, 2002), we consider in this paper the idea to generate *artificially transformed samples* (i.e., virtual samples) from the training samples to augment the set of labeled samples which the model is learned from. Recently, data augmentation strategies are frequently implemented in the context of deep learning procedures (Wang and Perez, 2017). Such techniques normally need a large pool of training data to generalize well. Consequently, ideas were followed to crop, rotate, or flip image data used for training to enlarge the training data and enhance the model accuracy (Yu *et al.*, 2017; Nogueira *et al.*, 2017; Audebert *et al.*, 2018). However, as indicated, SVM allow for a very efficient way to add artificially transformed samples into the model (in contrast to e.g., neural networks), since frequently only a few samples are needed from the whole set of available labeled samples to define the model. As such, artificially transformed samples are generated only from *a subset* of the labeled training samples (i.e., those samples that become a Support Vector after a preliminary model run). This renders the approach in particular feasible from a computational point of view.

Interestingly, the idea to augment the training set by using virtual samples in order to render a SVM model invariant was introduced quite recently to the context of remote sensing image classification. In this manner, the authors of (Izquierdo-Verdiguier et al., 2013) encode invariances to rotations and object scales in the context of patch-based image classification (i.e., in relation to square-shaped image subsets representing objects of interest). There, invariances are pre-engineered by an expert and added to the model without further constrains.

In contrast to that, we establish a procedure that first identifies labeled samples closest to the separating hyperplane with maximum margin - the SVs – from learning an initial SVM model. Those SVs are the basis for generating artificially transformed samples, i.e., virtual samples, by perturbing the features to which the model should be invariant. In previous works, the model is relearned using SVs and virtual samples to eventually alter the hyperplane with maximum margin and enhance generalization capabilities of decisions functions. This approach is called Virtual Support Vector Machines (VSVM) (Decoste and Schölkopf, 2002; Izquierdo-Verdiguier et al., 2013). However, it is very challenging and critical to add valuable virtual samples to a model with a high degree of automatization (i.e., minimizing the amount of necessary prior knowledge). If virtual samples are not properly generated and selected, they can introduce divergence and thus reduce classification accuracy of a relearned model (Izquierdo-Verdiguier et al., 2013). Moreover, only few informative virtual samples should be considered to enable improvements in model generalization capability while keeping simultaneously computational complexity low. To address the aforementioned considerations, we follow a self-learning strategy to eventually prune virtual samples from an arbitrary

invariance generation process to subsequently establish robust and sparse model solutions. To this purpose, generated virtual samples are evaluated *first* with respect to the Euclidean distance in feature space to their affiliated SVs and are pruned from the set of training samples if they exceed an empirically determined class-specific distance. *Second*, residual virtual samples are also evaluated with respect to their position to the margin and are pruned from the set of training samples if they exceed a specific margin distance. The model is relearned using SVs and residual virtual samples to establish an invariant SVM model. Thereby, determination of hyperparameters of the self-learning constraints is rendered as a further minimization objective. The term Virtual Support Vector Machines with self-learning strategy (VSVM-SL) is used in the subsequent paper when referring to this technique.

The self-learning strategy can be interpreted as an active learning procedure (Tuia et al., 2009), where the iterative human-machine interaction is substituted in favor of a singular machine-machine interaction. Related ideas were exploited in the context of semi-supervised classification approaches. Such methods iteratively label unlabeled samples based on a preliminary trained learning machine to enhance generalization capabilities in subsequent model learning stages and enable faster convergence (i.e., obtain higher accuracies for the same amount of encoded prior knowledge compared to purely supervised approaches) (Bruzzone et al., 2006). Thereby, active queries (Tuia et al., 2011) aim to enable the selection of few relevant unlabeled samples. This is done to only consider highly informative samples, for instance uncertain samples close to the border of the hyperplane (Demir et al., 2011). In this way significant improvements in accuracy become possible without taking a large number of unlabeled samples into account, and thus increase computational complexity. In addition, an unconstrained selection of unlabeled samples may lead to reduced accuracies if they add noise and blur distinctive class patterns in feature space (Dópido et al., 2013; Li and Zhou, 2015; Lu et al., 2016).

Besides the actual model learning procedure, we innovatively explore the invariance generation process in the context of an object-based image analysis (OBIA) framework (Blaschke, 2010; Geiß and Taubenböck, 2015; Geiß et al., 2016b). There, the aim is to aggregate image elements (i.e., pixels) to meaningful image objects (as represented by segments/superpixels) with a segmentation algorithm. From an initial singular segmentation level, invariances are encoded by varying hyperparameters of the segmentation algorithm in terms of scale (i.e., by establishing a hierarchical multi-level segmentation; (Bruzzone and Carlin 2006; Taubenböck et al., 2010) and shape (i.e., by considering multiple segmentation levels as obtained with varying shape-related hyperparameters).

The proposed method can be considered in particular relevant in situations where the ground sampling distance is much smaller than the objects of interest and the spectral resolution is limited as it is the case for multispectral imagery. This is related to the fact that geospatial variations of representations of objects are considered here for learning a model. Such a principle is less relevant if objects of interest are equal to the ground sampling distance (i.e., an image element corresponds to an object of interest and, thus, pixel-by-pixel techniques are more appropriate) or if objects of interest are smaller than the ground sampling distance (i.e., an image element covers multiple objects of interest and, thus, sub-pixel techniques are more

appropriate). The geospatial variations of representations of objects allow alleviating the limited spectral resolution by enabling the encoding of valuable additional spectral information compared to single pixels as well as consideration of shape-related properties. In general, this relation can occur in various remote sensing data. However, especially data from multispectral sensors with a very high spatial resolution (VHR) such as WorldView-I–IV or GeoEye, among others, feature such situations. Consequently, we apply the proposed method in the context of classification of VHR multispectral imagery from the WorldView-II sensor acquired over urban environments. To demonstrate the relevance of the proposed method, we compare it to classification results obtained with features computed from both single-level and multi-level segmentation. We also provide results of VSVM when learned with and without self-learning constraints. Thereby, a particular focus is on settings where only a small amount of labeled training samples is available.

The remainder of the paper is organized as follows. In Section 2 we detail the proposed method. We describe the experimental setup in Section 3. Experimental results are revealed in Section 4 and the paper is concluded and future directions are drawn in Section 5.

2 Proposed Methodology

The VSVM-SL approach consists of three main interlinked modules (Fig. 1): i) an *initial SVM model* is learned and *affiliated SVs are extracted*. ii) Those govern the *encoding of invariances* by computing additional object features, which are represented in the model as virtual samples. iii) Subsequently, a *self-learning strategy* is employed to eventually prune virtual samples and consider only informative virtual samples in conjunction with SVs for relearning the model. The functionality of VSVM is presented in Section 2.1, and the self-learning constraints are detailed in Section 2.2. Encoding of invariances is described in Section 2.3.

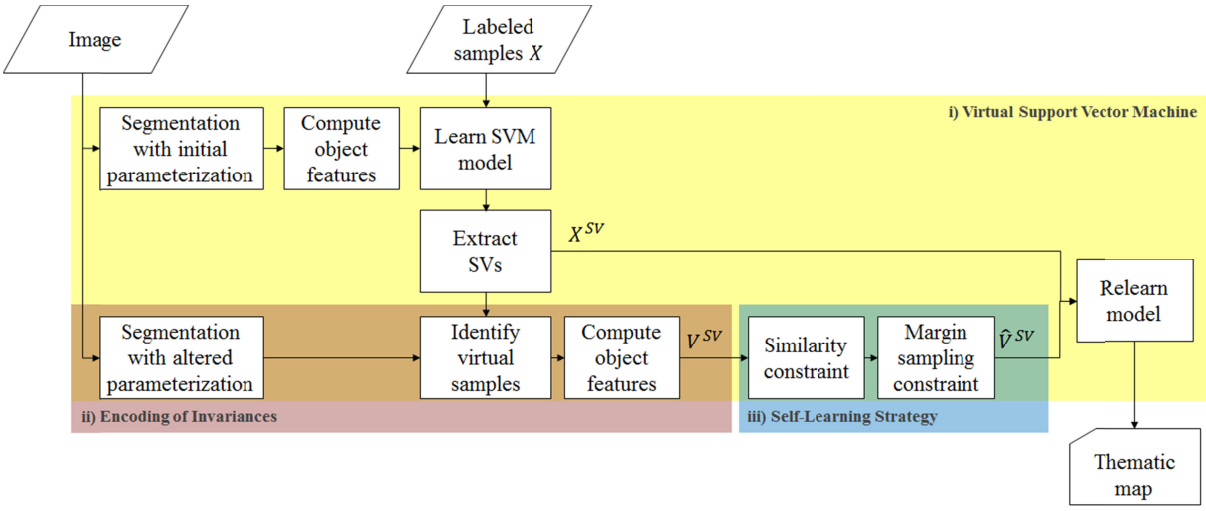


Fig. 1. Block scheme of the proposed VSVM approach with self-learning strategy

2.1 Virtual Support Vector Machines

VSVM represent a modification of the popular SVM approach (Cortes and Vapnik, 1995). The latter solves a minimization objective to establish a separating hyperplane with maximum margin between labeled samples of different classes (Fig. 2a-b). Let us consider an image $\boldsymbol{I}$ and corresponding labeled samples $X = \{\mathbf{x}_i, y_i\}_{i=1}^{n}$, with $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \{-1, +1\}$. The imagery is mapped through a nonlinear transformation $\phi(\cdot)$ to a space with a higher dimensionality. Then, the minimization objective is formulated as follows:

$$\min_{\mathbf{w},\xi_i,b} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \right\} \tag{1}$$

subject to

$$y_i(\langle \phi(\mathbf{x}_i), \mathbf{w}\rangle + b) \geq 1 - \xi_i \qquad \forall i = 1, \ldots, n \tag{2}$$

$$\xi_i \geq 0 \qquad \forall i = 1, \ldots, n \tag{3}$$

where **w** is the normal perpendicular to the optimal separating hyperplane and *b* is the nearest distance to the origin (**0**) of the coordinate system. These two parameters constitute a linear classifier, which separates the labeled samples of different classes with maximum margin. To enhance generalization capabilities and reduce over-fitting, positive slack variables ξ_i are introduced, which account for labeled samples lying on the incorrect side of the respective margin boundary. The constant *C* determines the trade-off between maximizing the margin and the number of incorrectly classified samples (training errors). The minimization objective of equation (1) is reformulated from its primal form to its dual form by introducing Lagrange multipliers, so that it can be solved efficiently with quadratic programming techniques (Schölkopf and Smola, 2002). Finally, a decision function is given that allows assigning a class label to an instance of unknown class membership $\mathbf{x}_*$

$$f(\mathbf{x}_*) = \text{sgn}\left(\sum_{i=1}^{n} y_i \alpha_i K(\mathbf{x}_i, \mathbf{x}_*) + b\right) \tag{4}$$

with α_i being the Lagrange multipliers and *K* being a kernel function. The kernel function *K* is expressed as the dot product of mapped instances $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j)\rangle$. The Lagrange multipliers are determined by optimization and feature nonzero values for instances lying on the margin – the SVs (Cortes and Vapnik, 1995; Camps-Valls and Bruzzone, 2009; Geiß et al., 2016a).

