Title: Batch Mode Active Learning Methods for the Interactive Classification of Remote Sensing Images

This paper appears in: IEEE Transactions on Geoscience and Remote Sensing

Date of Publication: 2011

Author(s): Begüm Demir, Claudio Persello and Lorenzo Bruzzone

Volume: 49 , Issue: 3

Page(s): 1014 - 1031

DOI: 10.1109/TGRS.2010.2072929

# Batch Mode Active Learning Methods for the Interactive Classification of Remote Sensing Images

Begüm DEMIR[1], *Student Member IEEE,* Claudio PERSELLO[2], *Student Member IEEE, and* Lorenzo BRUZZONE[2], *Senior Member IEEE,*

[1]Electronic and Telecomm. Eng. Dept., University of Kocaeli,
Umuttepe Campus, 41380 Kocaeli, Turkey
begum.demir@kocaeli.edu.tr.

[2]Dept. of Information Engineering and Computer Science, University of Trento,
Via Sommarive, 14, I-38123 Trento, Italy;
e-mail: claudio.persello@disi.unitn.it, lorenzo.bruzzone@ing.unitn.it.

***Abstract*—** This paper investigates different batch mode active learning techniques for the classification of remote sensing (RS) images with support vector machines (SVMs). This is done by generalizing to multiclass problems techniques defined for binary classifiers. The investigated techniques exploit different query functions, which are based on the evaluation of two criteria: uncertainty and diversity. The uncertainty criterion is associated to the confidence of the supervised algorithm in correctly classifying the considered sample, while the diversity criterion aims at selecting a set of unlabeled samples that are as more diverse (distant one another) as possible, thus reducing the redundancy among the selected samples. The combination of the two criteria results in the selection of the potentially most informative set of samples at each iteration of the active learning process. Moreover, we propose a novel query function that is based on a kernel clustering technique for assessing the diversity of samples and a new strategy for selecting the most informative representative sample from each cluster. The investigated and proposed techniques are theoretically and experimentally compared with state-of-the-art methods adopted

for RS applications. This is accomplished by considering VHR multispectral and hyperspectral images. By this comparison we observed that the proposed method resulted in better accuracy with respect to other investigated and state-of-the art methods on ~~both~~ the considered data sets. Furthermore, we derived some guidelines on the design of active learning systems for the classification of different types of RS images.

***Index Terms*** **– Active learning, query functions, image classification, hyperspectral images, very high spatial resolution images, support vector machines, remote sensing.**

## I. INTRODUCTION

Land cover classification from RS images is generally performed by using supervised classification techniques, which require the availability of labeled samples for training the classification algorithm. The amount and the quality of the available training samples are crucial for obtaining accurate classification maps. However, the collection of labeled samples is time consuming and costly, and the available training samples are often not enough for an adequate learning of the classifier. A possible approach to address this problem is to exploit unlabeled samples in the learning of the classification algorithm according to semisupervised or transductive classification procedure. The semisupervised approach has been widely investigated in the recent years in the RS community [1]-[5]. A different approach to both enrich the information given as input to the supervised classifier and improve the statistic of the classes is to iteratively expand the original training set according to a process that requires an interaction between the user and the automatic recognition system. This approach is known in the machine learning community as active learning (AL) and, although marginally considered in the RS community, can result very useful for different applications. The AL process is conducted according to an iterative process. At each iteration, the most informative unlabeled samples are chosen for a manual labeling and the supervised algorithm is retrained with the additional labeled samples. In this way, the unnecessary and redundant labeling of non informative samples is avoided, greatly reducing the labeling cost and time. Moreover, AL allows one to reduce the computational complexity of the training phase. In this paper we focus our attention on AL methods.

In RS classification problems, the collection of labeled samples for the initial training set and the labeling of queried samples can be derived according to: 1) in situ ground surveys, which are

associate to high cost and required time; 2) the expert interpretation of color composites (image photointerpretation), which is cheaper and faster; or 3) hybrid solutions where both photointerpretation and ground surveys are used. The choice of the labeling strategy depends on the considered problem and image type. For example, we can reasonably assume that for the classification of very high resolution (VHR) images, the labeling of samples can be easily carried out by photointerpretation. The metric or sub-metric resolution of these images allows a human expert to identify and label the objects on the ground on the basis of the inspection of their geometric and spectral properties in real or false color compositions. In case of medium (or low) resolution multispectral images and hyperspectral data are considered, the land-cover classes are characterized only on the basis of their spectral signatures (the geometric properties of the objects are not visible and cannot be used by the photointerpreter, and usually cannot be recognized with high reliability by a human expert. For example, hyperspectral data, thanks to a dense sampling of the spectral signature, allows one characterizing several different land-cover classes (e.g., associated to different arboreal species) that cannot be recognized by a visual analysis of different false color compositions. Thus, in these cases ground survey are necessary for the labeling of samples.

On the basis of the aforementioned considerations, depending on both the type of classification problem and the type of data, the cost and time associated to the labeling process significantly changes. These different scenarios require the definition of different AL schemes: we expect that in cases where photointerpretation is possible, several iterations of the labeling step are feasible; whereas in cases where ground truth surveys are necessary, only few iterations of the AL process are doable, because of both high cost and required time associated with in situ data collection.

Most of the previous studies in AL have focused on selecting the single most informative unlabeled sample to include in the training set at each iteration, by assessing its uncertainty [6]-[12]. This can be inefficient, since the classifier has to be retrained for each new labeled sample added to the training set. This approach can be inappropriate for RS image classification tasks for the abovementioned reasons. Thus, in this paper we focus on batch mode active learning, where a batch of $h > 1$ unlabeled samples is queried at each iteration. The problem with such an approach is that by selecting the samples of the batch on the basis of the uncertainty only, some of the selected samples could be similar to each other, and thus do not provide additional information for the model updating with respect to other samples in the batch. The key issue of batch mode AL is to select sets of samples with little redundancy, so that they can provide the highest possible information to the classifier. Thus, the query function adopted for selecting the batch of the most

informative samples should take into account two main criteria: 1) uncertainty, and 2) diversity of samples [13]-[15]. The uncertainty criterion is associated to the confidence of the supervised algorithm in correctly classifying the considered sample, while the diversity criterion aims at selecting a set of unlabeled samples that are as more diverse (distant one another) as possible, thus reducing the redundancy among the selected samples. The combination of the two criteria results in the selection of the potentially most informative set of samples at each iteration of the AL process.

The aim of this paper is to investigate different AL techniques proposed in the machine learning literature and to properly generalize them to the classification of RS images with multiclass problem addressed by support vector machines (SVMs). The investigated techniques use different query functions with different strategies to assess the uncertainty and diversity criteria in the multiclass case. Moreover, we propose a novel query function that is based on a kernel clustering technique for assessing the diversity of samples and a new strategy for selecting the most informative representative sample from each cluster. The investigated and proposed techniques are theoretically and experimentally compared among them and with other AL algorithms proposed in the RS literature in the classification of VHR images and hyperspectral data. On the basis of this comparison some guidelines are derived on the use of AL techniques for the classification of different types of RS images.

The rest of this paper is organized as follows. Section II reviews the background on AL methods and their application to RS problems. Section III presents the investigated batch mode AL techniques and the proposed generalization to multiclass problems. Section IV presents the proposed novel query function based on kernel clustering and an original selection of cluster most informative samples. Section V presents the description of the three considered data sets that include both VHR and hyperspectral images and the design of experiments. Section VI illustrates the results obtained by the extensive experimental analysis carried out on the considered data sets. Finally, Section VII draws the conclusion of this work.

## II. Background on Active Learning

### *A. Active Learning Process*

A general active learner can be modeled as a quintuple ($G$, $Q$, $S$, $T$, $U$) [6]. $G$ is a supervised classifier, which is trained on the labeled training set $T$. $Q$ is a query function used to select the most informative unlabeled samples from a pool $U$ of unlabeled samples. $S$ is a supervisor who can assign the true class label to any unlabeled sample of $U$. The AL process is an iterative process, where the supervisor $S$ interacts with the system by iteratively labeling the most

informative samples selected by the query function *Q* at each iteration. At the initial stage, an initial training set *T* of few labeled samples is required for the first training of the classifier *G*. After initialization, the query function *Q* is used to select a set of samples *X* from the pool *U* and the supervisor *S* assigns them the true class label. Then, these new labeled samples are included into *T* and the classifier *G* is retrained using the updated training set. The closed loop of querying and retraining continues for some predefined iterations or until a stop criterion is satisfied. Algorithm 1 gives a description of a general AL process.

**Algorithm 1: Active Learning procedure**

1. Train the classifier *G* with the initial training set *T*
2. Classify the unlabeled samples of the pool *U*

**Repeat**

3. Query a set of samples (with query function *Q*) from the pool *U*
4. A label is assigned to the queried samples by the supervisor *S*
5. Add the new labeled samples to the training set *T*
6. Retrain the classifier

**Until** a stopping criteria is satisfied.

The query function *Q* is of fundamental importance in AL techniques, which often differ only in their query functions. Several methods have been proposed so far in the machine learning literature. A probabilistic approach to AL is presented in [7], which is based on the estimation of the posterior probability density function of the classes both for obtaining the classification rule and to estimate the uncertainty of unlabeled samples. In the two-class case, the query of the most uncertain samples is obtained by choosing the samples closest to 0.5 (half of them below and half above this probability value). The query function proposed in [16] is designed to minimize future errors, i.e., the method selects the unlabeled pattern that, once labeled and added to the training data, is expected to result in the lowest error on test samples. This approach is applied to two regression models (i.e., weighted regression and mixture of Gaussians) where an optimal solution for minimizing future error rates can be obtained in closed form. Unfortunately, this solution is intractable to calculate the expected error rate for most classifiers without specific statistical models. A statistical learning approach is used in [17] for regression problems with multilayer perceptron. In [18], a method is proposed that selects the next example according to an optimal criterion (which minimizes the expected error rate on future test samples), but solves the problem by using a sampling estimation. The authors in [18] present two techniques for estimating future error rate. In the first technique, the future error rate is estimated by log-loss using the entropy of the posterior class distribution on the set of unlabeled samples. In the second technique, a 0-1 loss

function using the posterior probability of the most probable class for a set of unlabeled samples is used. Instead of estimating the expected error over the full distribution, the error is measured over the samples in the pool *U*. Furthermore, the estimation of the error is obtained using the learner at the previous iteration. The query function causes the selection of the examples which maximize the sharpness of learner's existing belief over the unlabeled examples. The method is implemented using naïve Bayes.

Another popular paradigm is given by committee-based active learners. The "query by committee" approach [19]-[21] is a general AL algorithm that has theoretical proofed guarantees on the reduction in prediction error with the number of queries. A committee of classifiers using different hypothesis about parameters is trained to label a set of unknown examples. The algorithm selects the samples where the disagreement between the classifiers is maximal. In [22], two query methods are proposed that combine the idea of query by committee and that of boosting and bagging.

An interesting category of AL approaches, which have gained significant success in numerous real-world learning tasks, is based on the use of support vector machines (SVMs) [8]-[14]. The SVM classifier [23]-[24] is particularly suited to AL due to its intrinsic high generalization capabilities and because its classification rule can be characterized by a small set of support vectors that can be easily updated over successive learning iterations [12]. One of the most popular (and effective) query heuristic for active SVM learning is margin sampling (MS), which selects the data point closest to the current separating hyperplane. This method results in the selection of the unlabeled sample with the lowest confidence, i.e., the maximal uncertainty on the true information class. The query strategy proposed in [10] is based on the splitting of the version space [10],[13]: the points which split the current version space into two halves having equal volumes are selected at each step, as they are likely to be the actual support vectors. Three heuristics for approximating the above criterion are described; the simplest among them selects the point closest to the hyperplane as in [8]. In [6], an approach is proposed that estimates the uncertainty level of each sample according to the output score of a classifier and selects only those samples whose outputs are within the uncertainty range. In [11], the authors present possible generalizations of the active SVM approach to multiclass problems.

It is important to observe that the abovementioned methods consider only the uncertainty of samples, which is an optimal criterion only for the selection of one sample at each iteration. Selecting a batch of $h > 1$ samples exclusively on the basis of the uncertainty (e.g., the distance to the classification hyperplane) may result in the selection of similar (redundant) samples that do not provide additional information. However, in many problems it is necessary to speed up the

learning process by selecting batches of more than one sample at each iteration. In order to address this shortcoming, in [13] an approach is presented especially designed to construct batches of samples by incorporating a diversity measure that considers the angles between the induced classification hyperplanes (more details on this approach are given in the next section). Another approach to consider the diversity in the query function is the use of clustering [14]-[15]. In [14], an AL heuristic is presented, which explores the clustering structure of samples and identifies uncertain samples avoiding redundancy (details of this approach are given in the next Section). In [25]-[26], the authors present a framework for batch mode AL that applies the Fisher information matrix to select a number of informative examples simultaneously.

Nevertheless, most of the abovementioned approaches are designed for binary classification and thus are not suitable for most of the RS classification problems. In this paper, we focus on multiclass SVM-based AL approaches that can select a batch of samples at each iteration for the classification of RS images. The next subsection provides a discussion and a review on the use of AL for the classification of RS images.

