Learning user's confidence for active learning

Devis Tuia, *Member, IEEE*, Jordi Muñoz-Marí *Member, IEEE*

Abstract

In this paper, we study the applicability of active learning in operative scenarios: more particularly, we consider the well-known contradiction between the active learning heuristics, which rank the pixels according to their uncertainty, and the user's confidence in labeling, which is related to both the homogeneity of the pixel context and user's knowledge of the scene. We propose a filtering scheme based on a classifier that learns the confidence of the user in labeling, thus minimizing the queries where the user would not be able to provide a class for the pixel. The capacity of a model to learn the user's confidence is studied in detail, also showing the effect of resolution is such a learning task. Experiments on two QuickBird images of different resolutions (with and without pansharpening) and considering committees of users prove the efficiency of the filtering scheme proposed, which maximizes the number of useful queries with respect to traditional active learning.

Index Terms

Active learning, photointerpretation, user's confidence, bad states, VHR imagery, SVM.

I. INTRODUCTION

The advent of remote sensing imagery has opened a wide range of possibilities for surveying and analyzing the processes occurring on the surface of the Earth, thus allowing significant advances in the monitoring of agricultural [1], [2] or urban processes [3]. Among all the products

retrieved from very high resolution (VHR) imagery, classification maps describing landuse remain the most common. Several methods have been proposed to perform the classification task, but until now supervised methods remain the most successful approaches in the remote sensing community [4], [5]. However, these approaches rely on a set of pixels for which the class is known and that are used to train the classifier: the training set. The representativity of this set is crucial for the success of the classification [6], [7]. There are two major ways of obtaining a training set: one is to organize in-situ campaigns, where the landuse represented by the pixel is assessed and georeferenced by teams on the field; the other is to proceed by photointerpretation, i.e., having a human operator define labeled polygons on screen. Photointerpretation is particularly successful when dealing with VHR images, since the objects are recognizable on screen. In this case, two problems arise: first, many redundant pixels are added to the training set and second, the pixels added are not necessarily the most relevant for the classifier, but those that were most convenient (for a variety of reasons) in the photointerpretation phase. This last point is crucial, since the photointerpreter tends to label easily-recognizable pixels and to avoid areas of high variance/contrast (unless this variance is the specificity of the class) or underrepresented landuse classes.

To make photointerpretation efficient, active learning methods have recently been proposed in the community (a review in [8]): with active learning, the model and the user interact, the first ranking unlabeled pixels by their classification uncertainty and the latter providing the labels of the highly ranked. After retraining with these difficult pixels (now labeled), the model is expected to improve its performance greatly.

Several ranking criteria (heuristics) have been proposed in the remote sensing literature: some use committees of machines working on subsets of training pixels [9] or of input features [10], [11], others use the SVM decision function [12]–[14], posterior probabilities [15]–[17] or cluster coherence [18], [19] as a criterion to rank pixels. Questions of batch diversity [9], [20], inter-iterations diversity [21] and inter-dataset adaptation [16], [22] have also been considered. All these studied proved the efficiency of active learning heuristics in querying the most informative and diverse pixels in a pool of possible candidates.

Despite the theoretical appeal of this solution, the constraints of photointerpretation are often contradictory to the common active learning setting: while the first are driven by the user's capacity to recognize the objects on the surface, the second ranks the pixels by their uncertainty,

Fig. 1.   Six examples of pixels (in the white circles) selected to be labeled by a standard active learning criterion. Most of them are placed in areas under shadows, or at the boundary between several classes.

i.e., the complexity and mixture of their signature. As a consequence, the user is constantly required to label those samples with the highest uncertainty, which is a very complex (and often unfeasible) task even for a trained operator.

Figure 1 illustrates this principle for a 2.4 m QuickBird image: frequently, the pixels queried by an active learning heuristic are situated on the borders between objets, in areas which are not homogeneous, between several classes or in shadowed areas.

In all recent active learning literature (see above), this problem was avoided since, in order to avoid tedious photointerpretation, a pre-labeled set of candidates was provided and the human user was replaced by the a-priori known labels. This produces two undesired effects: first, all the borders between the objects were avoided, since the pre-labeled set is usually defined by photointerpretation and does not cover the whole of the image; and secondly, the user was considered infallible, in the sense that he could always give the correct answer. Reality, however, is often very different, and even an experienced operator can have difficulties in labeling the pixels returned by an active learning heuristic.