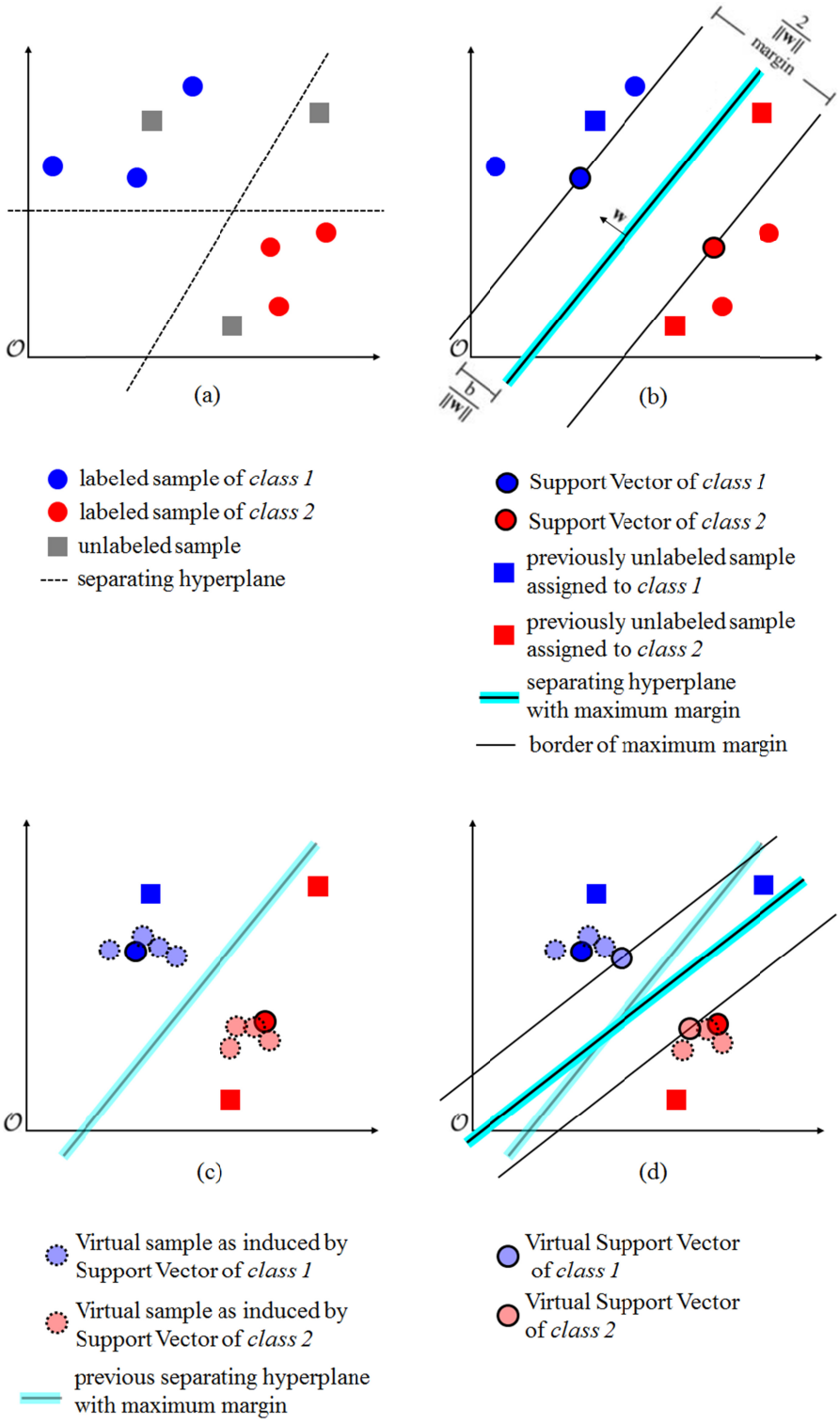


Fig. 2. Functionality of Virtual Support Vector Machines. (a) From an arbitrary set of separating hyperplanes that linearly separate the classes of the training samples, (b) first the hyperplane is established that separates the training samples with maximum margin. Only labeled samples closest to the separating hyperplane with maximum margin - the SVs - are needed to define the model. Subsequently, those SVs are used to (c) generate artificially transformed samples, i.e., virtual samples, to encode invariances. (d) The model is relearned using only labeled samples that became a SV after an initial model run in conjunction with affiliated virtual samples to eventually alter the hyperplane with maximum margin. Previously unlabeled samples are eventually relabeled according to their position with respect to the final separating hyperplane with maximum margin.

The VSVM approach builds upon an initially learned SVM model. This model is used to extract labeled samples that became a SV. Extracted SVs govern the encoding of invariances by altering previously modelled objects and recomputing affiliated features, which are added to the feature space as virtual samples (Fig. 2c). Those virtual samples are deployed in conjunction with SVs to relearn the model, which eventually alters the hyperplane with maximum margin (Fig. 2 d). In previous works, virtual samples as induced by SVs are called Virtual Support Vectors (VSVs). However, to allow for an unambiguous terminology, we only refer to VSVs if virtual samples become a sample closest to the separating hyperplane with maximum margin after the second model run.

It is important to note that hyperparameters of the VSVM model must not be optimized by cross-validation for the second model run. This is related to the circumstance that the number of virtual samples is always a multiple of the number of SVs (cf. Section 2.3), and thus it is possible to fit the model dominantly to virtual samples, while SVs lose influence. In addition to that, virtual samples are likely to show feature characteristics that resemble their corresponding SV. Hence, a consistent separation and simulation of unseen data in a cross-validation procedure might be violated when using a training set that contains virtual samples (also known as data leakage). That is why we use the holdout method (Foody, 2009) instead of cross-validation for model selection with optimal hyperparameters and consider it as an intrinsic part of the VSVM approach. Consequently, from the pool of labeled samples a training set $X_{Train} = \{\mathbf{x}_l, y_l\}_{l=1}^{j} \in X$ and test set $X_{Test} = \{\mathbf{x}_k, y_k\}_{k=j+1}^{m} \in X$ is drawn, whereby samples of the training set must not be included in the test set, i.e., $X_{Train} \cap X_{Test} = \emptyset$. To account for spatial autocorrelation, X_{Train} and X_{Test} should be also compiled in a strict spatially disjoint way to allow for unbiased estimates of model generalization capabilities (Geiß et al., 2017a). The procedure to learn VSVM is also described in the pseudocode under Algorithm 1.

Algorithm 1 Virtual Support Vector Machines

Inputs:
Pool of labeled samples: X_{Train}, X_{Test}
Output:
VSVM: SVM classifier retrained with training set $\check{X}_{Train}$

1. Learn SVM model with X_{Train}
2. Extract SVs and add them to a pool X_{Train}^{SV}
3. Perturb features based on X_{Train}^{SV} to generate a pool of virtual samples V^{SV}
4. Compile training set $\check{X}_{Train} = X_{Train}^{SV} \cup V^{SV}$
5. Learn SVM model with $\check{X}_{Train}$ and select model with optimal hyperparameters based on X_{Test}

2.2 Self-Learning Strategy

To eventually prune uninformative virtual samples from the features space, a two-step self-learning strategy is followed by consecutive consideration of a similarity and margin sampling constraint.

1) *Similarity constraint*: To only employ virtual samples that encode similar properties as their corresponding SVs, the Euclidian distance d between a virtual sample and its SV is computed in the feature space first (Fig. 3a) (Lu et al., 2016). Thereby it is assumed that virtual samples which show a large distance in the feature space with respect to their corresponding SVs do not represent reliable invariances, and are highly prone to induce divergence in the model. The distance measure d_{ij} is computed as follows:

$$d_{ij} = \sqrt{\sum_{m} (\mathbf{v}_{im}^{SV} - \mathbf{x}_{jm}^{SV})^2} \tag{5}$$

where $\mathbf{v}_i^{SV}$, $i = \{1,2,\dots,N\}$ denotes the ith virtual sample derived from $\mathbf{x}_j^{SV}$; $\mathbf{x}_j^{SV}$ is the jth SV, $j = \{1,2,\dots,N\}$, and m denotes the number of features. To eventually prune virtual samples, a maximum distance threshold δ is introduced. Since distances in the feature space are highly dependent on the scene characteristics and thematic class of interest, δ must be adjusted for an individual classification problem and a considered thematic class. To adjust δ in an automated manner, it is calculated per class as follows:

$$\delta_Q = \frac{2}{N_Q(N_Q - 1)} \sum_{i=1}^{N_Q - 1} \sum_{j=i+1}^{N} \sqrt{\sum_{m} (\mathbf{x}_{im}^{SV_Q} - \mathbf{x}_{jm}^{SV_Q})^2} \tag{6}$$

where N_Q is the number of SVs per thematic class Q, and $\mathbf{x}_{im}^{SV_Q}$ and $\mathbf{x}_{jm}^{SV_Q}$ denote the ith and jth SV, respectively, of class Q. In other words, δ_Q is the mean distance in feature space between all SVs that belong to the same thematic class Q. However, it can be beneficial to narrow δ_Q in situations, where already a large amount of information content is encoded in X^{SV}, typically induced by a high number of SVs, or to alternatively widen it in the contrary case. To introduce this flexibility to the model, we multiply δ_Q with a factor k:

$$\delta_{Qk} = \delta_Q * k \tag{7}$$

Thereby, smaller numeric values of k establish a more progressive pruning of virtual samples compared to larger numeric values. Finally, virtual samples are pruned from V^{SV} according to:

$$V_{\delta_{Qk}}^{SV} = V^{SV} \cap \left\{ \mathbf{v}_i^{SV} \middle| d_{ij} \leq \delta_{Qk} \right\} \tag{8}$$

where the pool $V_{\delta_{Qk}}^{SV}$ contains only virtual samples that lie within the radius of δ_{Qk} with respect to their corresponding SV.

2) *Margin sampling constraint*: We follow a margin sampling strategy (Tuia et al., 2009; Tuia et al., 2011; Geiß et al., 2017c) to further prune virtual samples that are located far apart from the margin and are unlikely to become a VSV in the final model and therefore contribute in a positive manner to the classification result. Consequently, virtual samples which were not pruned from the feature space by the similarity constraint are mapped into kernel space and only virtual samples in close distance to the hyperplane are selected. To specify the maximum acceptable distance of a virtual sample to the hyperplane the threshold l is introduced (Fig. 3b). The distance of a virtual sample to the hyperplane can be determined for a binary SVM by adapting the decision function (4) according to:

$$f(\mathbf{v}_{\delta_{Qkj}}^{SV}) = \sum_{i=1}^{n} y_i \alpha_i K\left(\mathbf{x}_i, \mathbf{v}_{\delta_{Qkj}}^{SV}\right) + b \tag{9}$$

where $\mathbf{v}_{\delta_{Qkj}}^{SV}$ is a virtual sample fulfilling the similarity constraint. In this work, we deploy a one-against-one architecture for multiclass problems. Consequently, a virtual sample is fulfilling the margin sampling constraint if its distance to the hyperplane is smaller than l for at least one of the class-specific hyperplanes.

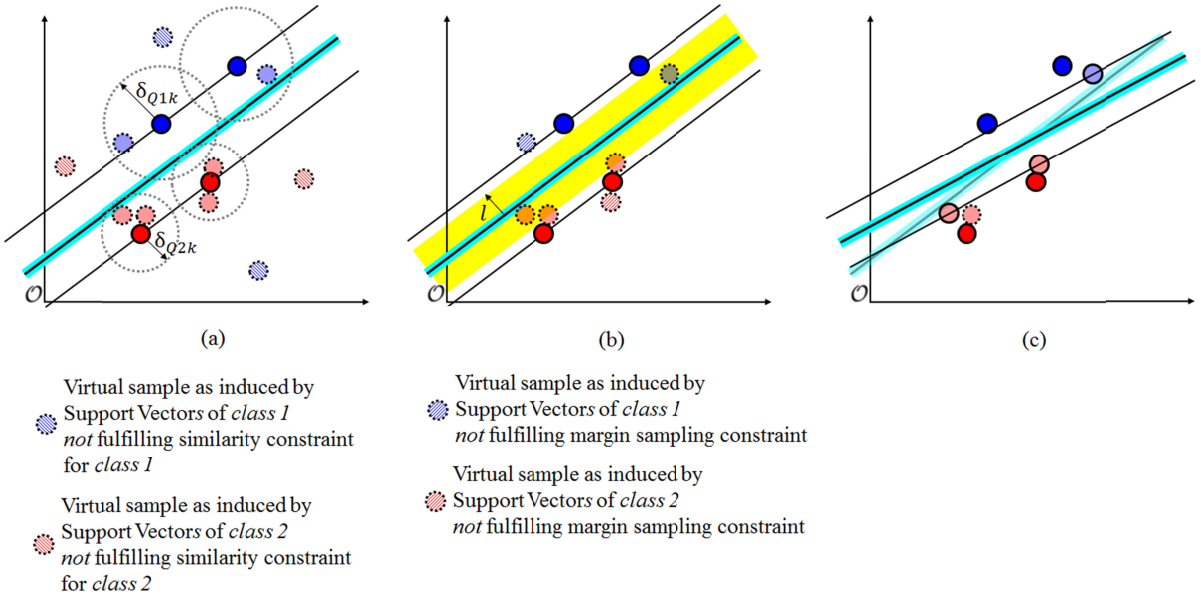


Fig. 3. Functionality of Self-Learning Strategy. (a) Virtual samples are evaluated with respect to their Euclidean distance to the affiliated SVs and are pruned from the set of training samples if they exceed an empirically determined class-specific distance. (b) Residual virtual samples are evaluated with respect to their position to the margin and are pruned from the set of training samples if they exceed a predefined distance. (c) The model is relearned using SVs and residual virtual samples to eventually alter the hyperplane with maximum margin.