*B. Active learning for the classification of RS data*

Active learning has been applied mainly to text categorization and image retrieval problems. However, the AL approach can be adopted for the interactive classification of RS images by taking into account the specific features of this domain. In RS problems, the supervisor $S$ is a human expert that can derive the land-cover type of the area on the ground associated to the selected patterns according to the three possible strategies identified in the introduction, i.e., photointerpretation, ground survey, or hybrid strategies. Here, these different strategies are associated with significantly different costs and times, and the choice of the strategy (and thus the costs and times) depends on the considered classification problem. The image photointerpretation is relatively cheep but it strongly depends on expert's ability to reliably identify the correct label of selected samples. The cost of ground surveys is normally much higher and depends on the considered area. According to these strategies, the AL approach can be run as 1) interactive expert-guided classification tool or 2) in-situ ground surveys planning and supervised classification tool. In [27], the AL problem is formulated considering a spatially dependent label acquisition costs. With the present work we observe that the labeling cost mainly depends on the type of the RS data, which affects the aforementioned labeling strategy. For example, in case of multispectral VHR images, often the labeling of samples can be carried out by photointerpretation, while in the case of medium/low resolution multispectral images and hyperspectral data, expensive ground surveys are necessary. No particular restrictions are usually considered for the definition of the

initial training set $T$ and his size $|T|$, since we expect that the AL process can be started up with few samples for each class without affecting the convergence capability (the initial samples can affect the number of iterations necessary for obtaining convergence). The pool of unlabeled samples $U$ can be associated to the whole considered image or to a portion of it (for reducing the computational time associated to the query function and/or for considering only the areas of the scene accessible for labeling). An important issue is related to the capability of the query function to select batches of $h > 1$ samples, which results to be of fundamental importance for the adoption of AL in real-world RS problems. It is worth to note here the importance of the choice of the $h$ value in the design of the AL classification system, as it affects the number of iterations and thus both the performance and the cost of the classification system. In general, we expect that for the classification of VHR images (where photointerpretation is possible), several iterations of the labeling step may be carried out and small values for $h$ can be adopted; whereas in cases where ground truth surveys are necessary, only few iterations of the AL process are possible and large $h$ values are necessary.

In the RS domain, AL was applied to the detection of subsurface targets, such as landmines and unexploded ordnance in [29]-[30]. In [30], an efficient AL procedure is developed, based on a mutual information measure. In this procedure, one initially performs excavation with the purpose of acquiring labels to improve the classifier, and once this AL phase is completed, the resulting classifier is applied to the remaining unlabeled signatures to quantify the probability that each item is an unexploded ordnance. Some preliminary works about the use of AL for RS classification problems can be found in [12], [31]-[32]. The technique proposed in [12] is based on MS and selects the most uncertain sample for each binary SVM in a One-Against-All (OAA) multiclass architecture (i.e., querying $h = n$ samples, where $n$ is the number of classes). In [31], two batch mode AL techniques for multiclass RS classification problems are proposed. The first technique is MS by closest support vector (MS-cSV), which considers the smallest distance of the unlabeled samples to the $n$ hyperplanes (associated to the $n$ binary SVMs in a (OAA) multiclass architecture) as the uncertainty value. At each iteration, the most uncertain unlabeled samples, which do not share the closest SV, are added to the training set. The second technique, called entropy query-by bagging (EQB), is a classifier independent approach based on the selection of unlabeled samples according to the maximum disagreement between a committee of classifiers. The committee is obtained by bagging: first different training sets (associated to different EQB predictors) are drawn with replacement from the original training data. In [31], each training set is used to train the OAA SVM architecture to predict the different labels for each unlabeled sample. Finally, the entropy of

the distribution of the different labels associated to each sample is calculated to evaluate the disagreement among the classifiers on the unlabeled samples. The samples with maximum entropy (i.e., those with maximum disagreement among the classifiers) are added to the current training set. In [32], an AL technique is presented, which selects the unlabeled sample that maximizes the information gain between the a posteriori probability distribution estimated from the current training set and the training set obtained by including that sample into it. The information gain is measured by the Kullback–Leibler (KL) divergence. This KL-Maximization (KL-Max) technique can be implemented with any classifier that can estimate the posterior class probabilities. However this technique can be used to select only one sample at each iteration.

## III. INVESTIGATED QUERY FUNCTIONS

In this section we investigate different query functions *Q* based on SVM for multiclass RS classification problems. SVM is a binary classifier, which goal is to divide the *d*-dimensional feature space into two subspaces (one for each class) using a separating hyperplane. Let us assume that a training set *T* made up of *N* pairs $\left(\mathbf{x}_i, y_i\right)_{i=1}^{N}$ is available, where $\mathbf{x}_i$ are the training samples and $y_i \in \{+1;-1\}$ are the associated labels. The decision rule used to find the membership of an unknown sample is based on the sign of the discrimination function $f(\mathbf{x}) = \langle \mathbf{w} \cdot \mathbf{x} \rangle + b$ associated to the hyperplane. An important property of SVMs is related to the possibility to project the original data into a higher dimensional feature space via a kernel operator $K(\cdot,\cdot)$, which satisfies the Mercer's conditions [28]. The training phase of the classifier can be formulated as a minimization problem by using the Lagrange optimization theory, which lead to the calculation of the values of Lagrange multipliers $\alpha_i$ associated with the original training patterns $\mathbf{x}_i \in \mathcal{X}$. After the training, the discrimination function is given by

$$f(\mathbf{x}) = \sum_{i \in SV} y_i \alpha_i K(\mathbf{x}_i \cdot \mathbf{x}) + b \tag{1}$$

where SV is the set of support vectors, i.e., the training samples associated to $\alpha_i > 0$. In order to address multiclass problems on the basis of binary classifiers, the general approach consists of defining an ensemble of binary classifiers and combining them according to some decision rules [24]. The design of the ensemble of binary classifiers involves the definition of a set of two-class problems, each modeled with two groups of classes. The selection of these subsets depends on the kind of approach adopted to combine the ensemble, e.g., according to *One-Against-All* (OAA) or *One-Against-One* (OAO) strategies [24]. In this work we adopt the OAA strategy, which involves

a parallel architecture made up of *n* SVMs, one for each information class. Each SVM solves a two-class problem defined by one information class against all the others. We refer the reader to [24] for greater details on SVM in RS.

The investigated AL techniques are based on standard methods; however, some of them are presented here with modifications with respect to the original version to overcome shortcomings that would affect their applicability to real RS problems. In particular, the presented techniques are adapted to classification problems characterized by a number of classes $n > 2$ (using a OAA multiclass strategy) and to the inclusion of a batch of $h > 1$ samples at each iteration in the training set (thus reducing the number of iterations required by the AL process to reach the desired accuracy, which is very important, taking into account the costs and the times associated to the labeling process in RS classification problems). The investigated query functions are based on the evaluation of the uncertainty and diversity criteria applied in two consecutive steps. The $m > h$ most uncertain samples are selected in the uncertainty step and the most diverse $h$ ($m > h > 1$) samples among these $m$ uncertain samples are chosen in the diversity step. The ratio $m / h$ provides an indication on the tradeoff between uncertainty and diversity. In this section we present different possible implementations for both steps, focusing on the OAA multiclass architecture.

*A. Techniques for Implementing the Uncertainty Criterion with Multiclass SVMs*

The uncertainty criterion aims at selecting the unlabeled samples that have maximum uncertainty about their correct label among all samples in the unlabeled sample pool *U*. Since the most uncertain samples have the lowest probability to be correctly classified by the current classification model, they are the most useful to be included in the training set. In this paper, we investigate two possible techniques in the framework of multiclass SVM: a) binary-level uncertainty (which evaluates uncertainty at the level of binary SVM classifiers), and b) multiclass-level uncertainty (which analyzes uncertainty within the considered OAA architecture).

*Binary-Level Uncertainty (BLU)*

The binary-level uncertainty (BLU) technique separately selects a batch of the most uncertain unlabeled samples from each binary SVM on the basis of the MS query function. In the technique adopted in [12], at each iteration only the (single) sample from *U* closest to the hyperplane of each binary SVM was added to the training set (i.e., $h = n$). In the presented BLU technique, at each iteration the most uncertain $q$ ($q > 1$) samples are selected from each binary SVM (instead of a single sample). In greater detail, *n* binary SVMs are initially trained with the

current training set and the functional distance $f_i(\mathbf{x})$, $i=1,...,n$ [given by (1)] of each unlabeled sample $\mathbf{x}\in U$ to the hyperplane is obtained. Then, the set of $q$ samples $\left\{\mathbf{x}_{1,i}^{BLU},\mathbf{x}_{2,i}^{BLU},...,\mathbf{x}_{q,i}^{BLU}\right\}$, $i=1,2,...,n$ closest to the corresponding hyperplane are selected for each binary SVM, where $\mathbf{x}_{j,i}^{BLU}$, $j=1,2,...,q$, represents the selected $j$-th sample from the $i$-th SVM. Totally $\rho=qn$ samples are taken. Since some unlabeled samples can be selected by more than one binary SVM, the redundant samples are removed. Thus, the total number $m$ of selected samples can actually be smaller than $\rho$ (i.e., $m\le\rho$). The set of $m$ most uncertain samples $\{\mathbf{x}_1^{BLU},\mathbf{x}_2^{BLU},...,\mathbf{x}_m^{BLU}\}$ is forwarded to the diversity step. Fig. 1 shows the architecture of the investigated BLU technique.

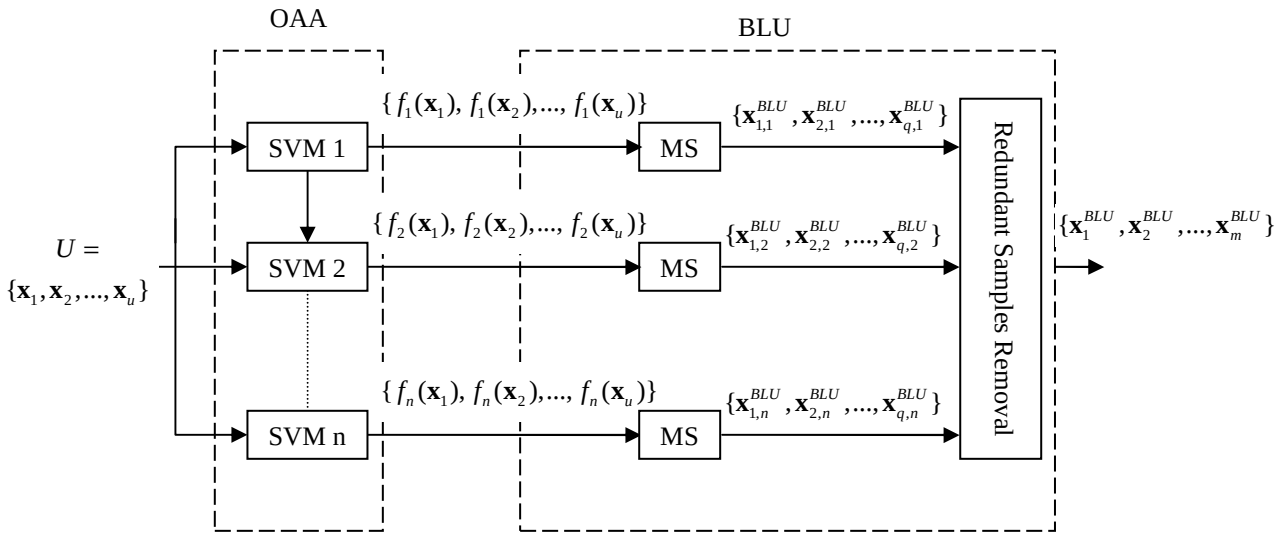



Fig. 1. Multiclass architecture adopted for the BLU technique

*Multiclass-Level Uncertainty (MCLU)*

The adopted multiclass-level uncertainty (MCLU) technique selects the most uncertain samples according to a confidence value $c(\mathbf{x})$, $\mathbf{x}\in U$, which is defined on the basis of their functional distance $f_i(\mathbf{x})$, $i=1,...,n$ to the $n$ decision boundaries of the binary SVM classifiers included in the OAA architecture [31], [33]. In this technique, the distance of each sample $\mathbf{x}\in U$ to each hyperplane is calculated and a set of $n$ distance values $\left\{f_1(\mathbf{x}), f_2(\mathbf{x}),... f_n(\mathbf{x})\right\}$ is obtained. Then, the confidence value $c(\mathbf{x})$ can be calculated using different strategies. Here, we consider two strategies: 1) the minimum distance function $c_{\min}(\mathbf{x})$ strategy, which is obtained by taking the smallest distance to the hyperplanes (as absolute value), i.e., [31]

$$c_{\min}(\mathbf{x})=\min_{i=1,2,...,n}\left\{abs[\,f_i(\mathbf{x})]\right\} \tag{2}$$

and 2) the difference function $c_{diff}(\mathbf{x})$ strategy, which considers the difference between the first largest and the second largest distance values to the hyperplanes, i.e, [33]

$$\begin{aligned} r_{1\max} &= \arg\max_{i=1,2,...,n} \{ f_i(\mathbf{x}) \} \\ r_{2\max} &= \arg\max_{j=1,2,...,n,\ j \neq r_{1\max}} \{ f_j(\mathbf{x}) \} \\ c_{diff}(\mathbf{x}) &= f_{r_{1\max}}(\mathbf{x}) - f_{r_{2\max}}(\mathbf{x}) \end{aligned} \tag{3}$$

The $c_{\min}(\mathbf{x})$ function models a simple strategy that computes the confidence of a sample $\mathbf{x}$ taking into account the minimum distance to the hyperplanes evaluated on the basis of the most uncertain binary SVM classifier. Differently, the $c_{diff}(\mathbf{x})$ strategy assesses the uncertainty between the two most likely classes. If this value is high, the sample $\mathbf{x}$ is assigned to $r_{1\max}$ with high confidence. On the contrary, if $c_{diff}(\mathbf{x})$ is small, the decision for $r_{1\max}$ is not reliable and there is a possible conflict with the class $r_{2\max}$ (i.e., the sample $\mathbf{x}$ is very close to the boundary between class $r_{1\max}$ and $r_{2\max}$). Thus, this sample is considered uncertain and is selected by the query function for better modeling the decision function in the corresponding position of the feature space. Once the $c(\mathbf{x})$ value of each $\mathbf{x} \in U$ is obtained based on one of the two above-mentioned strategies, the $m$ samples $\mathbf{x}_1^{MCLU}, \mathbf{x}_2^{MCLU}, ..., \mathbf{x}_m^{MCLU}$ with lower $c(\mathbf{x})$ are selected to be forwarded to the diversity step. Note that $\mathbf{x}_j^{MCLU}$ denotes the selected $j$-th most uncertain sample based on the MCLU strategy. Fig. 2 shows the architecture of the investigated MCLU technique.