In this paper, we propose a solution to maximize the chances of querying pixels that the user can label correctly. To do so, we consider the idea of *learning the confidence of the user in a photointerpreting task*. Each time a user is unable to label a query, he gives information about a query he does not want to encounter again in the process, i.e., a bad state [23]. Avoiding bad states through estimation of the confidence of the oracle performing the task has been studied in robotics [24]–[26] and for time-evolving queries in policy evaluation [27].

To avoid bad states, we propose to train a second classifier learning to separate valid states (pixels that the user can label) from bad states. Using the confidence of this classifier, we build a mask for the active learning heuristic, which then avoids queries where the model learnt that the user was unable to give an answer. In short, we introduce the concept of learning the confidence of the user to avoid bad states when performing active learning while he/she is photointerpreting. We apply the proposed technique to two VHR images of urban areas. In both cases, the proposed active learning system allows to reconstruct the scene with a minimal number of queries, also minimizing the number of bad states. This means that for the same effort on the part of the user, the learning curve is steeper and all the queries presented are answered and are relevant for the model. Finally, the proposed algorithm also returns a confidence map that shows where the user would be capable to provide an answer.

The remainder of the paper is organized as follows: Section II illustrates the proposed methodology. Section III describes the data considered and the experimental setup of the experiments. Section IV presents the results ans Section V gives the conclusions of the work.

II. ACTIVE LEARNING WITH USER'S CONFIDENCE

This section presents the Active Learning with User's Confidence (AL-UC) method proposed. We first review the standard active learning framework and then present the strategy used to avoid bad states.

A. Active learning and uncertainty sampling

Active learning [28], [29] is a way to address the problem of ranking a set of unlabeled pixels, U, according to a score providing information about their classification uncertainty. Starting at iteration $\epsilon = 1$ with a training set composed by l labeled pixels, $X^\epsilon = \{\mathbf{x}_i, y_i\}_{i=1}^l$, a supervised model is trained and the pixels in U^ϵ are ranked according to a heuristic accounting for the content in information carried by every unlabeled pixels for the model at the current iteration ϵ. The pixels related to maximal uncertainty are presented to an oracle (a photointerpreter in our case) who labels them, thus discovering their labels. The m newly labeled pixels form a batch $S^\epsilon = \{\mathbf{x}_j, y_j\}_{j=1}^m$ that is added to the training set ($X^{\epsilon+1} = X^\epsilon \cup S^\epsilon$) and removed from

the unlabeled set of candidates ($U^{\epsilon+1} = U \setminus S^{\epsilon}$). As stated in the introduction, many heuristics exist to perform the ranking of the candidates. In this paper, we used two different heuristics:

- The Multi-Class Level Uncertainty criterion proposed in [20], which is a state of the art criterion for uncertainty sampling. This criterion, based on the SVM decision function [30], ranks the pixels by confronting output of the two most confident classes in a One-Against-All setting. For a given iteration t and Ω possible classes, the most uncertain pixel is the one for which

$$\hat{\mathbf{x}}^{\text{MCLU}} = \arg \min_{\mathbf{x}_i \in U} \left\{ f(\mathbf{x}_i)^{\text{MC}} \right\} \tag{1}$$

$$\text{where} \qquad f(\mathbf{x}_i)^{\text{MC}} = \max_{\omega \in \Omega} |f(\mathbf{x}_i, \omega)| - \max_{\omega \in \Omega \setminus \omega^+} |f(\mathbf{x}_i, \omega)| \tag{2}$$

where ω^+ is the class showing maximal confidence, i.e. the argument of the first term of Eq. (2). A high value of this criterion corresponds to samples assigned with high certainty to the most confident class, while a small value represents unreliable classification.