The determination of the hyperparameters of the self-learning strategy (i.e., k, l) is rendered as a further minimization objective. Thereby, the combination which reveals the highest estimated generalization capabilities, as evaluated based on a classification accuracy measure, is selected. Analogous to the unconstrained VSVM framework, the final pool of constrained virtual samples $\hat{V}^{SV}$ is deployed in conjunction with SVs to relearn the model (Fig. 3c). The self-learning strategy to prune virtual samples from V^{SV} is also described in the pseudocode under Algorithm 2.

Algorithm 2 Self-learning strategy

Inputs:
Pool of SVs: X^{SV}
Pool of virtual samples: V^{SV}
Output:
A pool of constrained virtual samples: $\hat{V}^{SV}$

For $i = 1$ **to** N **in** V^{SV}

1. Compute Euclidean distance d_{ij} between $\mathbf{v}_i^{SV}$ and $\mathbf{x}_j^{SV}$;

End

2. Compute class-specific maximum distance thresholds δ_{Qk} according to (6) and (7);
3. Remain virtual samples which satisfy $d_{ij} \leq \delta_{Qk}$, and prune the others from V^{SV} according to (8) to establish a pool $V_{\delta_{Qk}}^{SV}$, which contains only virtual samples that lie within the radius of δ_{Qk};

For $i = 1$ **to** N **in** $V_{\delta_{Qk}}^{SV}$

4. Compute distance to hyperplane for class Q with decision function according to (9);

End

5. Remain virtual samples which satisfy the maximum acceptable distance l, and prune the others from $V_{\delta_{Qk}}^{SV}$ to establish a final pool of constrained virtual samples $\hat{V}^{SV}$;

2.3 Invariances

As stated in Section 1, encoding of invariances is established within an OBIA framework (Blaschke, 2010). There, in general, real-world objects are modelled with a segmentation algorithm and are represented as segments (i.e., superpixels). Given a complex classification scenario and very limited labeled samples, it is very likely that objects of thematic classes of interest are represented solely by a subset of its existing object variations in the training set. This is related to the circumstance that only a very limited number of objects are labeled, and optimal representation of all objects with corresponding segments is very challenging to achieve due to over- and undersegmentation (Taubenböck et al., 2010). To address the latter problem, approaches were designed to create an optimized single segmentation level from multi-level segmentation objectively (Geiß et al., 2016a). However, it remains very challenging to ensure an optimal object representation within a single segmentation level especially in complex environments such as urban areas due to a large variety of objects with completely different size and shape properties (Geiß et al., 2017b). Consequently, we follow a strategy to encode invariances by altering the hyperparameters of a segmentation algorithm (details on the segmentation algorithm and affiliated parametrization can be found in Section 3.2, where the experimental setup is described), which generates the segment-based

representation of objects, with respect to size (i.e., scale) and shape characteristics. From the segmentation level with an initial parameterization, object features are computed, an SVM model is learned, and corresponding SVs are extracted. In parallel, segmentation levels based on altered parameterizations with respect to the initial parameterization are generated (cf. process "segmentation with altered parametrization" in Fig. 1). Subsequent to that, SVs are located in the image domain and segments from the different segmentation levels are selected if they contain an SV (cf. Fig. 4 in the subsequent Section). Those segments are used to compute object features, which are represented in the feature space as virtual samples (cf. Fig. 2c). Consequently, the number of virtual samples corresponds to the number of SVs multiplied with the number of segmentation levels considered (cf. Fig. 4 in the subsequent Section).

1) *Invariances of Scale*

To generate virtual samples that aim to make the model invariant with respect to scale (i.e. size of the objects), we followed a hierarchical multi-level segmentation procedure (Geiß and Taubenböck, 2015; Geiß et al., 2016; Bruzzone and Carlin, 2006; Taubenböck et al., 2010; Aravena Pelizari, 2018). Thereby, $\boldsymbol{I}$ is partitioned based on a segmentation algorithm with a fixed set of shape-related hyperparameters g at a generic segmentation level s_g in N^{s_g} segments $O_i^{s_g}$ $(i = 1, 2, \dots, N^{s_g})$. To establish an unambiguous hierarchy of segmentation levels, the following constraint must be fulfilled:

$$\bigcup_{O_i^{s_g-1} \subseteq O_j^{s_g}} O_i^{s_g-1} = O_j^{s_g} \tag{10}$$

This way it is ensured that a segment at segmentation level $s_g - 1$ must be included in only one segment at level s_g (Bruzzone and Carlin, 2006). The computation of virtual samples is based on a series of generated hierarchical segmentation levels $S_g \in \{s_g - n, \dots, s_g - 1, s_g, s_g + 1, \dots, s_g + n\}$ from $\boldsymbol{I}$ (Fig. 4). Sizes of segments as generated by different scale parameters range between lower bounds (i.e., as obtained with scale parameters$s_g - n$) and upper bounds (i.e., as obtained with scale parameters$s_g + n$). The lower bound enables good representations of the smallest and most homogeneous real-world objects contained in $\boldsymbol{I}$ while inducing oversegmentation for other real-world objects. In contrast to that, the upper bound allows for a good representation of the largest and most heterogeneous real-world objects contained in $\boldsymbol{I}$ while inducing undersegmentation for other real-world objects.

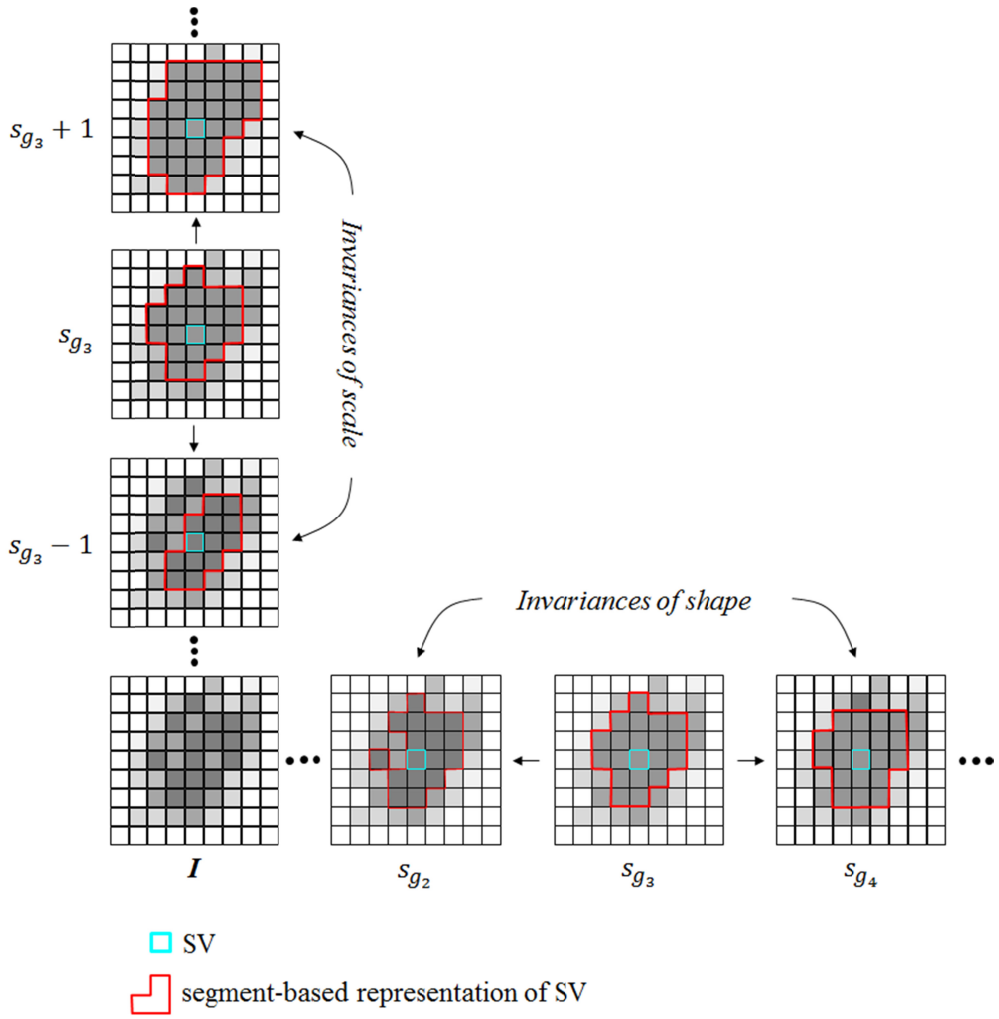


Fig. 4. Example for generation of virtual samples with respect to scale and shape. Here, $\boldsymbol{s_{g_3}}$ corresponds to the initial parameterization of the segmentation algorithm, which is used to identify SVs. Segments which contain an SV are selected and object features are computed for segments with altered parameterization (i.e., invariances of scale and shape).

2) *Invariances of Shape*

To generate virtual samples that aim to allow for an invariant model with respect to shape of the objects, we modify hyperparameters of the segmentation algorithm that influence the geometrical properties of modelled segments while keeping the scale parameter constant. Concordant to the processing scheme of scale invariance, a series of segmentation levels is established, i.e., $G_s \in \{s_{g_1}, s_{g_2}, \ldots, s_{g_n}\}$. Thereby, equation (10) is violated since a hierarchical structure cannot be followed. Instead, segments are created that feature approximately the same size but different geometries (Fig. 4).