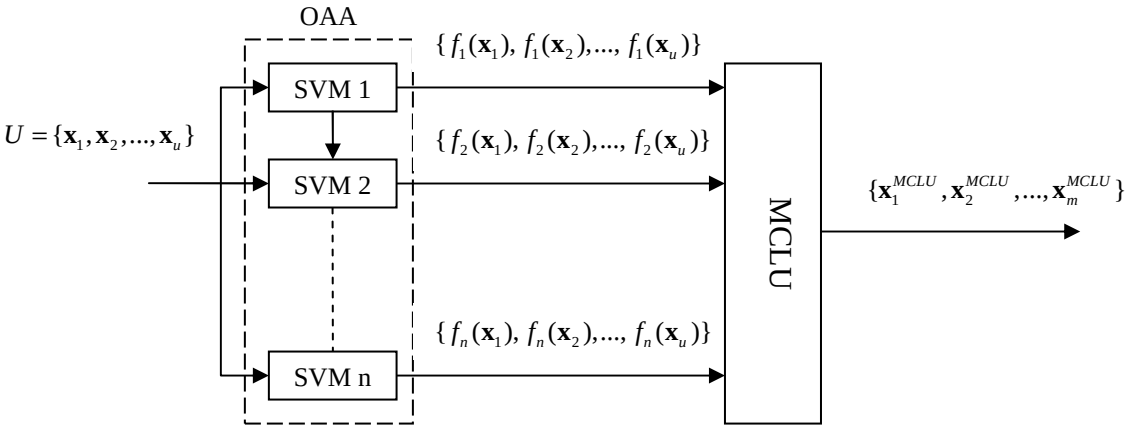



Fig. 2. Architecture adopted for the MCLU technique.

*B. Techniques for Implementing the Diversity Criterion*

The main idea of using diversity in AL is to select a batch of samples ($h>1$) which have low confidence values (i.e., the most uncertain ones), and at the same time are diverse from each other. In this paper, we consider two diversity methods: 1) the angle based diversity (ABD); and 2) the clustering based diversity (CBD). Before considering the multiclass formulation, in the following we recall their definitions for two-class problems.

*Angle Based Diversity (ABD)*

A possible way for measuring the diversity of uncertain samples is to consider the cosine angle distance. It is a similarity measure between two samples defined in the kernel space by [13]

$$\begin{aligned} \left|\cos\left(\angle(\mathbf{x}_i,\mathbf{x}_j)\right)\right| &= \frac{\left|\phi(\mathbf{x}_i)\cdot\phi(\mathbf{x}_j)\right|}{\left\|\phi(\mathbf{x}_i)\right\|\left\|\phi(\mathbf{x}_j)\right\|} = \frac{K(\mathbf{x}_i,\mathbf{x}_j)}{\sqrt{K(\mathbf{x}_i,\mathbf{x}_i)K(\mathbf{x}_j,\mathbf{x}_j)}} \\ \angle(\mathbf{x}_i,\mathbf{x}_j) &= \cos^{-1}\left(\frac{K(\mathbf{x}_i,\mathbf{x}_j)}{\sqrt{K(\mathbf{x}_i,\mathbf{x}_i)K(\mathbf{x}_j,\mathbf{x}_j)}}\right) \end{aligned} \tag{4}$$

where $\phi(\cdot)$ is a nonlinear mapping function and $K(\cdot,\cdot)$ is the kernel function. The cosine angle distance in the kernel space can be constructed using only the kernel function without considering the direct knowledge of the mapping function $\phi(\cdot)$. The angle between two samples is small (cosine of angle is high) if these samples are close to each other and vice versa.

*Clustering Based Diversity (CBD)*

Clustering techniques evaluate the distribution of the samples in a feature space and group the similar samples into the same clusters. In [14], the standard *k*-means clustering [34] was used in the diversity step of binary SVM AL technique. The aim of using clustering in the diversity step is to consider and analyze the distribution of uncertain samples. Since the samples within the same cluster are correlated and provide similar information, a representative sample is selected for each cluster. The advantage of this approach is that cluster prototypes are implicitly sparse in the feature space, i.e., distant one another. In [14], the sample that is closest to the corresponding cluster center (called medoid sample) is chosen as representative sample.

*C. Proposed combination of Uncertainty techniques and Diversity techniques generalized to Multiclass Problems*

In this paper, each uncertainty technique is combined with one of the (binary) diversity techniques presented in the previous section. In the uncertainty step, the *m* most uncertain samples

are selected using either MCLU or BLU. In the diversity step, the most diverse $h < m$ samples are chosen based on either ABD or CBD generalized to the multiclass case. Here, four possible combinations are investigated: 1) MCLU with ABD (denoted by MCLU-ABD), 2) BLU with ABD (denoted by BLU-ABD), 3) MCLU with CBD (denoted by MCLU-CBD), and 4) BLU with CBD (denoted by BLU-CBD).

*Combination of Uncertainty techniques with ABD for Multiclass SVMs (MCLU-ABD and BLU-ABD)*

In the binary AL algorithm presented in [13], the uncertainty and ABD criteria are combined based on a weighting parameter $\lambda$. On the basis of this combination, a new sample $\mathbf{x}_t$ is included in the selected batch *X* according to the following optimization problem:

$$x_t = \arg\min_{x_i \in I/X} \left\{ \lambda \left| f(\mathbf{x}_i) \right| + (1-\lambda) \left[ \max_{x_j \in X} \frac{K(\mathbf{x}_i, \mathbf{x}_j)}{\sqrt{K(\mathbf{x}_i, \mathbf{x}_i) K(\mathbf{x}_j, \mathbf{x}_j)}} \right] \right\} \tag{5}$$

where $I$ denotes the set of unlabeled examples whose distance to the classification hyperplane is less than one, i.e., $I = \{ x_i \in U : |f(x_i)| < 1 \}$, $I / X$ represents the set of unlabeled samples of *I* that are not contained in the current batch *X*, and $\lambda$ provides the tradeoff between uncertainty and diversity. The cosine angle distance between each sample in $I / X$ and the samples included in *X* is calculated and the maximum value is taken as the diversity value of the corresponding sample. Then, the sum of the uncertainty and diversity values weighted by $\lambda$ is considered to define the combined value. The unlabeled sample $\mathbf{x}_t$ that minimizes such value is included in *X*. This process is repeated until the number of samples of the set *X* ($|X|$) is equal to *h*. This technique guarantees that the selected unlabeled samples in *X* are diverse regarding to their angles to all the others in the kernel space. Since the initial size of *X* is zero, the first sample included in *X* is always the most uncertain sample of $I$ (i.e., the sample closest to the hyperplane). We generalize this technique to multiclass architectures presenting the MCLU-ABD and BLU-ABD algorithms (see Algorithms 2 and 3).

**Algorithm 2: MCLU-ABD**

**Inputs:**

$\lambda$ (weighting parameter that tune the tradeoff between uncertainty and diversity)

*m* (number of samples selected on the basis of their uncertainty)

*h* (batch size)

**Output:**

*X* (set of unlabeled samples to be included in the training set)

1. Compute $c(\mathbf{x})$ for each sample $\mathbf{x} \in U$ .

2. Select the set of *m* unlabeled samples with lower $c(\mathbf{x})$ value (most uncertain) $\{\mathbf{x}_1^{MCLU}, \mathbf{x}_2^{MCLU}, ..., \mathbf{x}_m^{MCLU}\}$.

3. Initialize *X* to the empty set.

4. Include in *X* the most uncertain sample (the one that has the lowest $c(\mathbf{x})$ value).

**Repeat**

5. Compute the combination of uncertainty and diversity with the following equation formulated for the multiclass architecture:

$$x_t = \arg\min_{i=1,...,m} \left\{ \lambda \left| c(\mathbf{x}_i) \right| + (1-\lambda) \left[ \max_{x_j \in X} \frac{K(\mathbf{x}_i, \mathbf{x}_j)}{\sqrt{K(\mathbf{x}_i, \mathbf{x}_i) K(\mathbf{x}_j, \mathbf{x}_j)}} \right] \right\} \quad (6)$$

where we consider the *m* most uncertain samples selected at the step 2 and $c(\mathbf{x})$ is calculated as explained in the MCLU subsection (with $c_{\min}(\mathbf{x})$ or $c_{diff}(\mathbf{x})$ strategy).

6. Include the unlabeled sample $\mathbf{x}_t$ in *X*.

**Until** $|X| = h$

7. The supervisor *S* adds the label to the set of samples $\{\mathbf{x}_1^{MCLU-ABD}, \mathbf{x}_2^{MCLU-ABD}, ..., \mathbf{x}_h^{MCLU-ABD}\} \in X$ and these samples are added to the current training set *T*.

It is worth noting that the main difference between (5) and (6) is that the uncertainty in (6) is evaluated considering the confidence function $c(\mathbf{x}_i)$ instead of the functional distance $f(\mathbf{x}_i)$ as in the binary case.

**Algorithm 3: BLU-ABD**

**Inputs:**

$\lambda$ (weighting parameter that tune the tradeoff between uncertainty and diversity)
*m* (number of samples selected on the basis of their uncertainty)
*h* (batch size)
*q* (number of unlabeled samples selected for each binary SVM in the BLU technique)
*n* (total class number)

**Output:**

*X* (set of unlabeled samples to be included in the training set)

1. Select the *q* most uncertain samples from each of the *n* binary SVM included in the multiclass OAA architecture (totally $\rho = qn$ samples are obtained).
2. Remove the redundant samples and consider the set of $m \le \rho$ patterns $\{\mathbf{x}_1^{BLU}, \mathbf{x}_2^{BLU}, ..., \mathbf{x}_m^{BLU}\}$.
3. Compute $c(\mathbf{x})$ for the set of $m$ samples as follows: if one sample is selected by more than one binary SVM, $c(\mathbf{x})$ is calculated as explained in the MCLU subsection (with $c_{\min}(\mathbf{x})$ or $c_{diff}(\mathbf{x})$ strategy); otherwise $c(\mathbf{x})$ is assigned to the corresponding functional distance $f(\mathbf{x})$.
4. Initialize *X* to the empty set.
5. Include in *X* the most uncertain sample (the one that has the lowest $c(\mathbf{x})$ value).

**Repeat**

6. Compute the combination of uncertainty and diversity with the equation (6).
7. Include the unlabeled sample $\mathbf{x}_t$ in *X*.

**Until** $|X| = h$

8. The supervisor *S* adds the label to the set of patterns $\{\mathbf{x}_1^{BLU-ABD}, \mathbf{x}_2^{BLU-ABD}, ..., \mathbf{x}_h^{BLU-ABD}\} \in X$ and these samples are added to the current training set.

*Combination of Uncertainty techniques with CBD for Multiclass SVMs* (MCLU-CBD and BLU-CBD)

The uncertainty and CBD were combined for binary SVM AL in [14]. The uncertain samples are identified according to the MS strategy based on their distance to the hyperplane. Then, the standard *k*-means clustering is applied in the original feature space to the unlabeled samples whose distance to the hyperplane (computed in the kernel space) is less than one (i.e., those that lie in the margin) and the $k=h$ clusters are obtained. The medoid sample of each cluster is added to *X* (i.e., $|X| = h$), labeled by the supervisor *S* and moved to the current training set. This algorithm evaluates the distribution of the uncertain samples within the margin and selects the representative of uncertain samples based on standard *k*-means clustering. We extend this technique to multiclass problems, by defining the MCLU-CBD and BLU-CBD algorithms (see Algorithms 4 and 5).

**Algorithm 4: MCLU-CBD**

**Inputs:**

*m* (number of samples selected on the basis of their uncertainty)

*h* (batch size)

**Output:**

*X* (set of unlabeled samples to be included in the training set)

1. Compute $c(\mathbf{x})$ for each sample $\mathbf{x} \in U$.

2. Select the set of *m* unlabeled samples with lowest $c(\mathbf{x})$ (with $c_{\min}(\mathbf{x})$ or $c_{diff}(\mathbf{x})$ strategy) value (most uncertain) $\{\mathbf{x}_1^{MCLU}, \mathbf{x}_2^{MCLU}, ..., \mathbf{x}_m^{MCLU}\}$.

3. Apply the *k*-means clustering (diversity criterion) to the selected *m* most uncertain samples with *k*=*h*.

4. Calculate the *h* cluster medoid samples $\{\mathbf{x}_1^{MCLU-CBD}, \mathbf{x}_2^{MCLU-CBD}, ..., \mathbf{x}_h^{MCLU-CBD}\}$, one for each cluster.

5. Initialize *X* to the empty set and include in *X* the set of *h* patterns $\{\mathbf{x}_1^{MCLU-CBD}, \mathbf{x}_2^{MCLU-CBD}, ..., \mathbf{x}_h^{MCLU-CBD}\} \in X$

6. The supervisor *S* adds the label to the set of *h* patterns $\{\mathbf{x}_1^{MCLU-CBD}, \mathbf{x}_2^{MCLU-CBD}, ..., \mathbf{x}_h^{MCLU-CBD}\} \in X$ and these samples are added to the current training set.

**Algorithm 5: BLU-CBD**

**Inputs:**

*m* (number of samples selected on the basis of their uncertainty)

*h* (batch size)

*q* (number of unlabeled samples selected for each binary SVM in the BLU technique)

*n* (total class number)

**Output:**

*X* (set of unlabeled samples to be included in the training set)

1. Select the *q* most uncertain samples from each of the *n* binary SVMs included in the multiclass OAA architecture (totally $\rho = qn$ samples are obtained).

2. Remove the redundant samples and consider the set of $m \le \rho$ patterns $\{\mathbf{x}_1^{BLU}, \mathbf{x}_2^{BLU}, ..., \mathbf{x}_m^{BLU}\}$.

3. Compute $c(\mathbf{x})$ for the set of $m$ samples as follows: if one sample is selected by more than one binary SVM, $c(\mathbf{x})$ is calculated as explained in the MCLU subsection (with $c_{\min}(\mathbf{x})$ or $c_{diff}(\mathbf{x})$ strategy); otherwise $c(\mathbf{x})$ is assigned to the corresponding functional distance $f(\mathbf{x})$.