- The Entropy-Query-by-Bagging (EQB) proposed in [9]. This criterion ranks the candidates using a committee of classifiers trained with a subset of the training data X^{ϵ}. Pixels related to maximal entropy in the predictions given by the committee (H^{BAG}) are retained. In this paper, we use the normalized version of the heuristic ($nEQB$) [8]

$$\hat{\mathbf{x}}^{n\text{EQB}} = \arg \max_{\mathbf{x}_i \in U} \left\{ \frac{H^{\text{BAG}}(\mathbf{x}_i)}{\log(N_i)} \right\} \tag{3}$$

where

$$H^{\text{BAG}}(\mathbf{x}_i) = -\sum_{\omega=1}^{N_i} p^{\text{BAG}}(y_i^* = \omega | \mathbf{x}_i) \log \left[p^{\text{BAG}}(y_i^* = \omega | \mathbf{x}_i) \right] \tag{4}$$

$$\text{where} \qquad p^{\text{BAG}}(y_i^* = \omega | \mathbf{x}_i) = \frac{\sum_{m=1}^{k} \delta(y_{i,m}^*, \omega)}{\sum_{m=1}^{k} \sum_{j=1}^{N_i} \delta(y_{i,m}^*, \omega_j)}$$

N_i is the number of classes predicted for pixel $\mathbf{x}_i$ by the committee, with $1 \leq N_i \leq N$. N is the total number of classes. δ is a function returning 1 if the class predicted by the m-th member of the committee is ω_j and 0 otherwise.

Once the ranking of the U set is provided, the user is asked to label the pixels minimizing the criterion in (1) or (3) respectively. This paper only considers these two heuristics because

i) all heuristics based on pixel's uncertainty only produce similar results [8] and ii) including a criterion on pixel diversity would make the queries more effective [20], but do not change the conclusions in the optic of this study. Therefore, we limit the experimentation to these two heuristics.

B. Learning user's confidence

Assuming that the photointerpreter is able to label all the pixels in U makes active learning algorithms efficient tools for semiautomatic training set composition. However, the pixels minimizing Eq. (1) are often difficult to label, since i) they correspond to pixels related to maximal uncertainty ii) they are situated on the border between classes. As a consequence, these pixels often lie on the border between objects in the spatial domain, as observed in the zooms reported in Fig. 1.

These difficult pixels can be considered as bad states, in the sense that the user may be uncertain or agnostic about the response to give for these queries, since their choice does not depend on the ability of the user nor on his/her knowledge of the scene [23].

A user can become frustrated when encountering several bad states, since he/she is forced to use useless resources (he / she cannot provide an answer repeatedly, and the time needed to get the necessary number of labeled pixels is therefore greatly lengthened) at the risk of degrading his/her performances (for example for an increased fatigue). Worse, the user can decide to give advice that is unreliable, thus degrading the model performance with mislabeled training samples.

To decide which states the user would or wouldn't like to encounter, we consider a strategy close to the Confident Execution proposed in [27]: at each iteration, the confidence of the user is assessed and a query minimizing Eq. (1) is presented to him/her only if the confidence about that pixel exceeds a given threshold θ. If the threshold is not met, the query is skipped and the definition of states is upgraded using this negative example. Practically, we train a second model that learns the user's confidence in labeling. This model learns to separate situations where labeling is feasible (with current knowledge) from other where the user is not supposed to be able to provide a label ($Y_\theta = [-1; +1]$). Contrarily to [27], we do not make the difference between states that are unfamiliar (that would correspond to new classes) and merely ambiguous.

To interpret the confidence as a probability and normalize it across iterations of the active learning loop, the outputs of the model are converted into probabilities: this operation is natural

for models as LDA or neural networks, but when using SVM (as in this study) an estimation as the one proposed by Platt [31] has to be used.

Algorithm 1 summarizes the flowchart of the proposed method. Note that in AL-UC there are two training sets: the first is the usual set of the classifier with output space $Y = [1, ..., \Omega]$, while the second is a training set containing the confidence samples and a binary output $Y_\theta = [-1; 1]$. If at the beginning the input samples coincide ($\mathbf{x}^\epsilon = \mathbf{x}_\theta^\epsilon, \epsilon = 1$), they start to diverge as soon as bad states are encountered: in this case, the batch of training examples for the multiclass classifier is not updated, but the uncertain sample is added to the confidence classifier as a negative example (line 16 of the Algorithm) . This way the confidence classifier is constantly updated as long as bad states are encountered. For this reason $|X^\epsilon| \leq |X_\theta^\epsilon|$.