Generally, the modifications of hyperparameters to establish S_g and G_s should be carried out in an exhaustive way to capture the entire spectrum of object scale and object shape present in the imagery for different thematic classes. As such, a more complex classification problem can benefit from more exhaustive modifications of hyperparameters. Thereby, it is acceptable if under- and oversegmentation occurs since the self-learning strategy is designed to prune misleading virtual samples in an automated manner from the model again. As such,

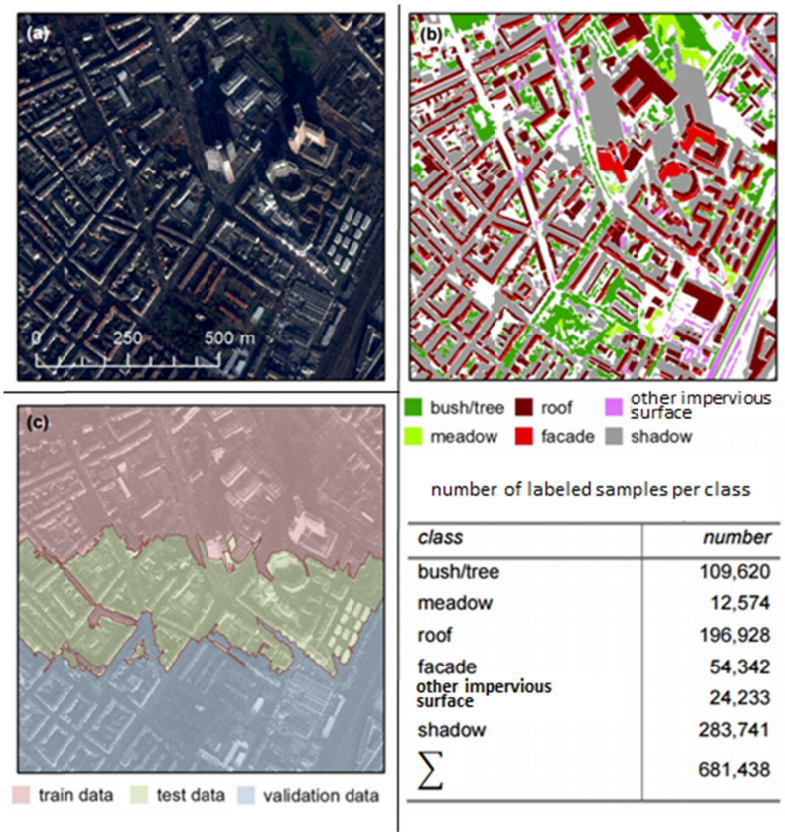
modification of hyperparameters can be regarded as tradeoff between computational burden and exploration of potentially useful information.

3 Data and Experimental Setup

3.1 VHR Multispectral Data

The experimental analysis was carried out by classifying two test data sets, each covering a spatial extent of 1000 x 1000m. Both data sets were taken from VHR multispectral (blue/green/red/near-infrared) imagery with a geometric resolution of 0.5m acquired by the WorldView-II sensor.

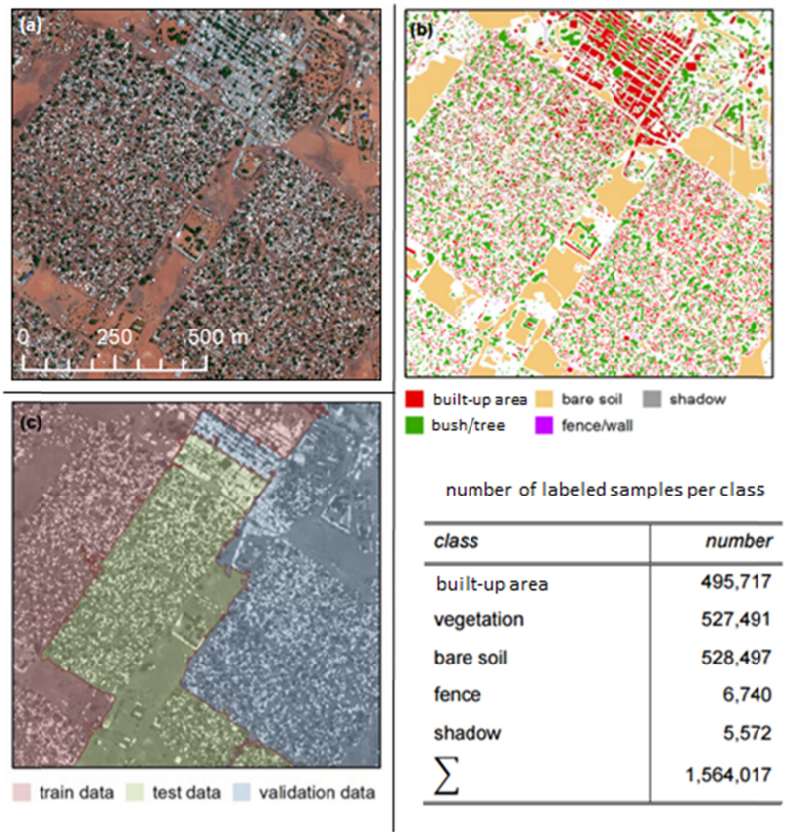
The first data set was acquired over the city of Cologne, Germany, on January 31, 2014 and shows an urban area which is dominated by buildings of commercial use (Fig. 5a). It features a complex composition of urban land cover. Shadow areas can be observed primarily adjacent to buildings. In addition, the imagery represents an off-nadir acquisition. As such, facades of individual buildings can be identified in the direction of the sensor view. To reduce the computational burden for the experiments, we resampled the subset with a nearest neighbor interpolation to 1000 x 1000 pixels with 1m pixel spacing. The pixels were organized in six relevant thematic classes, namely “bush/tree”, “meadow”, “roof”, “facade”, “shadow”, and “other impervious surface” (Fig. 5b). The latter class comprises impenetrable surfaces other than building-related ones such as roads or parking lots, which feature similar spectral characteristics. The thematic classes were determined based on photointerpretation analysis under consideration of additional aerial imagery and cadastral maps (Geiß and Taubenböck. 2015; Geiß et al., 2016b).



class	number
bush/tree	109,620
meadow	12,574
roof	196,928
facade	54,342
other impervious surface	24,233
shadow	283,741
∑	681,438

Fig. 5. (a) WorldView-II scene of Cologne, Germany; (b) available labeled samples per class; (c) spatially disjoint training, testing, and validation areas.

The second data set is a 2000 x 2000 pixel subset of an acquisition over the Hagadera refugee camp in Kenya, from March 01, 2012 (Fig. 6a). Unlike the Cologne data set, the original spatial resolution of 0.5m was kept. The test area shows a complex settlement pattern composed of buildings which are heterogeneous in size, orientation and materials, trees and bushes, fences, walls, as well as associated shadows and open spaces. Accordingly, we differentiate five thematic classes: “built-up area”, “bush/tree”, “bare soil”, “fence/wall”, and “shadow” with regard to our classification experiments (Fig. 6b). Labeled samples were derived by visual image interpretation under consideration of a camp map distributed by the United Nations High Commissioner for Refugees (UNHCR, 2012).



number of labeled samples per class

class	*number*
built-up area	495,717
vegetation	527,491
bare soil	528,497
fence	6,740
shadow	5,572
$\sum$	1,564,017

Fig. 6. (a) WorldView-II scene of Hagadera Refugee Camp, Kenya; (b) available labeled samples per class; (c) spatially disjoint training, testing, and validation areas.

3.2 Experimental Setup

Regarding the encoding of invariances, we used an iterative bottom-up region-growing segmentation algorithm for partition of $\boldsymbol{I}$ (i.e., fractal net evolution approach as implemented in the software environment eCognition (Baatz and Schäpe, 2000). There, segments are modeled according to the so-called scale parameter$H$. It internalizes spectral homogeneity ($h_{color} \in [0, \dots, 1]$) and shape homogeneity ($h_{shape} \in [0, \dots, 1]$), where $h_{shape} = 1 - h_{color}$. Further, h_{shape} is being composed of smoothness h_{smooth} and compactness $h_{compact}$ of segment boundaries ($h_{smooth} = 1 - h_{compact}$). Fusion of adjacent objects is enabled with local mutual best fit, whereby lowest increase of object heterogeneity within the merging process is ensured. An increasing value of H corresponds to an increasing insensitivity of the object fusion criteria, and thus the objects become larger (Martinis et al., 2011).

For establishing the segmentation level with an initial parameterization, we put more emphasis on shape heterogeneity rather than on gray-value heterogeneity. This is due to the circumstance that manmade structures such as buildings and other elements of urban environments have distinct shape and size properties, unlike, for example, natural features. Analogously, the weights for heterogeneity of smoothness and compactness were maintained equal (i.e., $\mathrm{h_{shape}}$: 0.7; $h_{compact}$: 0.5). A suitable value for H to balance the level of under- and oversegmentation for the initial segmentation was found to be 20 for the Cologne imagery and 25 for the Hagadera imagery. In the experiments, when exploring invariances of scale, we established nine additional segmentation levels with altered values for

$H = \{10,15,25,30,35,40,50,60,80\}$ for the Cologne data set and seven additional segmentation levels with altered values for $H = \{15,20,30,35,40,50,60\}$ for the Hagadera data set. When exploring invariances of shape, we used the following alterations of h_{shape} and $h_{compact}$: (0.1;0.9), (0.3;0.7), (0.3;0.5), (0.5;0.7), (0.5;0.5), (0.5;0.3), (0.7;0.3), (0.9;0.1) for both data sets.

Various features were compiled for characterization of modelled objects. Mean and standard deviation values of the different image bands were calculated. In addition, the *Normalized Differenced Vegetation Index* was computed. The spectral information was also used to compute rotation-invariant texture measures based on the grey-level co-occurrence matrix (GLCM) (Haralick et al., 1973), since it was shown that such features can provide supplementary information if the spectral resolution is limited and the ground sampling distance is much smaller than the objects of interest. Thereby, we selected three measures from the set of 14 originally proposed GLCM measures, since some are strongly correlated with each other (Pacifici et al., 2009). Namely, the GLCM measures *mean*, *homogeneity*, and *dissimilarity* were computed. Lastly, features were computed which approximate the shape of objects based on a comparison with two-dimensional geometrical forms such as square, rectangle, or ellipse (i.e., rectangular and elliptic fit, roundness, shape index, and compactness) (Sun et al., 2015). All features were computed in the software environment eCognition (Trimble, 2014) with already implemented or customized protocols. Numerical values of the different features using the segmentation with initial parameterization were normalized to a 0-1 interval. Feature values from segmentations with altered parameterization were aligned correspondingly.

For the actual SVM learning procedures, we deployed Gaussian RBF kernels, that take the form $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2)$, due to their interpretability in accordance with favorable performance properties in environmental applications (Volpi et al., 2013). Learning the most appropriate C-SVM in conjunction with an RBF kernel requires the definition of the cost parameter C and the kernel-width parameter γ. We carried out an exhaustive optimization of hyperparameters with $C = \{2^{-4}, 2^{-3}, \dots, 2^{12}\}$ and $\gamma = \{2^{-5},\ 2^{-4.5}, \dots, 2^{3}\}$. A one-against-one SVM architecture was deployed for multiclass problems. Hyperparameters of the self-learning strategy were optimized as follows: $\Phi \in \{k, l\};\ k = \{0.3,0.6,0.9\}$, $l = \{0.5,1.0,1.5\}$.

In the experiments, we evaluate the methods with respect to both, a binary and multiclass classification setting. To address the first, we aim to separate the class “bush/tree” from the residual classes for the Cologne data set, and “built-up area” from the residual classes for the Hagadera data set. In the multiclass classification setting, we aim to distinguish all six and five thematic classes contained in the Cologne and Hagadera data set, respectively. Labeled samples were drawn randomly in a stratified manner from the train and test set, whereby the same number of labeled samples per class was used for learning and selecting a model. The number of samples was varied to test sensitivity with respect to accuracy. To avoid a biased quantification of the effect of training set size on prediction accuracy, it was made sure that samples contained in one set are also contained in the affiliated set with a larger number of samples. Generalization capabilities are evaluated based on global accuracy measures comprising κ statistic, weighted mean $\overline{F_1}$ of F_1-measures, overall accuracy (OA), average

accuracy (AA), as well as class individual accuracies. Results are reported as average of 20 independent trials with affiliated standard deviation. To get insights into the complexity of learned models, we also provide the mean number of samples used for establishing a model.

4 Experimental results and discussion

Results are presented for the VSVM with and without self-learning strategy (i.e., VSVM-SL and VSVM), respectively. As a benchmark, we present accuracies obtained with an SVM, which is learned based on features from the segmentation level with initial parameterization (i.e., the model which is used for the extraction of SVs). In addition to that, an SVM is learned using multi-level segmentation (i.e., segmentations with altered parameterization; referred to as SVM-M), whereby additionally encoded object characteristics are represented in the model not as virtual samples but as additional features (a comparable approach can be found in (Bruzzone and Carlin, 2006).