4. Apply the *k*-means clustering (diversity criterion) to the selected *m* most uncertain samples (*k*=*h*).

5. Calculate the *h* cluster medoid samples $\{\mathbf{x}_1^{BLU-CBD}, \mathbf{x}_2^{BLU-CBD}, ..., \mathbf{x}_h^{BLU-CBD}\}$, one for each cluster.

6. Initialize *X* to the empty set and include in *X* the set of *h* patterns $\{\mathbf{x}_1^{BLU-CBD}, \mathbf{x}_2^{BLU-CBD}, ..., \mathbf{x}_h^{BLU-CBD}\} \in X$

7. The supervisor *S* adds the label to the set of *h* patterns $\{\mathbf{x}_1^{BLU-CBD}, \mathbf{x}_2^{BLU-CBD}, ..., \mathbf{x}_h^{BLU-CBD}\} \in X$ and these samples are added to the current training set.

## IV. PROPOSED NOVEL QUERY FUNCTION

Clustering is an effective way to select the most diverse samples considering the distribution of uncertain samples in the diversity step of the query function. In the previous section we generalized the CBD technique presented in [14] to the multiclass case. However, some other limitations can compromise its effectiveness: 1) the standard *k*-means clustering is applied to the original feature space and not in the kernel space where the SVM separating hyperplane operates, and 2) the medoid sample of each cluster is selected in the diversity step as the corresponding cluster representative sample (but more "informative" samples in that cluster could be selected). Indeed, because of the kernel mapping, the set of most diverse sample in the original space may not be the most diverse in the kernel space, thus not selecting the most informative samples for the classifier. In addition, the selection of the medoid of each cluster does not result in the selection of the most uncertain batch of samples.

To overcome these problems, we propose a novel query function that is based on the combination of one of the uncertainty criteria for multiclass problems presented in the previous section and a novel Enhanced CBD (ECBD) technique. In the proposed query function, MCLU is used with the difference $c_{diff}(\mathbf{x})$ strategy in the uncertainty step to select the *m* most uncertain samples. The proposed ECBD technique, unlike the standard CBD, works in the kernel space by applying the kernel *k*-means clustering [35], [36] to the *m* samples obtained in the uncertainty step to select the $h < m$ most diverse patterns. The kernel *k*-means clustering iteratively divides the *m* samples into *k*=*h* clusters ($C_1, C_2, ...C_h$) in the kernel space. At the first iteration, initial clusters $C_1, C_2, ...C_h$ are constructed assigning initial cluster labels to each sample [35]. In next iterations, a pseudo centre is chosen as the cluster center (the cluster centers in the kernel space $\phi(\mu_1), \phi(\mu_2), ...\phi(\mu_h)$ can not be expressed explicitly). Then the distance of each sample from all cluster centers in the kernel space is computed and each sample is assigned to the nearest cluster. The Euclidean distance between $\phi(\mathbf{x}_i)$ and $\phi(\mu_v)$, $v = 1, 2, ..., h$, is calculated as [35], [36]:

$$
\begin{aligned}
& D^2(\phi(\mathbf{x}_i), \phi(\mu_v)) = \left\| \phi(\mathbf{x}_i) - \phi(\mu_v) \right\|^2 \\
& = \left\| \phi(\mathbf{x}_i) - \frac{1}{|C_v|} \sum_{j=1}^{m} \delta(\phi(\mathbf{x}_j), C_v) \phi(\mathbf{x}_j) \right\|^2 \\
& = K(\mathbf{x}_i, \mathbf{x}_i) - \frac{2}{|C_v|} \sum_{j=1}^{m} \delta(\phi(\mathbf{x}_j), C_v) K(\mathbf{x}_i, \mathbf{x}_j) + \\
& \frac{1}{|C_v|^2} \sum_{j=1}^{m} \sum_{l=1}^{m} \delta(\phi(\mathbf{x}_j), C_v) \delta(\phi(\mathbf{x}_l), C_v) K(\mathbf{x}_j, \mathbf{x}_l)
\end{aligned}
\tag{7}
$$

where $\delta\left(\phi(\mathbf{x}_j), C_v\right)$ shows the indicator function. The $\delta\left(\phi(\mathbf{x}_j), C_v\right)$=1 only if $\mathbf{x}_j$ is assigned to $C_v$, otherwise $\delta\left(\phi(\mathbf{x}_j), C_v\right)$=0. The $|C_v|$ denotes the total number of samples in $C_v$ and is calculated as $|C_v| = \sum_{j=1}^{m} \delta\left(\phi(\mathbf{x}_j), C_v\right)$. As mentioned before, $\phi(\cdot)$ is a nonlinear mapping function from the original feature space to a higher dimensional space and $K(\cdot,\cdot)$ is the kernel function. The kernel *k*-means algorithm can be summarized as follows [35]:

1. The initial value of $\delta(\phi(\mathbf{x}_i), C_v)$, $i = 1, 2, ..., m$, $v = 1, 2, ..., h$, is assigned and *h* initial clusters $\{C_1, C_2, ...C_h\}$ are obtained.

2. Then $\mathbf{x}_i$ is assigned to the closest cluster.

$$\delta(\phi(\mathbf{x}_i), C_v) = \begin{cases} 1 & \text{if } D^2(\phi(\mathbf{x}_i), \phi(\mu_v)) < D^2(\phi(\mathbf{x}_i), \phi(\mu_j)) \quad \forall \mathrm{j} \neq v \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

3. The sample that is closest to $\mu_v$ (the Euclidean distance is calculated in the kernel space by equation (7)) is selected as the pseudo centre $\eta_v$ of $C_v$.

$$\eta_v = \arg\min_{\mathbf{x}_i \in C_v} D^2(\phi(\mathbf{x}_i), \phi(\mu_v)) \tag{9}$$

4. The algorithm is iterated until converge, which is achieved when samples do not change clusters anymore.

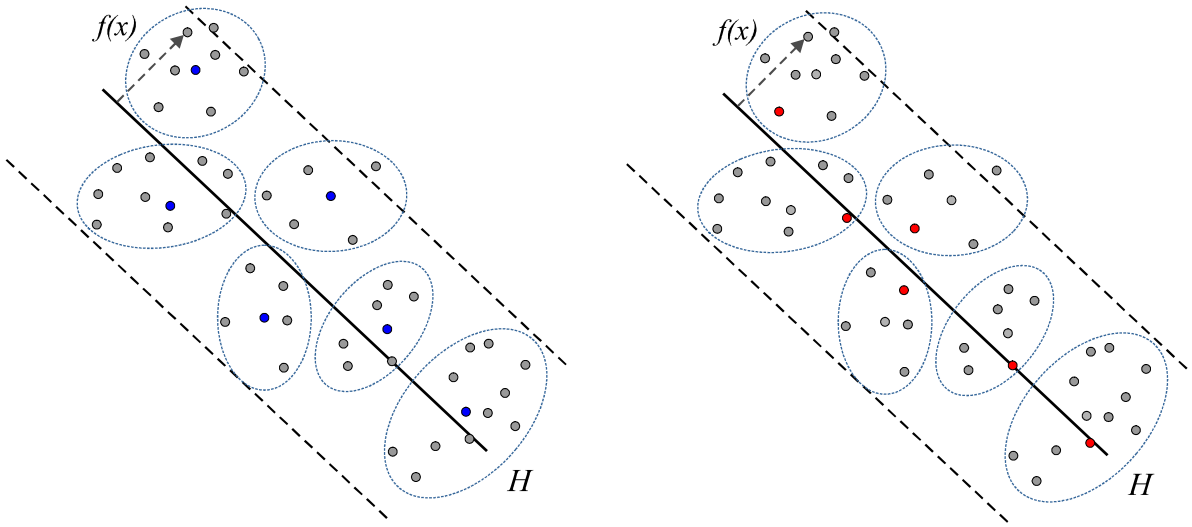
After $C_1, C_2, ...C_h$ are obtained, unlike in the standard CBD technique, the most informative (i.e., uncertain) sample is selected as the representative sample of each cluster. This sample is defined as

$$\mathbf{x}_v^{MCLU-ECBD} = \arg\min_{\phi(\mathbf{x}_i) \in C_v} \left\{ c_{diff}(\mathbf{x}_i^{MCLU}) \right\} \quad v = 1, 2, ..., h \tag{10}$$

where $\mathbf{x}_v^{MCLU-ECBD}$ represents the *v*-th sample selected using the proposed query function MCLU-ECBD and is the most uncertain sample of the *v*-th cluster (i.e., the sample that has minimum $c_{diff}(\mathbf{x})$ in the *v*-th cluster). Totally *h* samples are selected, one for each cluster, using (10).

In order to better understand the difference in the selection of the representative sample of each cluster between the query function of the CBD presented in [14] (which selects the medoid sample as cluster representative) and the proposed ECBD query function (which selects the most uncertain sample of each cluster), Fig. 3 presents a qualitative example. Note that, for simplicity, the example is presented for a binary SVM in order to visualize the confidence value $c_{diff}(\mathbf{x})$ as

the functional distance (MS is used instead of MCLU). The uncertain samples are firstly selected based on MS for both techniques, and then the diversity step is applied. The query function presented in [14] selects the medoid sample of each cluster (reported in blue in the figure), which however is not in agreement with the idea to chose the most uncertain sample in the cluster. On the contrary, the proposed query function considers the most uncertain sample of each cluster (reported in red in the figure), which in the binary example is the sample closest to the SVM hyperplane. This is a small difference with respect to the algorithmic implementation but a relevant difference from a theoretical viewpoint and for possible implications on the results.




(a) (b)

Fig. 3. Comparison between the samples selected by (a) the CBD technique presented in [14], and (b) the proposed ECBD technique.

The proposed MCLU-ECBD algorithm can be summarized as follows:

**Algorithm 6: Proposed MCLU-ECBD**

**Inputs:**

*m* (the number of samples selected on the basis of their uncertainty)

*h* (batch size)

**Output:**

*X* (set of unlabeled samples to be included in the training set)

1. Compute $c(\mathbf{x})$ for each sample $\mathbf{x} \in U$ .

2. Select the set of *m* unlabeled samples with lower $c(\mathbf{x})$ value (most uncertain) $\{\mathbf{x}_1^{MCLU}, \mathbf{x}_2^{MCLU}, ..., \mathbf{x}_m^{MCLU}\}$.

3. Apply the kernel *k*-means clustering (diversity criterion) to the selected *m* most uncertain samples with *k*=*h*.

4. Select the representative sample $\mathbf{x}_v^{MCLU-ECBD}, v = 1, 2, \ldots, h$ (i.e., the most uncertain sample) of each cluster according to (10).

5. Initialize *X* to the empty set and include in *X* the set of samples $\mathbf{x}_v^{MCLU-ECBD} \in X$ , $v = 1, 2, \ldots, h$ .

6. The supervisor *S* adds the label to the set of samples $\mathbf{x}_v^{MCLU-ECBD} \in X$ , $v = 1, 2, \ldots, h$ , and these samples are added to the current training set.

## V. DATA SET DESCRIPTION AND DESIGN OF EXPERIMENTS

### *A. Data set description*

In our experiments we used one VHR and two hyperspectral data sets. The first data set is a hyperspectral image acquired on a forest area on the Mount Bondone in the Italian Alps (near the city of Trento) on September 2007. This image consists of 1613×1048 pixels and 63 bands with a spatial resolution of 1 m. The available labeled data (4545 samples) were collected during a ground survey in summer 2007. The reader is referred to [37] for greater details on this dataset. The samples were randomly divided to derive a validation set *V* of 455 samples (which is used for model selection), a test set *TS* of 2272 samples (which is used for accuracy assessment), and a pool *U* of 1818 samples. 4% of the samples of each class are randomly chosen from *U* as initial training samples and the rest are considered as unlabeled samples. This choice was done in order to have a minimum reasonable number of samples for each class, limiting the number of the samples assumed initially available. The land cover classes and the related number of samples used in the experiments are shown in Table 1.

The second data set is a Quickbird multispectral image acquired on the city of Pavia (northern Italy) on June 23, 2002. This image includes the four pan-sharpened multispectral bands and the panchromatic channel with a spatial resolution of 0.7 m. The image size is 1024×1024 pixels. The reader is referred to [38] for greater details on this dataset. The available labeled data (6784 samples) were collected by photointerpretation. These samples were randomly divided to

derive a validation set *V* of 457 samples, a test set *TS* of 4502 samples and a pool *U* of 1825 samples. According to [38], Test pixels were collected on both homogeneous areas $TS_1$ and edge areas $TS_2$ of each class. 4% of the samples of each class in *U* are randomly selected as initial training samples, and the rest are considered as unlabeled samples. Table 2 shows the land cover classes and the related number of samples used in the experiments.

The third data set is a hyperspectral image acquired on the Kennedy Space Center (KSC), Florida, U.S., on March 23, 1996. This image consists of $512\times614$ pixels and 224 bands with a spatial resolution of 18 m. The number of bands is initially reduced to 176 by removing water absorption and low signal-to noise bands. The available labeled data (5121 samples) were collected using land cover maps derived from color infrared photography provided by KSC and Landsat Thematic Mapper (TM) imagery. The reader is referred to [40] for greater details on this dataset. After the elimination of noisy samples, the labeled samples were randomly divided to derive a validation set *V* of 513 samples, a test set *TS* of 2556 samples, and a pool *U* of 2052 samples. 4% of the samples of each class are randomly chosen from *U* as initial training samples and the rest are considered as unlabeled samples. The land cover classes and the related number of samples used in the experiments are shown in Table 3.

The first two data sets were used for all the experiments, whereas the third data set is not used for initial comparisons among investigated techniques and for the sensitivity analysis related to different parameter values (which require a huge set of experiments), but for further assessing the effectiveness of the proposed novel method and for comparing it only with the most effective investigated and literature methods.