III. DATA AND SETUP

In this section, we present the datasets considered in the experiments, as well as the general experimental setup adopted.

A. Datasets

Two urban VHR images are considered for the experiments (Fig. 2). They describe urban environments at different spatial resolutions, in order to assess differences in confidence of labeling related to the resolution of objects.

- QuickBird "Brüttisellen". The first image is a 4-bands optical image of a residential neighborhood of the city of Zurich named Brüttisellen acquired by the sensor QuickBird in 2002. The image has a size of 329×347 pixels, and a geometrical resolution of 2.4 m. Nine classes of interest have been highlighted by photointerpretation and $40,762$ pixels are available (see Tab I).

- QuickBird "Highway". The second image is another 4-bands optical image of an industrial neighborhood of the city of Zurich acquired by the sensor QuickBird in 2006. The image has a size of 828×889 pixels. The original image was pansharpened using Bayesian Data Fusion [32] to attain a spatial resolution of 0.6 m. Seven classes of interest have been highlighted by photointerpretation and $254,469$ pixels are available (see Tab I).

To account for the spatial context of the pixel, we stacked morphological features [33] to the spectral vector: we added opening and closing features computed on the first PCA extracted on
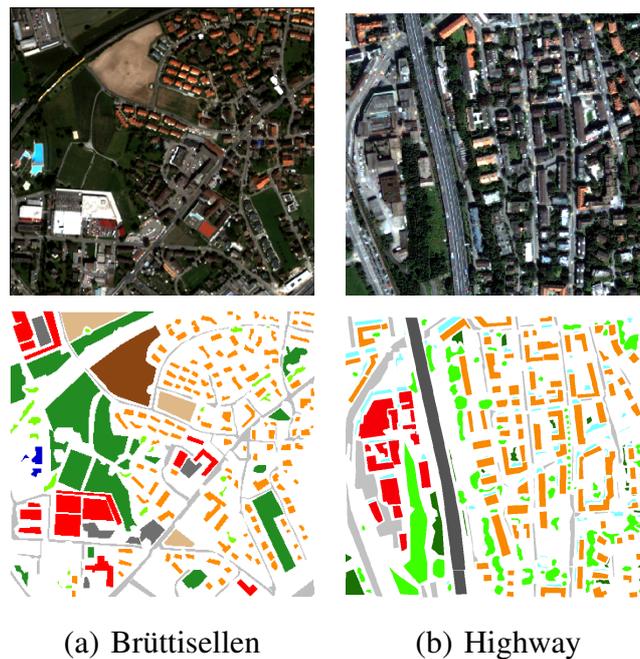
(a) Brüttisellen   (b) Highway

Fig. 2.   Images considered in the experiments, along with their corresponding GTs used for testing purposes.

the multispectral image, as in [34], [35], which is a valid alternative to the use of a panchromatic image (as in [36], [37]). This operation allows to separate land use classes of similar materials but with different spatial extents as, for instance, roads and parking lots. Since the images have different spatial resolutions, the structuring element sizes are $\{1, 3\}$ in radius for the Brüttisellen image and $\{3, 6, 9\}$ for the Highway image. Shape is kept circular in both cases.

The pixels highlighted by photointerpretation compose the test set on which the different approaches are evaluated. Their specific quantities are highlighted in Tab. I.

B. Experimental setup

To test the proposed active learning with user's confidence model we built a MATLAB graphic user interface (Fig. 3), where real users have the task to label the VHR images of Zurich, Switzerland presented above. Contrarily to active learning papers in remote sensing previously published, all the image can be sampled and a human user is performing the labeling.

After initialization, the user is invited to enter an initial training set by photointerpretation. As for their difference in size, 5 pixels per class are queried for the "Brüttisellen" image ($|X^1| = 45$), while 10 pixels per class are requested for the "Highway" image ($|X^1| = 70$). He/she is invited

TABLE I

NUMBER OF LABELED PIXELS USED FOR EVALUATION OF THE BRÜTTISELLEN AND HIGHWAY IMAGES.