4.1 Experimental results from data set I: Cologne

Fig. 7 contains obtained accuracy estimates in terms of κ statistic as a function of differing numbers of samples used for model learning and selection. To track model complexity and evaluate classification accuracy with respect to affiliated number of samples, corresponding mean numbers of samples used for establishing a model (i.e., SVs for the SVM-based methods and sum of SVs and VSVs for the VSVM-based methods) are also presented. The binary classification setting (Fig. 7a) shows a similar accuracy pattern for both types of invariances (i.e., scale and shape), whereby the VSVM-SL approach clearly allows obtaining higher accuracies compared to the other methods in situations where solely very few labeled samples are available. Simultaneously, a plateau on a high accuracy level is reached more rapidly. Generated models are also more robust as indicated by narrower standard deviations of κ statistics, and need considerably less samples for establishing a model compared to the unconstrained VSVM. Results from 20 realizations with 40 and 34 samples per class for model learning and selection regarding the scale and shape invariances, respectively (Table I), unambiguously underline the superior solutions provided by VSVM-SL in this example. With this model almost all accuracy measures achieved highest values, and e.g., mean κ statistics exceed the other methods constantly by more than five percentage points. These numbers are also mirrored in the corresponding classification map (Fig. 8a), where the VSVM-SL approach features less errors of commission regarding the class “bushes/trees”, and provides a spatially well regularized classification map. This favorable performance pattern can be attributed to the fact that the land cover class “bushes/trees” mainly encodes invariances in this example, and that the classification problem itself is comparably easy. Consequently, class distributions can be generally described well with few samples.

Results from the multiclass classification setting are depicted in Fig. 7b. In this setting, VSVM-SL and VSVM perform equally well and consistently outperform the two benchmark approaches. The governing principle to represent additional knowledge in the model as virtual samples and not as features shows favorable properties, especially in situations with very few

labeled samples. This is particularly indicated by the poor accuracies obtained with the multi-level segmentation approach (i.e., SVM-M) compared to the other methods, and can be related to the circumstance that the relation between number of labeled samples and number of features can become problematic in such settings (i.e., induces phenomena associated to the curse of dimensionality). Intuitively, more labeled samples are needed in the multiclass setting compared to the binary setting to reach a plateau on a high accuracy level. There it can be observed that both virtual samples-based methods still feature higher accuracies in terms of global accuracy measures compared to the two benchmark approaches (Table II). The VSVM-SL achieves best overall performance for the invariances of scale, whereas the VSVM features the best overall results for the invariances of shape, respectively. However, the VSVM-SL needs approximately solely half of the number of samples compared to the unconstrained approach to establish the models. Corresponding classification maps are shown in Fig. 8b. They underline the capability of the virtual samples-based methods to provide a spatially smooth and accurate classification map.

Overall, the results for this data set show the favorable performance properties of VSVM and VSVM-SL, whereby the self-learning strategy is particular useful in very small sample settings and obtaining sparse model solutions.

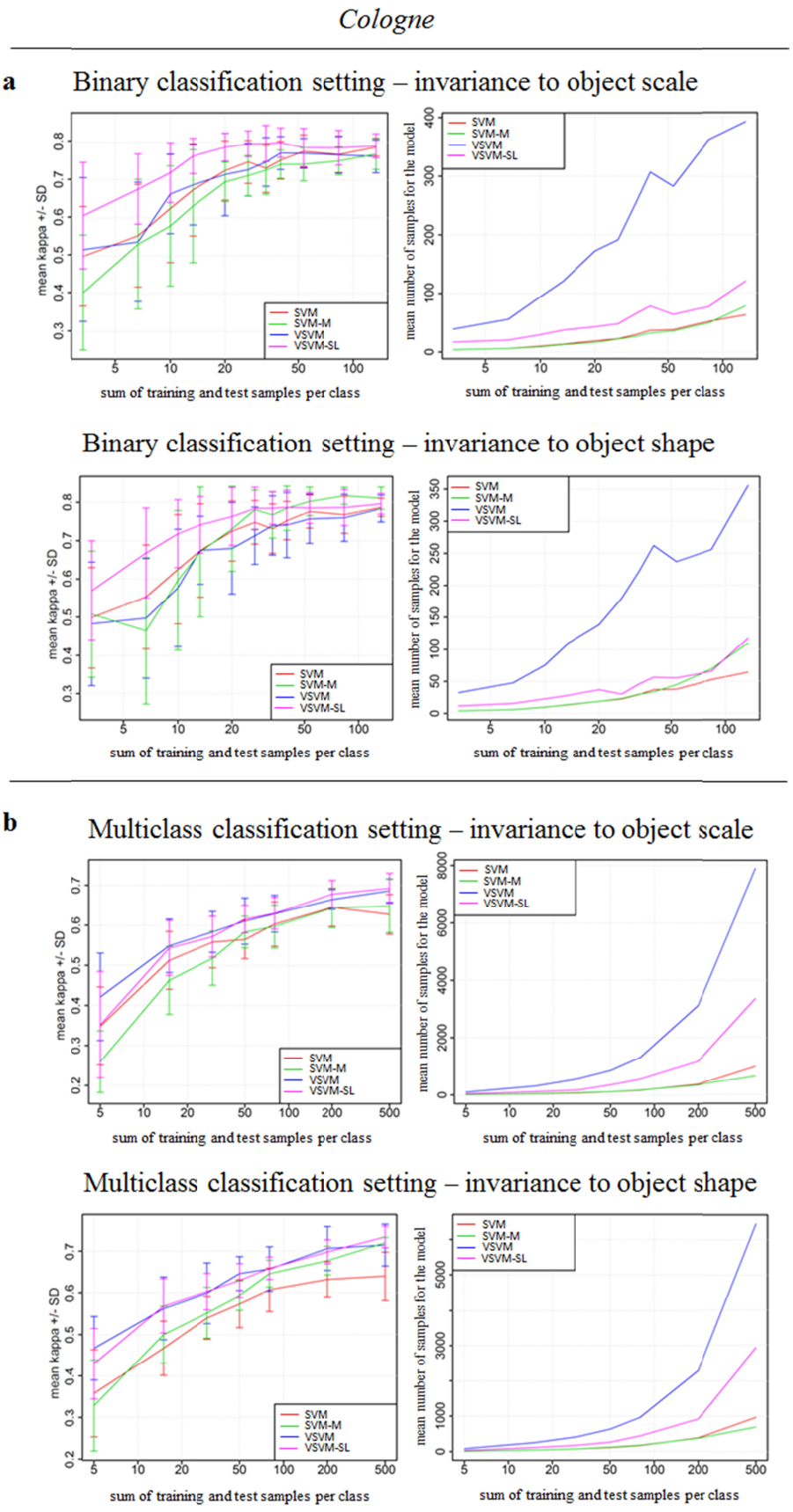


Fig. 7. κ statistics (reported as mean and standard deviation from twenty realizations with a varying configuration of labeled samples) and mean number of samples used for establishing a model for the different methods as a function of sum of training and test samples per class for the Cologne data set. (a) results for the binary classification setting; (b) results for the multiclass classification setting.

TABLE I

STUDY AREA 1 (COLOGNE): CLASSIFICATION ACCURACIES OBTAINED WITH THE DIFFERENT METHODS FOR THE BINARY CLASSIFICATION SETTING REPORTED AS MEAN AND STANDARD DEVIATION (IN BRACKETS) FROM TWENTY REALIZATIONS WITH A VARYING CONFIGURATION OF LABELED SAMPLES. OBTAINED RESULTS FROM A SINGLE REALIZATION ARE ALSO VISUALIZED IN FIG. 8A.

Invariance to object scale; number of samples used for model learning and selection: 40

Class	SVM		SVM-M		VSVM		VSVM-SL	
	F_1	ACC	F_1	ACC	F_1	ACC	F_1	ACC
Bush/tree	78.99 (± 5.82)	67.39 (± 8.33)	77.21 (± 4.74)	73.52 (± 5.47)	78.59 (± 6.19)	67.18 (± 8.58)	**82.52** (± 3.86)	**75.56** (± 6.28)
Other	92.84 (± 2.99)	**99.00** (± 0.71)	93.81 (± 1.26)	95.36 (± 2.00)	92.66 (± 3.56)	98.81 (± 0.80)	**94.89** (± 1.46)	97.81 (± 1.45)
κ	72.29 (± 8.24)		71.07 (± 5.89)		71.76 (± 8.87)		**77.53** (± 5.14)	
$\overline{F_1}$	90.61 (± 3.43)		91.14 (± 1.77)		90.40 (± 3.96)		**92.90** (± 1.82)	
OA	89.35 (± 4.03)		90.28 (± 1.95)		89.12 (± 4.61)		**92.11** (± 2.11)	
AA	83.19 (± 4.07)		84.44 (± 2.91)		83.00 (± 4.23)		**86.68** (± 2.95)	

Invariance to object shape; number of samples used for model learning and selection: 34

Class	SVM		SVM-M		VSVM		VSVM-SL	
	F_1	ACC	F_1	ACC	F_1	ACC	F_1	ACC
Bush/tree	78.23 (± 4.65)	65.93 (± 6.94)	79.6 (± 6.08)	68.21 (± 8.9)	79.36 (± 4.64)	68.79 (± 8.45)	**81.82** (± 3.59)	**77.15** (± 6.78)
Other	92.55 (± 2.37)	**99.14** (± 0.59)	92.86 (± 3.11)	99.05 (± 0.8)	93.19 (± 2.33)	98.78 (± 1.56)	**94.92** (± 1.24)	96.92 (± 1.78)
κ	71.25 (± 6.59)		72.91 (± 8.65)		72.93 (± 6.56)		**76.81** (± 4.69)	
$\overline{F_1}$	90.24 (± 2.73)		90.72 (± 3.58)		90.96 (± 2.68)		**92.81** (± 1.58)	
OA	88.91 (± 3.22)		89.45 (± 4.20)		89.79 (± 3.19)		**92.07** (± 1.82)	
AA	82.53 (± 3.34)		83.63 (± 4.35)		83.78 (± 3.74)		**87.04** (± 3.06)	

TABLE II

STUDY AREA 1 (COLOGNE): CLASSIFICATION ACCURACIES OBTAINED WITH THE DIFFERENT METHODS FOR THE MULTICLASS CLASSIFICATION SETTING REPORTED AS MEAN AND STANDARD DEVIATION (IN BRACKETS) FROM TWENTY REALIZATIONS WITH A VARYING CONFIGURATION OF LABELED SAMPLES. OBTAINED RESULTS FROM A SINGLE REALIZATION ARE ALSO VISUALIZED IN FIG. 8B.