TABLE 1. NUMBER OF SAMPLES OF EACH CLASS IN *U*, *V* AND *TS* FOR THE TRENTO DATA SET.

| **Class** | ***U*** | **V** | ***TS*** |
|---|---|---|---|
| Fagus Sylvatica | 720 | 180 | 900 |
| Larix Decidua | 172 | 43 | 215 |
| Ostrya Carpinifolia | 160 | 40 | 200 |
| Pinus Nigra | 186 | 47 | 232 |
| Pinus Sylvestris | 340 | 85 | 425 |
| Quercus Pubescens | 240 | 60 | 300 |
| Total | 1818 | 455 | 2272 |

TABLE 2. NUMBER OF SAMPLES OF EACH CLASS IN *U*, *V*, *TS1* AND *TS2* FOR THE PAVIA DATA SET.

| **Class** | ***U*** | ***V*** | $TS_1$ | $TS_2$ |
|---|---|---|---|---|
| Water | 58 | 14 | 154 | 61 |
| Tree areas | 111 | 28 | 273 | 118 |
| Grass areas | 103 | 26 | 206 | 115 |
| Roads | 316 | 79 | 402 | 211 |
| Shadow | 230 | 57 | 355 | 311 |
| Red buildings | 734 | 184 | 1040 | 580 |
| Gray buildings | 191 | 48 | 250 | 177 |
| White building | 82 | 21 | 144 | 105 |
| Total | 1825 | 457 | 2824 | 1678 |

TABLE 3. NUMBER OF SAMPLES OF EACH CLASS IN *U*, *V* AND *TS* FOR THE KSC DATA SET.

| **Class** | ***U*** | ***V*** | ***TS*** |
|---|---|---|---|
| Scrub | 305 | 76 | 380 |
| Willow swamp | 97 | 24 | 120 |
| Cabbage palm hammock | 102 | 26 | 128 |
| Cabbage palm/oak hammock | 101 | 25 | 125 |
| Slash pine | 65 | 16 | 80 |
| Oak/broadleaaf hammock | 92 | 23 | 114 |
| Hardwood swamp | 42 | 11 | 52 |
| Graminoid marsh | 173 | 43 | 215 |
| Spartina marsh | 208 | 52 | 260 |
| Cattail marsh | 151 | 38 | 188 |
| Salt marsh | 168 | 42 | 209 |
| Mud flats | 185 | 46 | 231 |
| Water | 363 | 91 | 454 |
| Total | 2052 | 513 | 2556 |

*B. Design of Experiments*

In our experiments, without loosing in generality, we adopt an SVM classifier with RBF kernel. The values for the regularization parameter *C* and the spread $\gamma$ of the RBF kernel parameters are chosen performing a grid-search model selection only at the first iteration of the AL process. Indeed, initial experiments revealed that, if a reasonable number of initial training samples is considered, performing the model selection at each iteration does not increase significantly the classification accuracies at the cost of a much higher computational burden. The MCLU step is implemented with different *m* values defined on the basis of the value of *h* (i.e., $m = 4h,\ 6h, 10h$), with *h*=5,10,40,100. In the BLU technique, the *q*=*h* most uncertain samples are selected for each binary SVM. Thus the total number of selected samples for all SVMs is $\rho = qn$. After removing repetitive patterns, $m \le \rho$ samples are obtained. The value of $\lambda$ used in the MCLU-ABD and the BLU-ABD [for computing (6)] is varied as $\lambda = 0.3,0.5,0.6,0.8$. The total cluster number *k* for both kernel *k*-means clustering and standard *k*-means clustering is fixed to *h*. All the investigated techniques and the proposed MCLU-ECBD technique are compared with the

EQB and the MS-cSV techniques presented in [31]. The results of EQB are obtained fixing the number of EQB predictors to eight and selecting bootstrap samples containing 75 % of initial training patterns. These values have been suggested in [31]. Since the MS-cSV technique selects diverse uncertain samples according to their distance to the SVs, and its literature formulation can consider at most one sample related to each SV [31], it is not possible to define $h$ greater than the total number of SVs. For this reason we can provide MS-cSV results for only $h$=5,10 for Trento and Pavia data sets, $h$=5,10,40 for KSC data set. Also the results obtained by the KL-Max technique proposed in [32] are provided for comparison purposes. Since the computational complexity of KL-Max implemented with SVM is very high, in our experiments at each iteration an unlabeled sample is chosen from a randomly selected subset (made up of 100 samples) of unlabeled data. Note that the KL-Max technique can be implemented with any classifier that exploits posterior class probabilities for determining the decision boundaries [32]. In order to implement KL-Max technique with SVM, we converted the outputs of each binary SVM to posterior probabilities exploiting the Platt's method [39].

All experimental results are referred to the average accuracies obtained in ten trials according to ten initial randomly selected training sets. Results are provided as learning rate curves, which show the average classification accuracy versus the number of training samples used to train the SVM classifier. In all the experiments, the size of final training set $|T|$ is fixed to 473 for the Trento data set, 472 for the Pavia data set, and 483 for the KSC dataset. The total number of iterations is given by the ratio between the number of samples to be added to the initial training set and the pre-defined value of $h$.

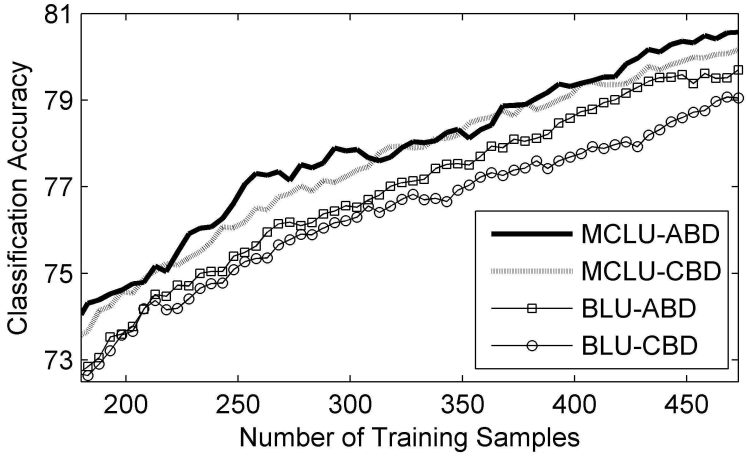
## VI. EXPERIMENTAL RESULTS

We carried out different kinds of experiments in order to: 1) compare the effectiveness of the different investigated techniques that we generalized to the multiclass case in different conditions; 2) assess the effectiveness of the novel ECBD technique; 3) compare the investigated methods and the proposed MCLU-ECBD technique with the techniques used in the RS literature; and 4) perform a sensitivity analysis with respect to different parameter settings and strategies.

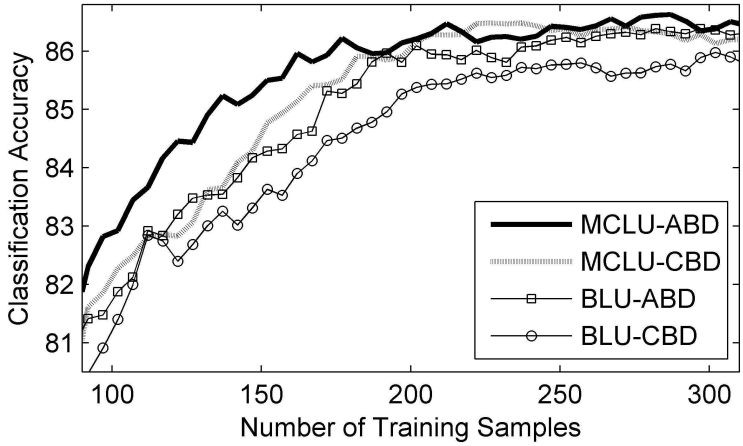
### *A. Results: Comparison among Investigated Techniques Generalized to the Multiclass Case*

In the first set of trials, we analyze the effectiveness of the investigated techniques generalized to multiclass problems. As an example, Fig. 4 compares the overall accuracies versus the number of initial training samples obtained by the MCLU-ABD, the MCLU-CBD, the BLU-

ABD and the BLU-CBD techniques with $h = 5$, $k$=5 and $\lambda = 0.6$. In the MCLU, $m$=20 samples are selected for both data sets. In the BLU, $m \leq 30$ and $m \leq 40$ samples are chosen for the Trento and Pavia data sets, respectively. The confidence value is calculated with the $c_{diff}(\mathbf{x})$ strategy for both MCLU and BLU, as preliminary tests pointed out that by fixing the query function, the $c_{diff}(\mathbf{x})$ strategy is more effective than the $c_{\min}(\mathbf{x})$ strategy in case of using MCLU, whether it provides similar classification performance to the $c_{\min}(\mathbf{x})$ strategy when using BLU. Fig. 4 shows that the MCLU-ABD technique is the most effective on both Trento and Pavia data sets. Note that similar behaviors are obtained by using different values of parameters (i.e., $m$, $h$, $\lambda$ and $k$). The effectiveness of the MCLU and BLU techniques for uncertainty assessment can be analyzed by comparing the results obtained by combining them with the same diversity techniques under the same conditions (i.e., same values for parameters). From Fig. 4, one can observe that the MCLU technique is more effective than the BLU in the selection of the most uncertain samples on both data sets (i.e., the average accuracies provided by the MCLU-ABD are higher than those obtained by the BLU-ABD and a similar behavior is obtained with the CBD). This trend is confirmed by using different values of parameters (i.e., $m$, $h$, $\lambda$ and $k$ ). The ABD and CBD techniques can be compared by combining them with the same uncertainty technique under the same conditions (i.e., same values for parameters). From Fig. 4, one can see that the ABD technique is more effective than the CBD technique. The same behavior can also be observed by varying the values of parameters (i.e., $m$, $h$, $\lambda$ and $k$).




(a)




(b)

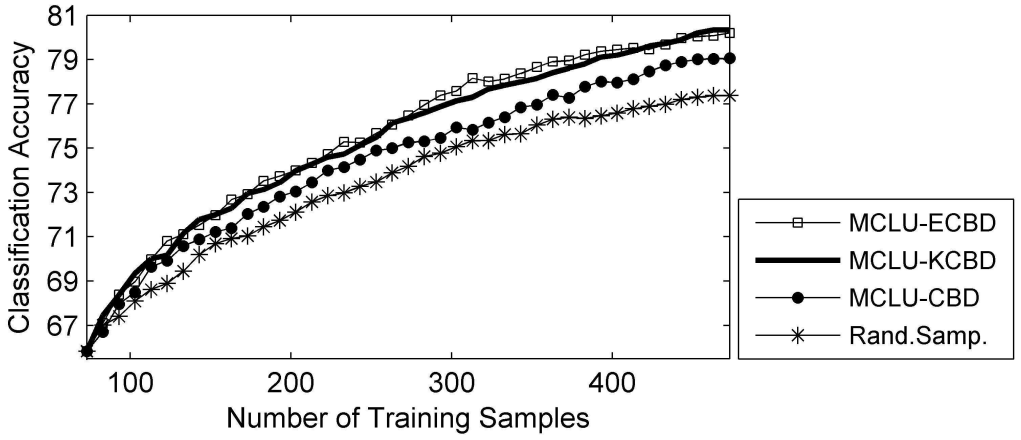
Fig. 4. Overall classification accuracy obtained by the MCLU and BLU uncertainty criteria when combined with the ABD and CBD diversity techniques in the same conditions for (a) Trento, and b) Pavia data sets. The learning curve of Pavia data set is reported starting from 87 samples until 312 in order to better highlight the small differences.

*B. Results: Proposed MCLU-ECBD Technique*

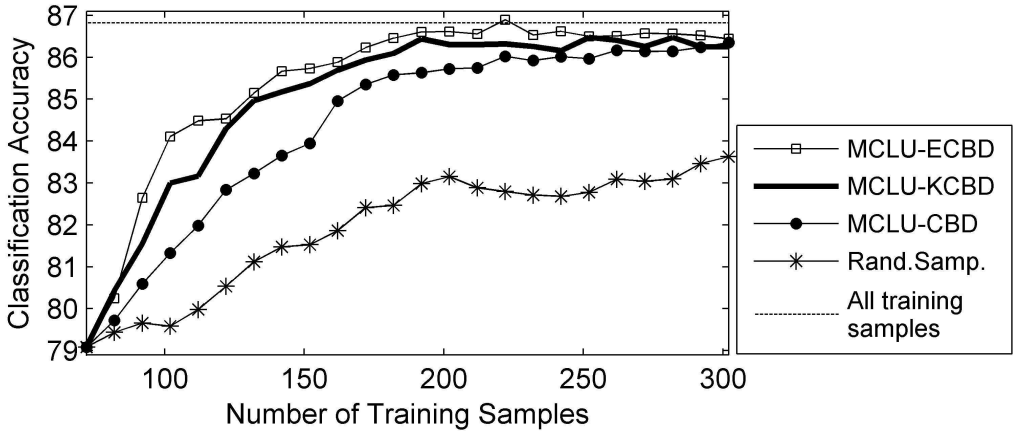
In the second set of trials, we compare the standard CBD with the proposed novel ECBD using the MCLU uncertainty technique with the $c_{diff}(\mathbf{x})$ strategy and fixing the same parameter values. For discriminating the contributions of the two novel components of the proposed ECBD, we report also the classification results obtained with kernel CBD (KCBD), where the clustering is performed with kernel *k*-means but the medoid sample is selected from each cluster, instead of the proposed most uncertain sample.

As an example, Fig. 5 shows the results obtained with $m = 40, h = 10, k = 10$ for the three data sets. From these graphs one can see that the ECBD technique provides the selection of more informative samples compared to CBD and KCBD techniques, achieving higher accuracies for the same number of samples (or the same accuracy with less samples). Moreover, we can observe that performing the clustering in the kernel space can improve the results with respect to standard CBD, while the selection of the most uncertain sample from each cluster (in the kernel space) allows one to further slightly improve the classification accuracies. Tables 4, 5 and 6 report the mean and standard deviation of classification accuracies obtained on ten trials versus different iteration numbers and different training data size $|T|$ for the three considered data sets. From these tables we can observe that the classification accuracies obtained with the proposed MCLU-ECBD are generally higher and also more stable (i.e., with lower standard deviation over the ten trials)

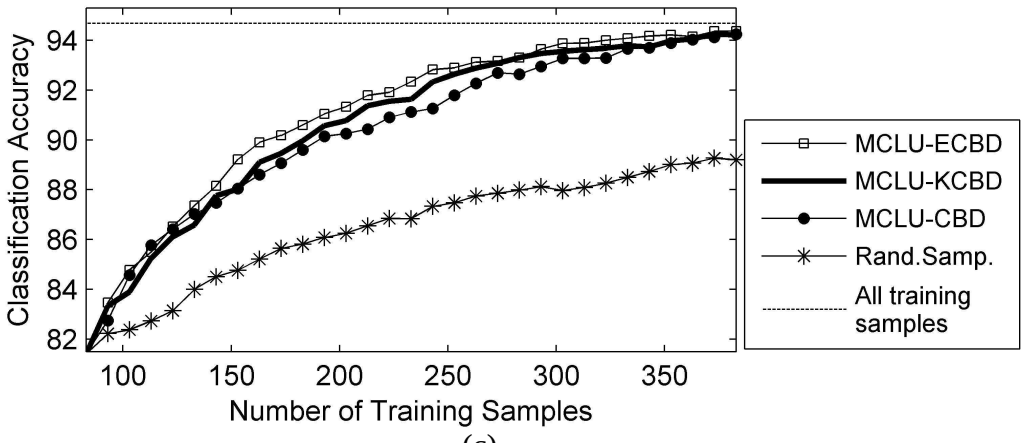
with respect to both the CBD and KCBD. These results are also confirmed in other experiments with different values of parameters (not reported for space constraints).




(a)




(b)




(c)

Fig. 5. Overall classification accuracy obtained by the MCLU uncertainty criterion when combined with the standard CBD, KCBD and the proposed ECBD diversity techniques for (a) Trento, (b) Pavia and (c) KSC data sets. The line "All training samples" reported in (b) and (c) shows the accuracy obtained using the full pool as training set.

TABLE 4. AVERAGE CLASSIFICATION ACCURACY (CA) AND STANDARD DEVIATION (STD) OBTAINED ON TEN TRIALS FOR DIFFERENT TRAINING DATA SIZE $(|T|)$ AND ITERATION NUMBERS (ITER. NUM) (TRENTO DATA SET)

| Technique | $\|\mathbf{T}\|$ = 163 (Iter.Num. 9) | | $\|\mathbf{T}\|$ = 193 (Iter. Num. 12) | | $\|\mathbf{T}\|$ = 333 (Iter. Num. 26) | |
|---|---|---|---|---|---|---|
| | CA | std | CA | Std | CA | std |
| Proposed MCLU-ECBD | 72.67 | 0.91 | 73.73 | 1.26 | 78.12 | 0.83 |
| MCLU-KCBD | 72.28 | 1.25 | 73.44 | 1.51 | 77.83 | 1.00 |
| MCLU-CBD | 71.39 | 1.60 | 72.81 | 1.30 | 76.39 | 1.22 |

TABLE 5. AVERAGE CLASSIFICATION ACCURACY (CA) AND STANDARD DEVIATION (STD) OBTAINED ON TEN TRIALS FOR DIFFERENT TRAINING DATA SIZE $(|T|)$ AND ITERATION NUMBERS (ITER. NUM) (PAVIA DATA SET)

| Technique | $\|\mathbf{T}\|$ = 102 (Iter.Num. 3) | | $\|\mathbf{T}\|$ = 142 (Iter. Num. 7) | | $\|\mathbf{T}\|$ = 172 (Iter. Num. 10) | |
|---|---|---|---|---|---|---|
| | CA | std | CA | std | CA | std |
| Proposed MCLU-ECBD | 84.10 | 1.66 | 85.66 | 1.29 | 86.23 | 1.09 |
| MCLU-KCBD | 83.58 | 1.78 | 85.32 | 1.44 | 85.34 | 1.17 |
| MCLU-CBD | 81.32 | 1.77 | 83.65 | 1.62 | 85.35 | 1.39 |

TABLE 6. AVERAGE CLASSIFICATION ACCURACY (CA) AND STANDARD DEVIATION (STD) OBTAINED ON TEN TRIALS FOR DIFFERENT TRAINING DATA SIZE $(|T|)$ AND ITERATION NUMBERS (ITER. NUM) (KSC DATA SET)

| Technique | $\|\mathbf{T}\|$ = 173 (Iter.Num. 9) | | $\|\mathbf{T}\|$ = 203 (Iter. Num. 12) | | $\|\mathbf{T}\|$ = 243 (Iter. Num. 16) | |
|---|---|---|---|---|---|---|
| | CA | std | CA | std | CA | std |
| Proposed MCLU-ECBD | 90.18 | 1.69 | 91.33 | 1.37 | 92.82 | 0.46 |
| MCLU-KCBD | 89.46 | 1.65 | 90.77 | 1.29 | 92.33 | 0.82 |
| MCLU-CBD | 89.06 | 1.94 | 90.25 | 1.04 | 91.26 | 1.20 |

*C) Results: Comparison Among the Proposed AL Techniques and Literature Methods*

In the third set of trials, we compare the investigated and proposed techniques with AL techniques proposed in the RS literature. We compare the MCLU-ECBD and the MCLU-ABD techniques with the MS-cSV [31], the EQB [31] and the KL-Max [32] methods. According to the accuracies reported in section V.A, we present the results obtained with the MCLU, which is more effective than the BLU. Fig. 6 shows the average accuracies versus the number of training samples obtained in the case of $h = 5$ ($h$=1 only for KL-Max) for the three data sets. In the figure we report the highest average accuracy (obtained with the best values of the parameters $\lambda$ and $m$) of each technique. Note that, since the MCLU-CBD proved less accurate than the MCLU-ECBD (see section V. B), its results are no more reported here. For the Trento and KSC data sets, the highest accuracies for MCLU-ECBD are obtained with $m = 30$ (while $k$=5), whereas the best results for

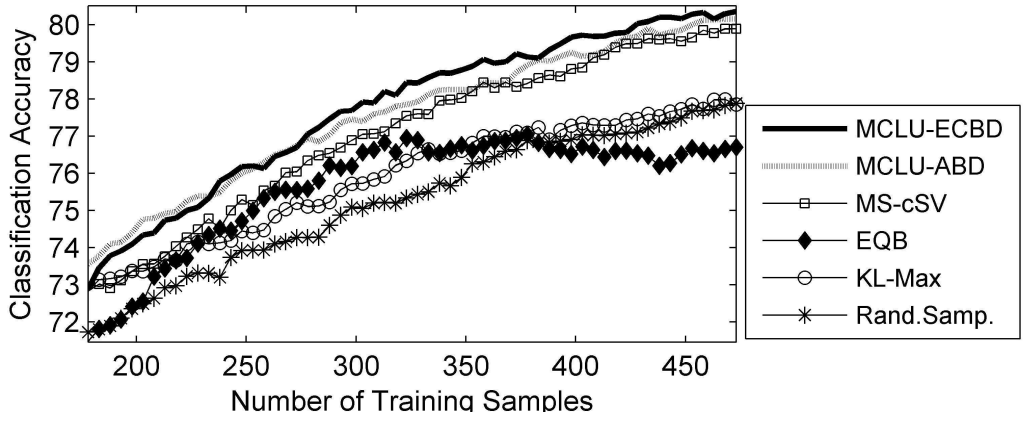
MCLU-ABD are obtained with $\lambda$=0.6 and $m = 20$. For the Pavia data set, the highest accuracies for MCLU-ECBD are obtained with $m = 20$ (while $k$=5), whereas the best results for MCLU-ABD are obtained with $\lambda$=0.6 and $m = 20$.

By analyzing Fig. 6a (Trento data set) one can observe that MCLU-ECBD and MCLU-ABD results are in general better than MS-cSV and significantly better than EQB and KL-Max results. The KL-Max accuracies are similar to the MS-cSV accuracies at early iterations but significantly decrease with bigger sizes of the training set with respect to MCLU-ECBD, MCLU-ABD, and MS-cSV. The accuracy at the final size of the training set (473 samples) obtained with the EQB is significantly smaller than those obtained with the other techniques. This accuracy could be improved by re-estimating the SVM model, but in our experiments, in order to have a fair comparison, we decided not to perform model re-estimation for any of the considered AL methods. On this data set, the classification accuracy obtained with the final training size does not reach the convergence with none of the considered AL methods; i.e., using the full pool for training the SVM classifier (i.e., 1818 samples) the obtained accuracy is 84.24%, while the maximum accuracy reached with AL methods with 473 samples does not reach 81% of accuracy. The results obtained on the Pavia data set (see Fig. 6b) show that the proposed MCLU-ECBD technique leads to the highest accuracies in most of the iterations; furthermore, it achieves convergence in less iterations (and thus with a smaller number of labeled samples) than the other techniques. The MCLU-ECBD technique yield an accuracy of 86.77% with only 232 samples, while using the full pool as training set (1825 samples) we obtain an accuracy of 86.82%. The MCLU-ABD method provides in general slightly lower accuracy than MCLU-ECBD and higher accuracies than MS-cSV. The EQB method provides significantly lower accuracies with respect to MCLU-ECBD, MCLU-ABD and MS-cSV in the first iterations and reaches the convergence around 250 samples. The KL-Max technique accuracies are in general significantly smaller than those achieved with other techniques for the same numbers of labeled samples. The results obtained on the KSC data set (Fig. 6c) show that the proposed MCLU-ECBD provides similar results compared to MCLU-ABD, and both of them significantly outperforms the rest of the other considered methods. The MCLU-ECBD technique reaches an accuracy of 94.64% with only 413 samples, while using the full pool as training set (2052 samples) we obtain an accuracy of 94.68%. As opposed to the other data sets, with KSC the EQB method provides in general better accuracies than the MS-cSV technique. This is in agreement with what is reported in [31].

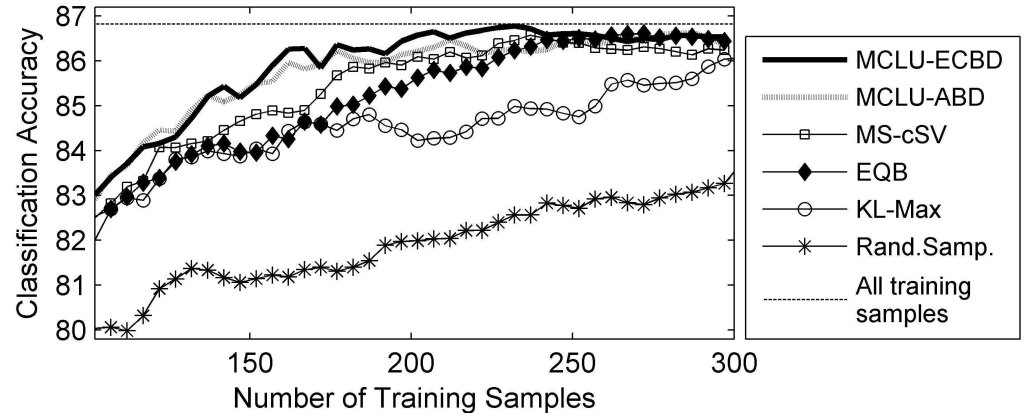
For a more detailed comparison, additional experiments were carried out on varying the values of the parameters (numerical results are not reported for space constraints). In all cases, we observed that MCLU-ECBD and MCLU-ABD techniques yield higher classification accuracies

than the other AL techniques when small $h$ values are considered. Thus, the values for the parameter $m$ and $\lambda$, which should be defined by the user, are not critical for the accuracies of both the MCLU-ECBD and MCLU-ABD techniques. Moreover, we observed that the EQB technique is not effective when selecting a small number $h$ of samples, but when relatively high $h$ values are considered it can lead to accuracies close to those of MCLU-ECBD and MCLU-ABD. MS-cSV can not be used for high $h$ values when small initial training set are available since the maximum number of $h$ is equal to the total number of SVs. KL-Max results can only be provided for $h$=1 and the related accuracies are smaller than those obtained with both MCLU-ECBD and MCLU-ABD methods. In order to assess the statistical significance of the difference between the proposed method and state-of-the-art methods we computed the McNemar test [41] for five different training set sizes for all data sets. We found that the accuracies obtained by the MCLU-ECBD technique are statistically different from those obtained by MS-cSV and EQB with a confidence greater than 95%.

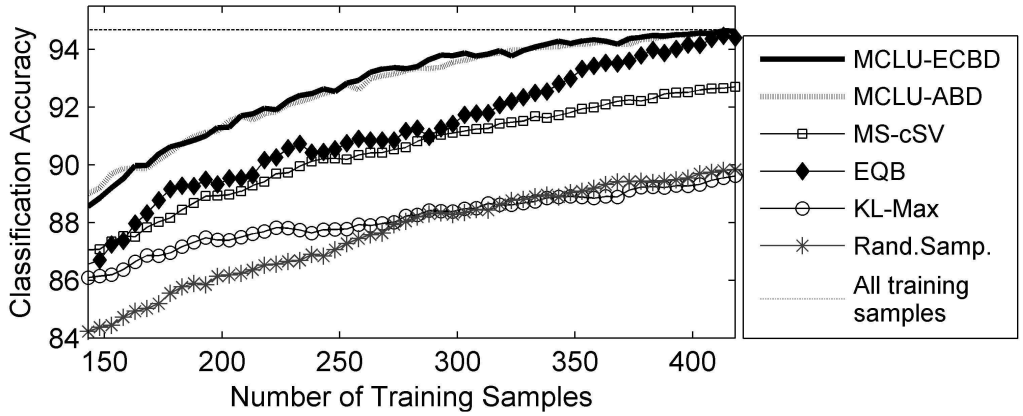
Table 7 reports the computational time (in seconds) required (for one trial) by MCLU-ECBD, MCLU-ABD, MS-cSV, and EQB for different $h$ values, and the computational time taken from KL-Max (related to $h$=1). In this case, the value of $m$ for MCLU-ECBD and MCLU-ABD is fixed to $4h$. It can be noted that MCLU-ECBD and MCLU-ABD are fast both for small and high values of $h$. The computational time of MS-cSV and EQB is very high in the case of small $h$ values, whereas it decreases by increasing the $h$ value. The largest computational time is obtained with KL-Max that with an SVM classifier requires the use of the Platt algorithm for computing the class posterior probabilities. The results obtained on the three data sets confirm that both the proposed MCLU-ECBD and the investigated MCLU-ABD are very effective in terms of both classification accuracy and computation complexity, and they outperforms state-of-the-art methods in most of the cases.