Invariance to object scale; number of samples used for model learning and selection: 200

Class	SVM		SVM-M		VSVM		VSVM-SL	
	F_1	ACC	F_1	ACC	F_1	ACC	F_1	ACC
Bush/tree	**83.64** (±2.41)	80.66 (±2.78)	81.58 (±4.45)	**83.59** (±2.51)	82.48 (±2.74)	81.79 (±2.20)	83.27 (±1.96)	80.58 (±3.42)
Meadow	52.97 (±6.44)	40.30 (±7.95)	**55.55** (±11.11)	**47.84** (±18.51)	47.30 (±7.38)	33.42 (±7.87)	51.99 (±6.31)	39.26 (±7.20)
Roof	58.34 (±9.41)	78.78 (±3.88)	**67.18** (±7.52)	72.68 (±5.30)	64.84 (±5.16)	**80.50** (±1.76)	67.11 (±7.55)	78.99 (±2.77)
Facade	57.74 (±3.35)	49.56 (±3.96)	56.06 (±4.52)	49.50 (±8.58)	**58.91** (±2.39)	**52.27** (±3.62)	56.86 (±3.56)	47.99 (±3.83)
Other imp. surf.	37.62 (±7.25)	27.06 (±6.98)	14.92 (±7.61)	26.20 (±15.01)	44.08 (±6.12)	32.10 (±5.51)	**46.09** (±8.21)	**37.12** (±8.92)
Shadow	87.18 (±1.10)	90.63 (±1.50)	84.50 (±1.52)	82.13 (±2.82)	**88.17** (±1.04)	**91.70** (±1.17)	87.48 (±1.28)	91.52 (±1.35)
κ	63.49 (±4.62)		63.40 (±3.59)		66.09 (±3.05)		**67.25** (±3.94)	
$\overline{F_1}$	73.54 (±3.46)		73.75 (±2.16)		75.86 (±2.01)		**76.35** (±2.76)	
OA	72.73 (±3.78)		73.84 (±2.83)		74.77 (±2.46)		**75.77** (±3.24)	
AA	61.16 (±2.96)		60.32 (±3.23)		61.96 (±1.96)		**62.58** (±2.18)	

Invariance to object shape; number of samples used for model learning and selection: 200

Class	SVM		SVM-M		VSVM		VSVM-SL	
	F_1	ACC	F_1	ACC	F_1	ACC	F_1	ACC
Bush/tree	82.94 (± 1.99)	80.58 (± 2.34)	84.83 (± 1.85)	**83.88** (± 1.92)	**85.84** (± 2.71)	82.72 (± 2.98)	84.73 (± 2.65)	82.21 (± 2.64)
Meadow	51.02 (± 6.55)	38.31 (± 8.34)	47.14 (± 4.54)	33.42 (± 4.36)	**57.62** (± 8.51)	**48.53** (± 13.23)	50.69 (± 9.36)	42.09 (± 11.42)
Roof	55.45 (± 10.75)	78.10 (± 3.61)	64.03 (± 4.80)	76.58 (± 4.58)	67.29 (± 8.58)	**78.95** (± 2.46)	**68.80** (± 6.56)	75.78 (± 3.43)
Facade	56.27 (± 4.49)	48.48 (± 5.96)	56.02 (± 4.17)	47.72 (± 5.1)	**58.60** (± 2.69)	**55.53** (± 4.10)	54.37 (± 3.87)	45.50 (± 5.09)
Other imp. surf.	37.24 (± 9.71)	26.86 (± 9.38)	**48.02** (± 4.29)	35.67 (± 4.14)	43.70 (± 13.70)	35.07 (± 13.32)	46.07 (± 11.92)	**40.73** (± 11.97)
Shadow	86.90 (± 1.01)	90.21 (± 1.28)	**88.65** (± 1.58)	**93.33** (± 0.94)	87.91 (± 1.50)	90.04 (± 1.68)	86.93 (± 1.85)	90.93 (± 1.80)
κ	61.96 (± 5.19)		67.06 (± 2.72)		**68.32** (± 5.12)		67.86 (± 4.59)	
$\overline{F_1}$	72.30 (± 3.80)		76.11 (± 2.07)		**77.15** (± 3.44)		76.62 (± 3.06)	
OA	71.49 (± 4.27)		75.40 (± 2.16)		**76.76** (± 3.96)		76.38 (± 3.67)	
AA	60.42 (± 2.87)		61.76 (± 2.29)		**65.14** (± 4.01)		62.88 (± 4.19)	

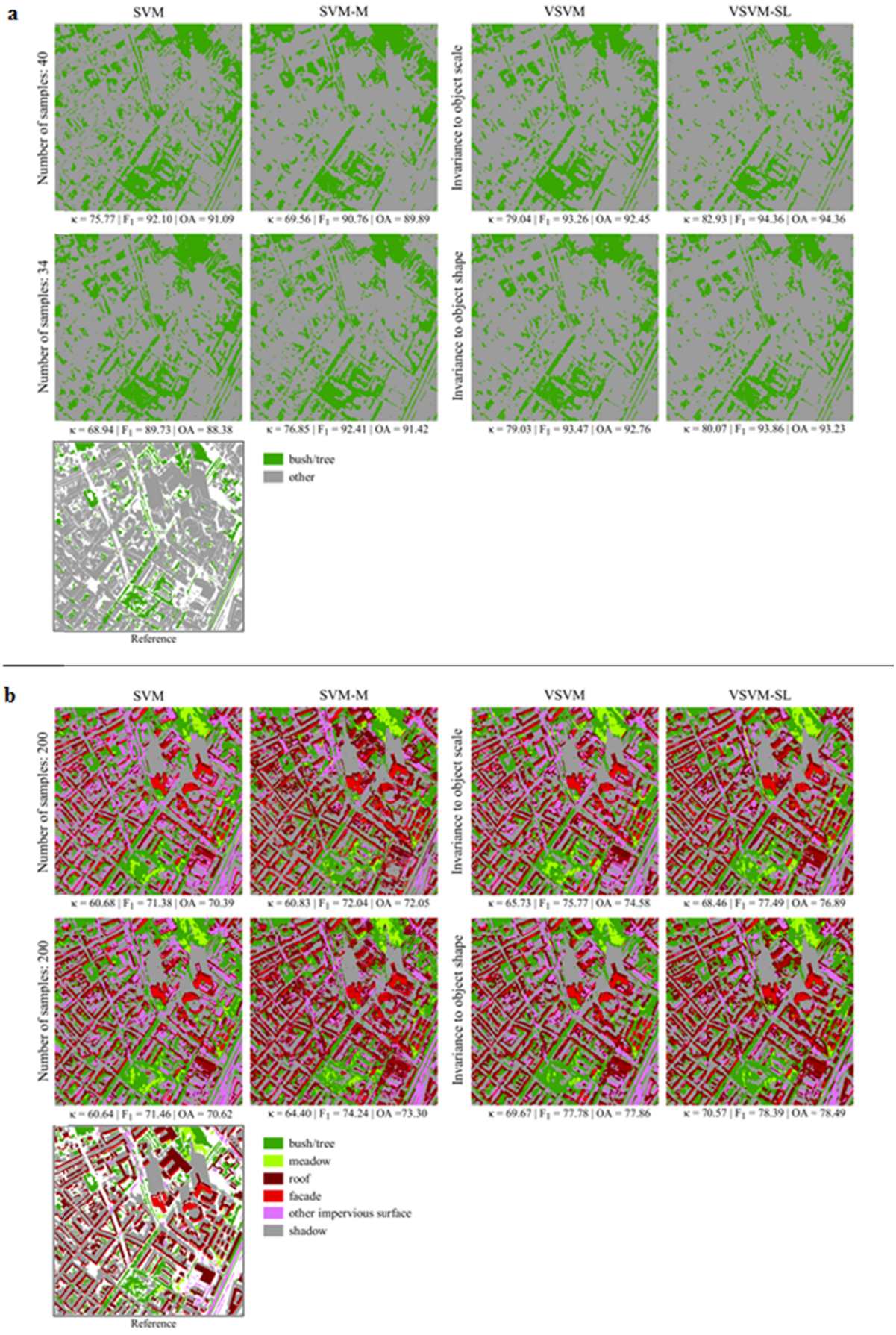


Fig. 8. Visualized results from a single realization with the different methods for the binary (a) and multiclass (b) classification setting for the Cologne data set.

4.2 Experimental results from data set II: Hagadera

Analogous to the presentation of results from data set I, we also provide obtained accuracies as a function of differing number of samples, and corresponding mean numbers of samples used for establishing a model for data set II (i.e., SVs for the SVM-based methods and sum of SVs and VSVs for the VSVM-based methods) (Fig. 9). For the binary classification setting it can be observed that the virtual samples-based approaches show better accuracies in comparison to the benchmark approaches (Fig. 9a). Thereby, VSVM and VSVM-SL alternate with respect to the best model accuracies until all methods converge to a plateau of maximum accuracy when more than 50 samples per class were used for model learning and selection. As a further means, results from 20 realizations with 20 and 34 samples per class for model learning and selection regarding the scale and shape invariances, respectively are provided in Table III. It can be observed that invariances of scale were exploited by VSVM in the most beneficial way, and the VSVM-SL method enables the highest accuracies with respect to invariances of shape. These numbers are also reflected in the corresponding classification map (Fig. 10a). The maps obtained with VSVM and VSVM-SL feature considerably less errors of commission regarding the class “built-up area” for both types of invariances while keeping a spatially well regularized yet fine-grained classification map.

Results from the multiclass classification setting are shown in Fig. 9b. In this example, the VSVM-based models clearly outperform the other methods for settings with very few labeled samples. However, they are directly followed by the VSVM-SL-based models. Analogous to the binary classification setting, all methods converge to a plateau of maximum accuracy when models are learned and selected with 50 or more samples per class. Nevertheless, Table IV documents in detail the beneficial performance properties of the VSVM approach in very challenging classification settings with solely 20 samples per class. It permits the most favorable global as well as class individual accuracies. Corresponding classification maps are provided by Fig. 10b. Generally, it can be observed that there are predominately commission errors with respect to the classes “fence” and “shadow”. These can be attributed to their non-discriminative spectral appearance in the imagery and frequent occurrence in direct spatial proximity. However, the VSVM provides least errors of omission with respect to the residual land cover classes and also allows for the spatially most homogeneous and consistent mapping results. This can be related to the circumstance that the VSVM encodes most additional prior knowledge in the model in a beneficial way, what is favorable here to learn discriminative functions for multiple complex class distributions.

In conclusion, also the results for this data set demonstrate the beneficial characteristics of the virtual samples-based methods, whereby the unconstrained VSVM particularly allowed for the best accuracies in very challenging classification problems.

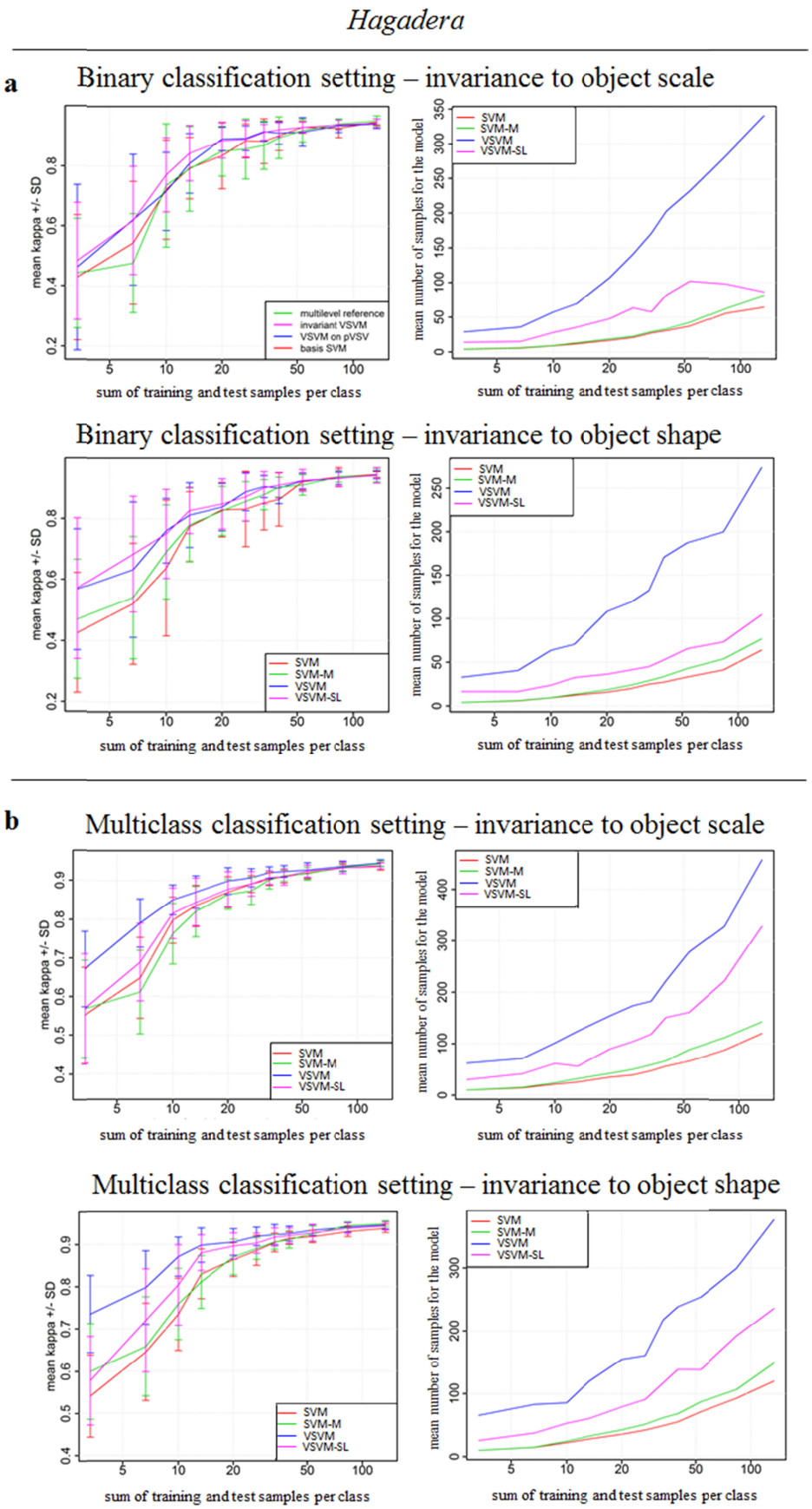


Fig. 9. κ statistics (reported as mean and standard deviation from twenty realizations with a varying configuration of labeled samples) and mean number of samples used for establishing a model for the different methods as a function of sum of training and test samples per class for the Hagadera data set. (a) results for the binary classification setting; (b) results for the multiclass classification setting.