(a)




(b)




(c)

Fig. 6. Overall classification accuracy obtained by the MCLU-ECBD, MCLU-ABD, MS-cSV, EQB and KL-Max techniques for (a) Trento, (b) Pavia, and (c) KSC data sets. The learning curves are reported starting from 178 samples for Trento, 92 samples for Pavia, and 142 samples for KSC data sets in order to better highlight the differences. The line "All training samples" reported in (b) and (c) shows the accuracy obtained using the full pool as training set.

Table 7. Examples of computational time (in seconds) taken from the MCLU-ECBD, MCLU-ABD, MS-cSV, EQB and KL-Max techniques

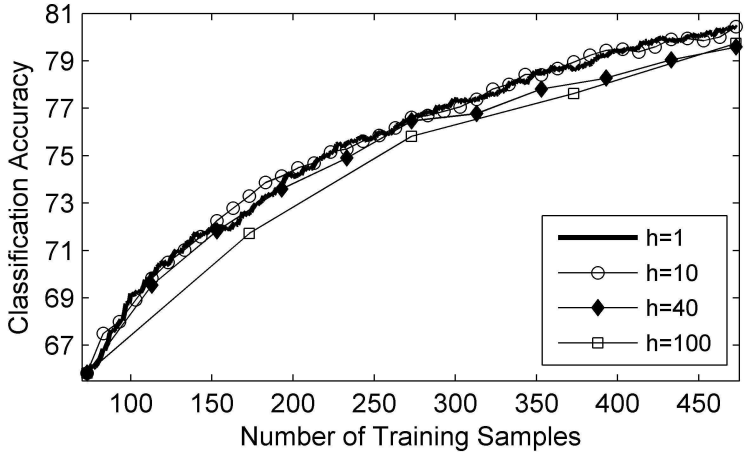
| **Data Set** | **Technique** | ***h*** | | | | |
|---|---|---|---|---|---|---|
| | | **1** | **5** | **10** | **40** | **100** |
| Trento | Proposed MCLU-ECBD | - | 10 | 6 | 8 | 12 |
| | MCLU-ABD | - | 10 | 6 | 7 | 10 |
| | MS-cSV | - | 584 | 452 | - | - |
| | EQB | - | 300 | 148 | 34 | 12 |
| | KL-Max | 72401 | - | - | - | - |
| Pavia | Proposed MCLU-ECBD | - | 10 | 6 | 7 | 11 |
| | MCLU-ABD | - | 10 | 5 | 6 | 10 |
| | MS-cSV | - | 384 | 193 | - | - |
| | EQB | - | 138 | 68 | 16 | 6 |
| | KL-Max | 71380 | - | - | - | - |
| KSC | Proposed MCLU-ECBD | - | 17 | 14 | 15 | 19 |
| | MCLU-ABD | - | 18 | 13 | 15 | 19 |
| | MS-cSV | - | 977 | 614 | 134 | - |
| | EQB | - | 652 | 312 | 95 | 21 |
| | KL-Max | 97309 | - | - | - | - |

*D. Results: Sensitivity Analysis with Respect to Different Parameter Settings and Strategies*

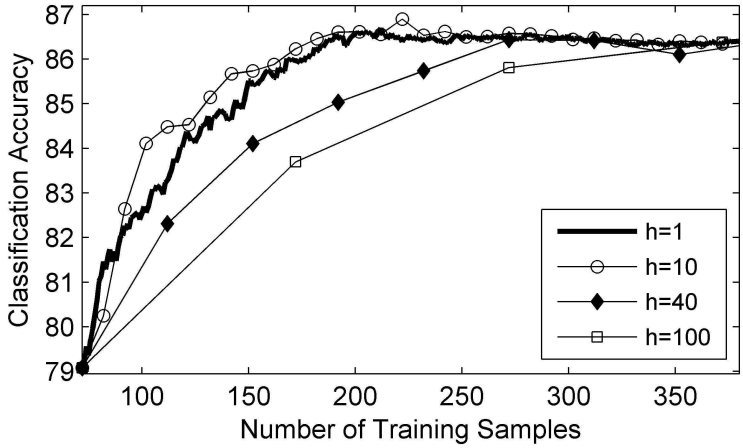
The aim of the fourth set of trials is to analyze the considered AL techniques under different parameter settings and strategies.

*Analysis of the effect of different batch size values*

We carried out an analysis of the performances of different AL techniques varying the value of the batch size *h* by fixing the query function. As an example, Fig. 7 shows the accuracies versus the number of training samples obtained on Trento and Pavia data sets adopting the proposed MCLU-ECBD query function. The results are obtained with $m = 4h$ and $k = h$. The computational time taken from the MCLU-ECBD (related to one trial) for different *h* values is given in Table 8. From the table one can observe that the largest learning time corresponds to the case where one sample is selected at each iteration. The computational time decreases by increasing the *h* value. From Fig. 7, one can see that for both data sets selecting small *h* values results in similar (or better) classification accuracies compared to those obtained selecting only one sample at each iteration. On the contrary, high *h* values decrease the classification accuracy without decreasing the computational time if compared to small *h* values. Another interesting observation is that on the Pavia data set, when using small *h* values, convergence is achieved with less samples than when using large values. Note that similar behaviors are obtained with the other query functions.




(a)




(b)

Fig. 7. Overall classification accuracy versus the number of training samples obtained by the MCLU-ECBD technique with different *h* values for a) Trento and b) Pavia data sets

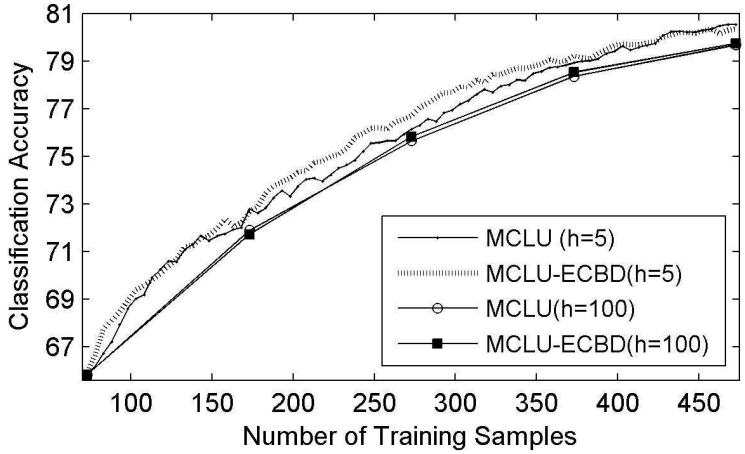
TABLE 8. EXAMPLES OF COMPUTATIONAL TIME (IN SECONDS) TAKEN FROM THE MCLU-ECBD TECHNIQUE WITH RESPECT TO DIFFERENT *H* VALUES

| **Data Set** | **MCLU** | **MCLU-ECBD** | | |
|---|---|---|---|---|
| | ***h*** | ***h*** | | |
| | **1** | **10** | **40** | **100** |
| Trento | 47 | 6 | 8 | 12 |
| Pavia | 46 | 6 | 7 | 11 |

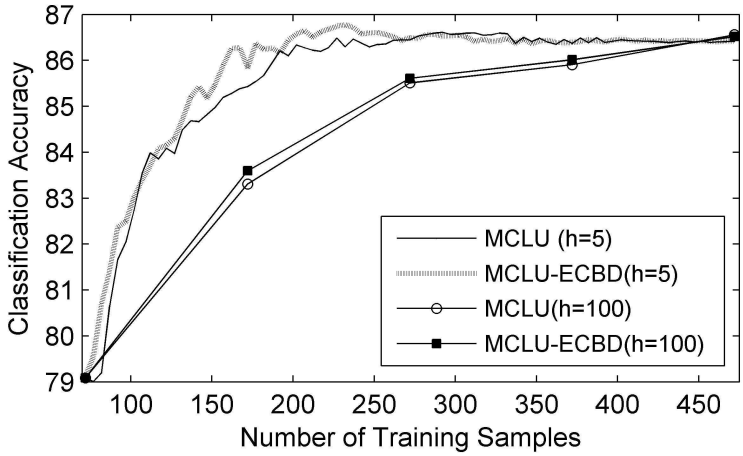
*Analysis of the effect of different batch size values h on the diversity criteria*

Finally, we analyze the accuracy obtained by using only uncertainty criteria and the combination of uncertainty with diversity criteria for different *h* values. As an example, Fig. 8 shows the average accuracy versus the number of training samples obtained by MCLU (*m* is fixed equal to *h* for a fair comparison) and MCLU-ECBD with $m = 4h, h = 5,100$ and $k = h$. One can observe that, as expected, using only the uncertainty criterion provides poor accuracies when *h* is small, whereas the classification performances are significantly improved by using both

uncertainty and diversity criteria. On the contrary, the choice of complex query functions is not justified when a large batch of samples is added to the training set at each iteration (i.e., similar results can be obtained with and without considering diversity). This mainly depends on the intrinsic capability of a large number of samples *h* to represent patterns in different positions of the feature space. Similar behaviors are observed with the other query functions.




(a)




(b)

Fig. 8. Overall classification accuracy versus the number of training samples for the uncertainty criterion and the combination of uncertainty and diversity criteria with different *h* values: a) Trento and b) Pavia data sets

## VII. DISCUSSION AND CONCLUSION

In this paper, AL in RS classification problems has been addressed. The use of AL techniques for the classification of RS images reduces the computational time and the number of labeled samples used for training the supervised algorithm (which is associated to cost and time for defining the training set) and increases the classification accuracy with respect to traditional passive techniques. Query functions based on MCLU and BLU in the uncertainty step, and ABD and CBD in the diversity step have been generalized to multiclass problems and experimentally

compared on different RS data sets. Furthermore, a novel MCLU-ECBD query function has been proposed. This query function is based on MCLU in the uncertainty step and on the analysis of the distribution of most uncertain samples by means of *k*-means clustering in the kernel space. Moreover, it selects the batch of samples at each iteration according to the identification of the most uncertain sample of each cluster.

In the experimental analysis we compared the investigated and the proposed techniques with state-of-the-art AL methods adopted in RS applications for the classification of both VHR multispectral and hyperspectral images. By this comparison we observed that the proposed MCLU-ECBD method and the investigated MCLU-ABD method resulted in higher accuracy with respect to other state-of-the-art methods on the three considered data sets for the same number of labeled samples. In addition they can reach convergence with a smaller number of labeled samples than the other techniques. We underline that this is a very important advantage, because the main goal of AL is to perform an effective learning of a classifier with the minimum possible number of labeled samples. It was also shown that MCLU-ECBD and MCLU-ABD techniques are generally more effective than the other considered techniques also in terms of computational time (especially for small values of *h*). Thus, they are actually well-suited for applications in which sample labeling is carried out with both ground survey and image photointerpretation. Moreover, we showed that: 1) the MCLU technique is more effective in the selection of the most uncertain samples for multiclass problems than the BLU technique; 2) the $c_{diff}(\mathbf{x})$ strategy is more precise than the $c_{\min}(\mathbf{x})$ strategy to assess the confidence value in the MCLU technique; 3) it is possible to have similar (sometimes better) classification accuracies with lower computational complexity when selecting small batches of *h* samples rather than selecting only one sample at each iteration; 4) the use of both uncertainty and diversity criteria is necessary when *h* is small, whereas high *h* values do not require the use of complex query functions; 5) the performance of the standard CBD technique can be significantly improved by adopting the ECBD technique, thanks to both the kernel *k*-means clustering and the selection of the most uncertain sample of each cluster instead of the medoid sample.

In greater detail, on the basis of our experiments we can state that:

1) The proposed novel MCLU-ECBD technique shows excellent performance in terms of classification accuracy and computational complexity, improving the performance of the standard CBD method. It is important to note that this technique has a computational complexity suitable to the selection of batch of samples made up of any desired number of patterns, thus it is compatible with both photointerpretation and ground-survey based labeling of unlabeled data.

2) The MCLU-ABD technique provides slightly lower or similar classification accuracies than the MCLU-ECBD method in most of the cases, with a similar computational time. It can be used for selecting a batch made up of any desired number of *h* samples. Thus, also the MCLU-ABD technique is suitable for both photointerpretation and ground-survey based labeling of unlabeled data.

3) The MS-cSV technique provides quite good classification accuracies. However, the maximum value of *h* that can be used is equal to the total number of SVs $|\text{SVs}|$ (i.e., $h \leq |\text{SVs}|$ and therefore it can not be implemented for any *h* value). Nevertheless, the original algorithm presented in [31] could be modified in order to avoid this issue. In the case of small *h* values, the computational complexity of this technique is much higher than that of the other investigated and proposed techniques. This complexity decreases when *h* increases.

4) The EQB technique results in poorer classification accuracies with respect to the other techniques with small values of *h* and comparable classification accuracies with high values of *h*. The computational complexity of this technique is very high in case of selecting few samples, and decreases while *h* increases. Although it is possible to select any desired number of *h* samples with the EQB, it is not properly suitable when few labeled samples are included by photointerpretation at each iteration due to its high computational complexity and poor classification performance with small *h* values.

5) The KL-Max technique is different from the above mentioned methods since it is only able to select one sample at each iteration and can be implemented with any classifier that estimates a posteriori class probabilities. In our experiments we converted the SVM results into probabilities and results showed that this technique is not effective with SVM classifiers and requires very high computational complexity.