TABLE III

STUDY AREA 2 (HAGADERA): CLASSIFICATION ACCURACIES OBTAINED WITH THE DIFFERENT METHODS FOR THE BINARY CLASSIFICATION SETTING REPORTED AS MEAN AND STANDARD DEVIATION (IN BRACKETS) FROM TWENTY REALIZATIONS WITH A VARYING CONFIGURATION OF LABELED SAMPLES. OBTAINED RESULTS FROM A SINGLE REALIZATION ARE ALSO VISUALIZED IN FIG. 10A.

Invariance to object scale; number of samples used for model learning and selection: 20

Class	SVM		SVM-M		VSVM		VSVM-SL	
	F_1	ACC	F_1	ACC	F_1	ACC	F_1	ACC
Built-up area	90.09 (± 4.90)	90.43 (± 7.07)	89.93 (± 5.05)	89.75 (± 7.54)	**91.01** (± 4.80)	91.70 (± 7.48)	89.38 (± 8.14)	**96.28** (± 3.86)
Other	94.69 (± 2.85)	94.91 (± 3.05)	94.55 (± 2.85)	**95.59** (± 4.11)	95.19 (± 2.79)	95.56 (± 3.51)	**95.23** (± 2.67)	92.81 (± 5.52)
κ	84.81 (± 7.67)		84.56 (± 7.61)		**86.27** (± 7.34)		84.79 (± 10.29)	
$\overline{F_1}$	93.14 (± 3.52)		92.99 (± 3.52)		**93.78** (± 3.41)		93.26 (± 4.50)	
OA	93.10 (± 3.57)		92.96 (± 3.52)		**93.76** (± 3.42)		93.45 (± 4.03)	
AA	92.67 (± 4.00)		92.67 (± 3.37)		93.63 (± 3.46)		**94.54** (± 2.40)	

Invariance to object shape; number of samples used for model learning and selection: 34

Class	SVM		SVM-M		VSVM		VSVM-SL	
	F_1	ACC	F_1	ACC	F_1	ACC	F_1	ACC
Built-up area	91.98 (± 3.17)	89.94 (± 5.41)	93.12 (± 2.10)	92.21 (± 4.22)	91.59 (± 4.62)	90.11 (± 7.42)	**93.39** (± 3.97)	**95.49** (± 5.04)
Other	95.53 (± 1.92)	**96.97** (± 1.77)	96.25 (± 1.27)	**96.97** (± 1.77)	95.30 (± 2.79)	96.62 (± 2.62)	**96.60** (± 2.20)	95.67 (± 2.39)
κ	87.53 (± 5.05)		89.38 (± 3.34)		86.93 (± 7.34)		**90.00** (± 6.15)	
$\overline{F_1}$	94.33 (± 2.33)		95.19 (± 1.54)		94.05 (± 3.39)		**95.52** (± 2.80)	
OA	94.27 (± 2.38)		95.15 (± 1.58)		93.98 (± 3.46)		**95.51** (± 2.83)	
AA	93.46 (± 2.81)		94.59 (± 1.89)		93.36 (± 3.93)		**95.58** (± 3.12)	

TABLE VI

STUDY AREA 2 (HAGADERA): CLASSIFICATION ACCURACIES OBTAINED WITH THE DIFFERENT METHODS FOR THE MULTICLASS CLASSIFICATION SETTING REPORTED AS MEAN AND STANDARD DEVIATION (IN BRACKETS) FROM TWENTY REALIZATIONS WITH A VARYING CONFIGURATION OF LABELED SAMPLES. OBTAINED RESULTS FROM A SINGLE REALIZATION ARE ALSO VISUALIZED IN FIG. 10B.

Invariance to object scale; number of samples used for model learning and selection: 20

Class	SVM		SVM-M		VSVM		VSVM-SL	
	F_1	ACC	F_1	ACC	F_1	ACC	F_1	ACC
Built-up area	89.08 (±4.10)	97.77 (±1.67)	90.86 (±2.30)	97.90 (±1.86)	**92.28** (±2.74)	**98.42** (±1.55)	89.70 (±4.68)	95.76 (±6.77)
Vegetation	95.98 (±2.38)	**99.60** (±0.25)	96.11 (±1.72)	99.52 (±0.44)	97.07 (±0.87)	99.44 (±0.31)	96.62 (±1.79)	99.36 (±0.51)
Bare soil	94.32 (±3.03)	93.43 (±4.99)	94.23 (±2.53)	93.95 (±3.44)	**96.12** (±2.43)	**95.46** (±2.50)	93.76 (±5.27)	94.21 (±3.65)
Fence	13.10 (±5.20)	7.28 (±3.11)	14.29 (±4.21)	8.00 (±2.62)	**19.28** (±7.24)	**11.21** (±4.77)	17.01 (±7.52)	9.90 (±5.31)
Shadow	16.35 (±6.85)	9.46 (±4.56)	18.06 (±7.88)	10.49 (±5.21)	**23.53** (±10.93)	**14.46** (±8.01)	20.20 (±7.27)	11.91 (±5.15)
κ	85.20 (±3.87)		86.38 (±2.72)		**89.29** (±3.43)		86.57 (±4.90)	
$\overline{F_1}$	92.59 (±2.18)		93.18 (±1.38)		**94.63** (±1.51)		92.84 (±3.30)	
OA	89.70 (±2.79)		90.55 (±1.98)		**92.61** (±2.47)		90.74 (±3.38)	
AA	61.51 (±1.62)		61.97 (±0.90)		**63.80** (±2.14)		62.23 (±1.85)	

Invariance to object shape; number of samples used for model learning and selection: 20

Class	SVM		SVM-M		VSVM		VSVM-SL	
	F_1	ACC	F_1	ACC	F_1	ACC	F_1	ACC
Built-up area	90.42 (±3.77)	97.25 (±1.72)	88.69 (±3.15)	96.78 (±2.25)	**93.24** (±2.61)	97.57 (±1.23)	91.64 (±3.21)	**97.66** (±1.44)
Vegetation	96.41 (±1.23)	99.45 (±0.59)	97.48 (±1.57)	99.60 (±0.35)	**97.08** (±1.06)	**99.62** (±0.27)	96.45 (±1.10)	99.52 (±0.46)
Bare soil	94.45 (±2.29)	93.70 (±3.47)	93.06 (±2.90)	93.00 (±3.93)	**96.45** (±0.96)	**96.21** (±2.30)	94.93 (±1.98)	94.55 (±3.00)
Fence	14.60 (±3.50)	8.17 (±2.11)	13.94 (±4.71)	7.86 (±2.99)	**20.37** (±7.45)	**11.96** (±5.13)	16.06 (±5.82)	9.14 (±3.63)
Shadow	18.94 (±6.96)	11.13 (±4.76)	14.04 (±4.29)	7.83 (±2.61)	**28.88** (±11.60)	**18.12** (±8.43)	22.91 (±7.85)	13.55 (±5.29)
κ	86.83 (±3.49)		85.52 (±3.59)		**90.30** (±2.84)		87.80 (±3.54)	
$\overline{F_1}$	93.22 (±1.96)		92.55 (±2.04)		**95.07** (±1.21)		93.80 (±1.67)	
OA	90.89 (±2.51)		89.96 (±2.57)		**93.34** (±2.02)		91.56 (±2.60)	
AA	61.94 (±1.45)		61.01 (±1.54)		**64.70** (±2.28)		62.88 (±1.39)	

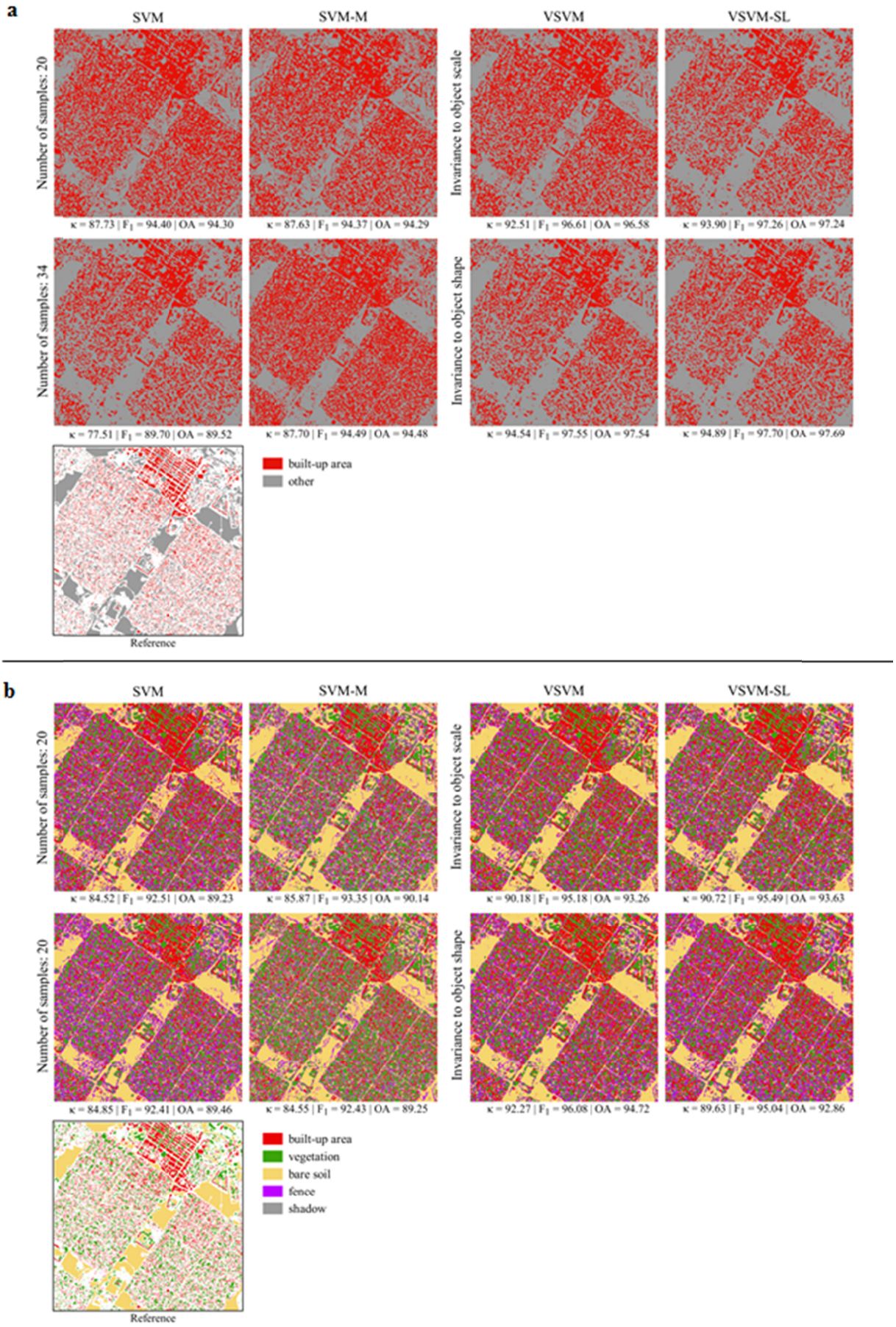


Fig. 10. Visualized results from a single realization with the different methods for the binary (a) and multiclass (b) classification setting for the Hagadera data set.