We assessed the compatibility of the considered AL techniques with the strategies to label unlabeled samples by image photointerpretation or ground data collection in order to provide some guidelines to the users under different conditions. As mentioned before, in the case of VHR images, in many applications the labeling of unlabeled samples can be achieved by photointerpretation, which is compatible with several iterations of the AL process in which a small value *h* of samples are included in the training set at each step according to an interactive procedure of labeling carried out by an expert operator. On our VHR data set, we observed that batches of $h = 5$ or 10 samples can result in the best tradeoff between accuracy and number of considered labeled samples. In the case of hyperspectral or medium/low resolution multispectral data, expensive and time consuming ground surveys are usually necessary for the labeling process.

Under this last condition, only very few iterations of the AL process are realistic. Thus, it is reasonable to collect large batches (of e.g., hundreds of samples) for each iteration. In this case, we observed that sophisticated query functions are not necessary, as with many samples often the uncertainty criterion alone is sufficient for obtaining good accuracies. As a final remark, we point out that in real applications, some geographical areas may be not accessible for ground survey (or the process might be too expensive). Thus, the definition of the pool $U$ should be carried out carefully, in order to avoid these areas.

As a future development, we consider to extend the proposed method by including a spatially-dependent labeling costs, which takes into account that traveling to a certain area involves some type of costs that should be taken into account in the selection of batch of unlabeled samples [27]. In addition, we plan to define hybrid approaches that integrate semisupervised and AL methods in the classification of RS images.

## APPENDIX

TABLE 9. TABLE OF SYMBOLS

| Symbol | Description | Symbol | Description |
|---|---|---|---|
| $n$ | Total class number | $\mathbf{x}_v^{MCLU-ECBD}$ | $v$-th sample selected using ECBD |
| $m$ | Number of unlabeled samples selected at the uncertainty step | $I$ | Set of indices of $m$ most uncertain samples |
| $h$ | Total number of unlabeled samples added to the training set at each iteration (batch size) | $X$ | Set of $h$ samples selected by a query function |
| $q$ | Number of unlabeled samples selected for each binary SVM in the BLU technique | $I \,/\, X$ | Indices of unlabeled samples of $I$ that are not contained in $X$ |
| $\rho$ | Number of total samples selected in the BLU technique (i.e., $\rho = qn$) | $\lvert X \rvert$ | Cardinality of set $X$ |
| $u$ | Total number of unlabeled samples | $t$ | Index of an unlabeled sample that will be included in $X$ |
| $\mathbf{x}_{j,i}^{BLU}$ | Selected $j$-th sample from the $i$-th SVM based on the BLU technique | $\lambda$ | Weighting parameter for the ABD technique |
| $\mathbf{x}_j^{BLU}$ | Selected $j$-th sample based on the BLU technique | $S$ | Supervisor |
| $\mathbf{x}_j^{MCLU}$ | Selected $j$-th sample based on the MCLU technique | $Q$ | Query function |
| $c(\mathbf{x})$ | Confidence value of pattern $\mathbf{x}$ | $T$ | Training set |
| $c_{\min}(\mathbf{x})$ | Minimum distance function of pattern $\mathbf{x}$ | $U$ | Unlabeled sample pool |
| $c_{diff}(\mathbf{x})$ | Difference function of pattern $\mathbf{x}$ | $G$ | Classifier |
| $r_{1\max}$ | Index of the binary SVM with highest output score | $TS$ | Test set |
| $r_{2\max}$ | Index of the binary SVM with the second highest output score | $V$ | Validation set |

| | | | |
|---|---|---|---|
| $f_i(\mathbf{x})$ | Functional distance of pattern $\mathbf{x}$ to the *i*-th hyperplane | $k$ | Number of Clusters for the CBD or ECBD techniques |
| $K(\cdot,\cdot)$ | Kernel function | $C_v$ | *v*-th cluster |
| $\phi(\cdot)$ | Nonlinear mapping function | $\mu_v$ | *v*-th cluster center |
| $\gamma$ | Spread of the RBF kernel function | $\delta(\cdot)$ | Indicator function |
| $C$ | SVM penalty parameter | $\eta_v$ | Pseudo centre of *v*-th cluster |

TABLE 10. TABLE OF ACRONYMS

| Acronyms | Description | Acronyms | Description |
|---|---|---|---|
| RS | Remote Sensing | KCBD | Kernel CBD |
| AL | Active Learning | ECBD | Enhanced CBD |
| SVM | Support Vector Machine | BLU-ABD | BLU with ABD |
| SV | Support Vector | BLU-CBD | BLU with CBD |
| RBF | Radial Basis Function | MCLU-ABD | MCLU with ABD |
| OAA | One Against All | MCLU-CBD | MCLU with CBD |
| MS | Margin Sampling | MCLU-ECBD | MCLU with ECBD |
| BLU | Binary-Level Uncertainty | MS-cSV | MS by closest Support Vector |
| MCLU | Multiclass-Level Uncertainty | EQB | Entropy Query-by Bagging |
| ABD | Angle Based Diversity | KL-Max | Kullback–Leibler-Max technique |
| CBD | Clustering Based Diversity | | |


## ACKNOWLEDGMENT

This work was developed during an internship of Begüm Demir at the Remote Sensing Laboratory of the Department of Information Engineering and Computer Science, University of Trento. The internship was supported by the International Research Fellowship Programme (2214) of The Scientific and Technological Research Council of Turkey (TÜBİTAK).


## REFERENCES


[1] B. M. Shahshahani and D. A. Landgrebe, "The effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 32, vo. 5, pp. 1087–1095, September 1994.

[2] L. Bruzzone, M. Chi, M. Marconcini, "A Novel Transductive SVM for the Semisupervised Classification of Remote-Sensing Images", *IEEE Trans. on Geoscience and Remote Sensing*, vol. 44, no. 11, pp. 3363-3373, 2006.

[3] M. Chi, L. Bruzzone, "Semi-supervised Classification of Hyperspectral Images by SVMs Optimized in the Primal", *IEEE Trans. on Geoscience and Remote Sensing*, vol. 45, no. 6, Part 2, pp. 1870-1880, June 2007.

[4] M. Marconcini, G. Camps-Valls, L. Bruzzone, “A Composite Semisupervised SVM for Classification of Hyperspectral Images”, *IEEE Geoscience and Remote Sensing Letters* , vol. 6, no. 2, pp. 234-238, 2009.

[5] G. Camps-Valls, T.V. Bandos Marsheva, and D. Zhou, “Semi-Supervised Graph-Based Hyperspectral Image Classification”, *IEEE Trans. on Geoscience and Remote Sensing*, vol. 45, no. 10, pp. 3044-3054, October 2007.

[6] M. Li and I. Sethi, “Confidence-Based active learning”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 8, pp. 1251-1261, 2006.

[7] D. D. Lewis and W. A. Gale, “A sequential algorithm for training text classifiers,” *in Proc. 17th Annu. Int. ACM-SIGIR Conf. Res. Dev. Inf. Retrieval*, W. B. Croft and C. J. van Rijsbergen, Eds., London, U.K., pp. 3–12, 1994.

[8] C. Campbell, N. Cristianini, and A. Smola, “Query Learning with Large Margin Classifiers”, *Proc. 17th Int’l Conf. Machine Learning (ICML ’00)*, pp. 111-118, 2000.

[9] G. Schohn and D. Cohn, “Less is More: Active Learning with Support Vector Machines”, *Proc. 17th Int’l Conf. Machine Learning (ICML ’00)*, pp. 839-846, 2000.

[10] S. Tong and D. Koller, “Support Vector Machine Active Learning with Applications to Text Classification”, *Proc. 17th Int’l Conf. Machine Learning (ICML ’00)*, pp. 999-1006, 2000.

[11] T. Luo, K. Kramer, D.B. Goldgof, L.O. Hall, S. Samson, A. Remsen, and T. Hopkins, “Active Learning to Recognize Multiple Types of Plankton,” J. Machine Learning Research, vol. 6, pp. 589-613, 2005.

[12] P. Mitra, B. U. Shankar, and S. K. Pal, “Segmentation of multispectral remote sensing images using active support vector machines,” *Pattern Recognit. Lett*., vol. 25, no. 9, pp. 1067–1074, Jul. 2004.

[13] K. Brinker, “Incorporating Diversity in Active Learning with Support Vector Machines,” *Proceedings of the International Conference on Machine Learning*, Washington DC, pp. 59-66, 2003.

[14] Z. Xu, K. Yu, V. Tresp, X. Xu, and J. Wang, “Representative sampling for text classification using support vector machines,” *25th European Conf. on Information Retrieval Research*,pp. 393-407, 2003.

[15] H. T. Nguyen and A. Smeulders, “Active learning using pre-clustering,” in Proc. 21th ICML, Banff, AB, Canada, pp. 623-630, 2004.

[16] D. Cohn, Z. Ghahramani, and M.I. Jordan, “Active Learning with Statistical Models,” J. Artificial Intelligence Research, vol. 4, pp. 129-145, 1996.

[17] K. Fukumizu, “Statistical Active Learning in Multilayer Perceptrons”, *IEEE Trans. Neural Networks*, vol. 11 , no. 1, pp. 17-26, Jan. 2000.

[18] N. Roy and A. McCallum, “Toward optimal active learning through sampling estimation of error reduction,*” in Proc. ICML, Williamstown*, MA, 2001, pp. 441–448.

[19] H. S. Seung, M. Opper, and H. Smopolinsky, “Query by committee”, *Proc. 5th Annu. ACM Workshop Comput. Learning Theory, Pittsburgh,PA*, pp. 287–294, 1992.

[20] Y. Freund, H.S. Seung, E. Shamir, and N. Tishby, “Selective Sampling Using the Query by Committee Algorithm,” Machine Learning, vol. 28, pp. 133-168, 1997.

[21] I. Dagan and S. P. Engelson, “Committee-based sampling for training probabilistic classifiers,” in Proc. ICML, San Francisco, CA, 1995, pp. 150–157.

[22] N. Abe and H. Mamitsuka, “Query learning strategies using boosting and bagging,” *in Proc. ICML*, Madison, WI, pp. 1–9, 1998.

[23] V. N. Vapnik, *The Nature of Statistical Learning Theory*, 2nd ed., New York: Springer, 2001.

[24] F. Melgani, L. Bruzzone, “Classification of Hyperspectral Remote Sensing Images With Support Vector Machines,” *IEEE Trans. on Geoscience and Remote Sensing*, vol. 42, no. 8, pp. 1778- 1790, Aug. 2004.

[25] S.C. Hoi, R. Jin, J. Zhu, and M.R. Lyu, “Batch Mode Active Learning and Its Application to Medical Image Classification,” Proc. 23rd Int’l Conf. Machine Learning (ICML ’06), pp.417-424, June 2006.

[26] S.C. Hoi, R. Jin, J. Zhu, and M.R. Lyu, “Batch Mode Active Learning with Applications to Text Categorization and Image Retrieval”, *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1233 – 1248, Sept. 2009.

[27] A. Liu, G. Jun, J. Ghosh, “Active learning of hyperspectral data with spatially dependent label acquisition costs”, *IEEE Int. Geoscience and Remote Sensing Symposium 2009, (IGARSS '09)*, Cape Town, South Africa.

[28] C. Burges, “A tutorial on support vector machines for pattern recognition,” *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[29] Y. Zhang, X. Liao, and L. Carin, "Detection of Buried Targets Via Active Selection of Labeled Data: Application to Sensing Subsurface UXO", *IEEE Trans. on Geoscience and Remote Sensing*, vol. 42, no. 7, pp. 2535-2543, November 2004.

[30] Q. Liu, X. Liao, and L. Carin, "Detection of Unexploded Ordnance via Efficient Semisupervised and Active Learning", *IEEE Trans. on Geoscience and Remote Sensing*, vol. 46, no. 9, pp. 2558-2567, September 2008.

[31] D. Tuia, F. Ratle, F. Pacifici, M. Kanevski, and W. J. Emery, "Active Learning methods for remote sensing image classification," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 47, no. 7, pp. 2218 -2232, Jul. 2009.

[32] S. Rajan, J. Ghosh, and M. M. Crawford, "An active learning approach to hyperspectral data classification," *IEEE Trans. Geosci. Remote Sens*., vol. 46, no. 4, pp. 1231-1242, Apr. 2008.

[33] A. Vlachos, "A stopping criterion for active learning," in Computer, Speech and Language, vol. 22, no. 3, pp. 295-312, Jul. 2008.

[34] A. K. Jain and R. C. Dubes, Algorithms for Clustering Data. Upper Saddle River, NJ: Prentice-Hall, 1988.

[35] R. Zhang and A. I. Rudnicky, "A Large scale clustering scheme for kernel k-means," *IEEE International Conference on Pattern Recognition*, 11-15 August 2002, Quebec, Canada, pp. 289-292.

[36] B. Scholkopf, A. Smola, and K. R. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol.10, pp. 1299-1319, July 1998.

[37] M. Dalponte, L. Bruzzone, and D. Gianelle, "Fusion of hyperspectral and LIDAR remote sensing data for the estimation of tree stem diameters," *IEEE Int. Geoscience and Remote Sensing Symposium 2009, (IGARSS '09),* Cape Town, South Africa.

[38] L. Bruzzone, L. Carlin, "A Multilevel Context-Based System for Classification of Very High Spatial Resolution Images", *IEEE Trans. on Geoscience and Remote Sensing*, vol. 44, no. 9, pp 2587-2600, 2006.

[39] J.C. Platt, "Probabilities for SV Machines," in Advances in Large Margin Classifiers, A. Smola, P. Bartlett, B.Schölkopf, and D. Schuurmans, Eds., MIT Press, pp. 61-74, 1999.

[40] J. Ham, Y. Chen, M. M. Crawford, and J. Ghosh, "Investigation of the Random Forest Framework for Classification of Hyperspectral Data", *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 43, no. 3, 2005, pp. 492-501.

[41] G.M. Foody, “Thematic Map Comparison: Evaluating the Statistical Significance of Differences in Classification Accuracy”, *Photogrammetric Engineering and Remote Sensing,* vol 70, no. 5, 2004, pp. 627-633.