5 Conclusions and Outlook

In this paper, we proposed a novel learning algorithm based on SVM, which encodes additional prior knowledge based on virtual samples to render the decisions functions of a classification model invariant. Thereby, we followed a self-learning strategy to eventually prune non-informative virtual samples from the training set. This procedure is intended to allow for enhanced generalization capabilities particularly in situations with very few labeled samples. The proposed technique was applied to two portions of very high spatial resolution multispectral imagery acquired by the WorldView-II sensor. Experimental results were obtained for binary and multiclass classification problems. They underline the effectiveness of the proposed methods, which allow for significantly increased classification accuracies compared to related benchmark methods and enhanced spatial consistency of corresponding classification maps. The VSVM approach particularly allowed for the best accuracies in very challenging classification problems, and the constrained VSVM-SL approach was particularly beneficial in very small sample settings while obtaining simultaneously sparse model solutions.

In future works we aim to learn VSVM on a semi-supervised inference scheme to possibly further increase classification accuracies in settings with very few labeled samples. In this context, the deployed self-learning strategy can be the basis to include only informative unlabeled samples in the model in an efficient way. Moreover, we aim to adapt the proposed method within an adequate processing framework for classification of hyperspectral data.

ACKNOWLEDGEMENT

We want to acknowledge the support of the German Federal Ministry for Economic Affairs and Energy's initiative "Smart Data innovations from data" under grant agreement: "smart data for catastrophe management (sd-kama, 01MD15008B)". The work of Christian Geiß was supported by the Helmholtz Association under the grant "pre_DICT" (PD-305). We thank the editor and reviewers for their valuable comments on the initial manuscript and Michael Seibert (studio unfun) for very fruitful discussions on "Pinocchio – A linear program".

REFERENCES

Aravena Pelizari, P., Spröhnle, K., Geiß, C., Schoepfer, E., Plank, S., Taubenböck, H., 2018. Multi-sensor feature fusion for very high spatial resolution built-up area extraction in temporary settlements. in press, Remote Sens. Environ. 209, 793-807.

Audebert, N., Le Saux, B., Lefevre, S., 2018. Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks. ISPRS J. Photogramm. Remote Sens. 140, 20-32.

Baatz, M., Schäpe, A., 2000. Multiresolution Segmentation—An Optimization Approach for High Quality Multi-Scale Image Segmentation, ser. Angewandte Geographische Informations-Verarbeitung XII, Strobl, J., Blaschke, T., Griesebner, G., Eds. Karlsruhe, Germany: Herbert Wichmann Verlag, 12-23.

Blaschke, T., 2010. Object based image analysis for remote sensing. ISPRS J. Photogramm. Remote Sens. 65, 2-16.

Breiman, L., 2001. Random forests. Mach. Learn. 45, 5-32.

Bruzzone, L., Carlin, L., 2006. A multilevel context-based system for classification of very high spatial resolution images. IEEE Trans. Geosci. Remote Sens. 44, no. 9, 2587-2600.

Bruzzone, L., Chi, M., Marconcini, M., 2006. A novel transductive SVM for semisupervised classification of remote-sensing images. IEEE Trans. Geosci. Remote Sens. 44, no. 11, 3363-3373.

Burges, C. J. C., 1998. A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery. 2, 121-167.

Camps-Valls, G., Bruzzone, L., 2009. Kernel Methods for Remote Sensing Data Analysis. John Wiley & Sons, New York.

Camps-Valls, G., Tuia, D., Bruzzone, L., Benediktsson, J. A., 2014. Advances in Hyperspectral Image Classification: Earth monitoring with statistical learning methods. IEEE Signal Processing Magazine 31, no.1, 45-54.

Cortes, C., Vapnik, V., 1995. Support vector networks. Mach. Learn. 20, 273-297.

Decoste, D., Schölkopf, B., 2002. Training Invariant Support Vector Machines. Mach. Learn. 46, 161-190.

Demir, B., Persello, C., Bruzzone, L., 2011. Batch-Mode Active-Learning Methods for the Interactive Classification of Remote Sensing Images. IEEE Trans. Geosci. Remote Sens. 49, no. 3, 1014-1031.

Dópido, I., Li, J., Marpu, P. R., Plaza, A., Bioucas Dias, J. M., Benediktsson, J.A., 2013. Semisupervised Self-Learning for Hyperspectral Image Classification. IEEE Trans. Geosci. Remote Sens. 51, no. 7, 4032-4044.

Duda, R. O., Hart, P. E., Stork, D. G., 2001. Pattern Classification, 2nd ed., John Wiley & Sons, New York.

Fernández-Delgado, M., Cernadas, E., Barro, S., Amorim, D., 2014. Do we Need Hundreds of Classifiers to Solve Real World Classification Problems? Journal of Machine Learning Research. 15, 3133-3181.

Foody, G. M., 2009. On Training and Evaluation of SVM for Remote Sensing Applications, Camps-Valls, G., Bruzzone, L. (Eds.), Kernel Methods for Remote Sensing Data Analysis. John Wiley & Sons, Ltd, Chichester, UK, 85-109.

Geiß, C., Taubenböck, H., 2015. Object-based postclassification relearning. IEEE Geosci. Remote Sens. Lett. 12, no. 11, 2336-2340.

Geiß, C., Jilge, M., Lakes, T., Taubenböck, H., 2016a. Estimation of Seismic Vulnerability Levels of Urban Structures with Multisensor Remote Sensing. IEEE Journal of Selected Topics in Applied Earth Observation and Remote Sensing 9, no. 5, 1913-1936.

Geiß, C., Klotz, M., Schmitt, A., Taubenböck, H., 2016b. Object-based Morphological Profiles for Classification of Remote Sensing Imagery. IEEE Trans. Geosci. Remote Sens. 54, no. 10, 5952-5963.

Geiß, C., Aravena Pelizari, P., Schrade, H., Brenning, A., Taubenböck, H., 2017a. On the Effect of Spatially Non-disjoint Training and Test Samples on Estimated Model Generalization Capabilities in Supervised Classification with Spatial Features. IEEE Geosci. Remote Sens. Lett. 14, no. 11, 2008-2012.

Geiß, C., Schauß, A., Riedlinger, T., Dech, S., Zelaya, C., Guzman, N., Hube, M., Arsanjani, J. J., Taubenböck, H., 2017b. Joint use of remote sensing data and volunteered geographic information for exposure estimation – evidence from Valparaíso, Chile. Natural Hazards 86, 81-105.

Geiß, C., Thoma, M., Pittore, M., Wieland, M., Dech, S., Taubenböck, H., 2017c. Multitask Active Learning for Characterization of Built Environments with Multisensor Earth Observation Data. IEEE Journal of Selected Topics in Applied Earth Observation and Remote Sensing 10, no. 12, 5583-5597.

Izquierdo-Verdiguier, E., Laparra, V., Gómez-Chova, L., Camps-Valls, G., 2013. Encoding Invariances in Remote Sensing Image Classification with SVM. IEEE Geosci. Remote Sens. Lett. 10, no. 5, 981-985.

Haralick, R., Shanmugam, K., Dinstein, I., 1973. Textural features for image classification. IEEE Transactions on Systems, Man, and Cybernetics, SMC-3: 610-621.

Leinenkugel, P., Esch, T., Künzer, C., 2011. Settlement detection and impervious surface estimation in the Mekong Delta using optical and SAR remote sensing data. Remote Sens. Environ. 115, 3007-3019.

Li, Y.-F., Zhou, Z.-H., 2015. Towards Making Unlabeled Data Never Hurt. IEEE Transactions on Pattern Analysis and Machine Intelligence 37, no. 1.

Lu, X., Zhang, J., Li, T., Zhang, Y., 2016. A Novel Synergetic Classification Approach for Hyperspectral and Panchromatic Images Based on Self-Learning. IEEE Trans. Geosci. Remote Sens. 54, no. 8, 4917-4928.

Martinis, S., Twele, A., Voigt, S., 2011. Unsupervised Extraction of Flood-Induced Backscatter Changes in SAR Data Using Markov Image Modeling on Irregular Graphs. IEEE Trans. Geosci. Remote Sens. 49, no. 8, 4917-4928.

Melgani, F., Bruzzone, L., 2004. Classification of hyperspectral remote sensing images with support vector machines. IEEE Trans. Geosci. Remote Sens. 42, no. 8, 1778-1790.

Nogueira, K., Penatti, O. A., & dos Santos, J. A. (2017). Towards better exploiting convolutional neural networks for remote sensing scene classification. Pattern Recognition, 61, 539-556.

Pacifici, F., Chini, M., Emery, W., 2009. A neural network approach using multi-scale textural metrics from very high resolution panchromatic imagery for urban land-use classification. Remote Sens. Environ. 113, no. 6, 1276-1292.

Schölkopf, B., Smola, A., 2002. Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. Cambridge, MA, USA: MIT Press.

Stumpf, A., Kerle, N., 2011. Object-oriented mapping of landslides using random forests. Remote Sens. Environ. 115, no. 10, 2564-2577.

Sun, Z., Fang, H., Deng, M., Chen, A., Yue, P., Di, L., 2015. Regular shape similarity index: a novel index for accurate extraction of regular objects from remote sensing images. IEEE Trans Geosci Remote Sens. 53, no. 7, 3737-3748.

Taubenböck, H., Esch, T., Wurm, M., Roth, A., Dech, S., 2010. Object-based feature extraction using high spatial resolution satellite data of urban areas. J Spatial Sci. 55, no. 1, 117-133.

Tuia, D., Ratle, F., Pacifici, F., Kanevski, M.F., Emery, W.J., 2009. Active Learning Methods for Remote Sensing Image Classification. IEEE Trans. Geosci. Remote Sens. 47, no. 7, 2218-2232.

Tuia, D., Copa, L., Kanevski, M., Munoz-Mari, J., 2011. A Survey of Active Learning Algorithms for Supervised Remote Sensing Image Classification. IEEE Journal of Selected Topics in Signal Processing. 5, no. 3, 606 - 617.

Trimble, 2014. eCognition developer 9.0 reference book. München: Germany Trimble Documentation.

UNHCR, Hagadera Refugee Camp Overview as of January 2012. Geographic Information Systems and Mapping Unit, UNHCR Regional Support Hub, Nairobi, 2012. Available online: https://data2.unhcr.org/en/documents/download/31535

Volpi, M., Tuia, D., Bovolo, F., Kanevski, M., Bruzzone, L., 2013. Supervised change detection in VHR images using contextual information and support vector machines. Int. J. Appl. Earth Observ. Geoinf. 20, 77-85.

Wang, J., Perez, L., 2017. The Effectiveness of Data Augmentation in Image Classification using Deep Learning. arXiv preprint arXiv:1712.04621.

Wolpert, H., 1996. The lack of a priori distinctions between learning algorithms,. Neural Computation 9, 1341-1390.

Yu, X., Wu, X., Luo, C., & Ren, P. (2017). Deep learning in remote sensing scene classification: a data augmentation enhanced convolutional neural network framework. GIScience & Remote Sensing, 54(5), 741-758.