Image	Class	GT pixels	Legend color
Brüttisellen	Trees	$1,095$	Light green
	Meadows	$13,123$	Dark green
	Harvested vegetation	$2,523$	Light brown
	Bare soil	$3,822$	Brown
	Residential buildings	$6,746$	Orange
	Commercial buildings	$5,277$	Red
	Asphalt	$6,158$	Light gray
	Parkings	$1,749$	Dark gray
	Pools	269	Blue
Highway	Trees	$52,813$	Light green
	Meadows	$12,347$	Dark green
	Residential buildings	$78,018$	Orange
	Commercial buildings	$25,389$	Red
	Highway	$28,827$	Dark gray
	Asphalt	$43,005$	Light gray
	Shadows	$14,071$	Cyan

to choose these pixels on the image through an interactive window. These samples receive a label by the user and a positive label for the confidence classifier. Then, the active learning process starts.

The user is asked to label the pixels selected by an active learning algorithm into one of the Ω classes of interest ($Y = [1, ..., \Omega]$), detailing the different urban land use types, or into an "unknown class", if the user does not know which label to assign. Every 20 valid labeled pixels (i.e. every 20 answers other than "unknown class") the classifier retrains with the increased training set and produces a new raking of the unlabeled pixels. For EQB, we considered a committee of 10 models, each one using 75% of the available training data drawn randomly from X^ϵ.

As a base classification model, we used a nonlinear SVM with RBF Gaussian kernel. As a lower bound of performance, we used a random selection of locations (RS). Since the first training set X^1 is very small, the RBF kernel parameter is estimated using the median distance
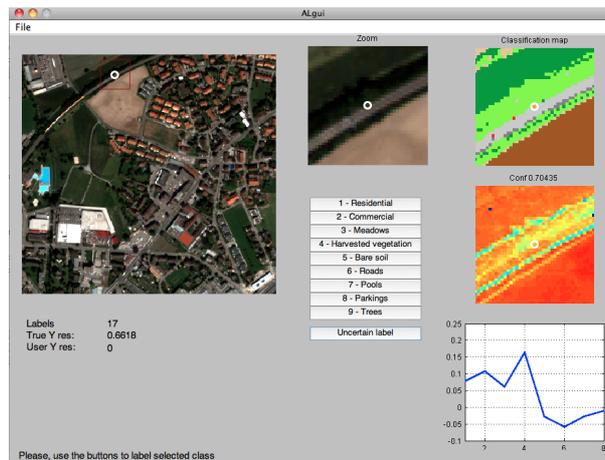
Fig. 3. Graphic user interface developed for testing the AL-UC model. On the left, the image to be labeled; in the middle, class buttons and a zoom; on the right the current classification map, the confidence map and the spectrum of the pixel in the white circle.

between pixels in the image. The Torch library is used for the multiclass SVM [38], which implements the one-against-all (OAA) strategy.

To avoid bad states on the active learning output, we learn the user's confidence by training a second binary SVM classifier with RBF kernel. We consider the probabilistic output using Platt's method [31]. The LibSVM solver is used, as it returns these posterior probabilities. This classifier is trained only since iteration $\epsilon = 2$, because at the first iteration there are no negative samples (X_θ^1 contains only positive samples chosen by the user). To avoid overfitting of specific situations, we kept the search range for the σ kernel bandwidth large in a 4-folds crossvalidation strategy ($\sigma = [10^{-1}, ..., 10^3]$). The threshold θ is fixed at 0.6 after experimental testing: it constitutes a good tradeoff between a filtering that is too strong (which would return uninformative pixels) and one that is too weak (which would be identical to common AL).

In the experiments reported in Sections IV-A and IV-B, a single user performed 5 independent experiments, where he choose the initial training set by clicking on the image. The user knows the image, as well as the task to be performed (i.e. the ground truth). On the contrary, in Section IV-C we considered different users in the labeling task. In this case, we compared the performance of five users, three with experience in remote sensing and labeling tasks, and two who are not familiar with those tasks. Each user performed a single experiment with the three models (random, standard active learning and the proposed AL-UC, both with MCLU as a base

(a) AL criterion   (b) Confidence map   (c) Confidence mask, $\theta = 0.6$
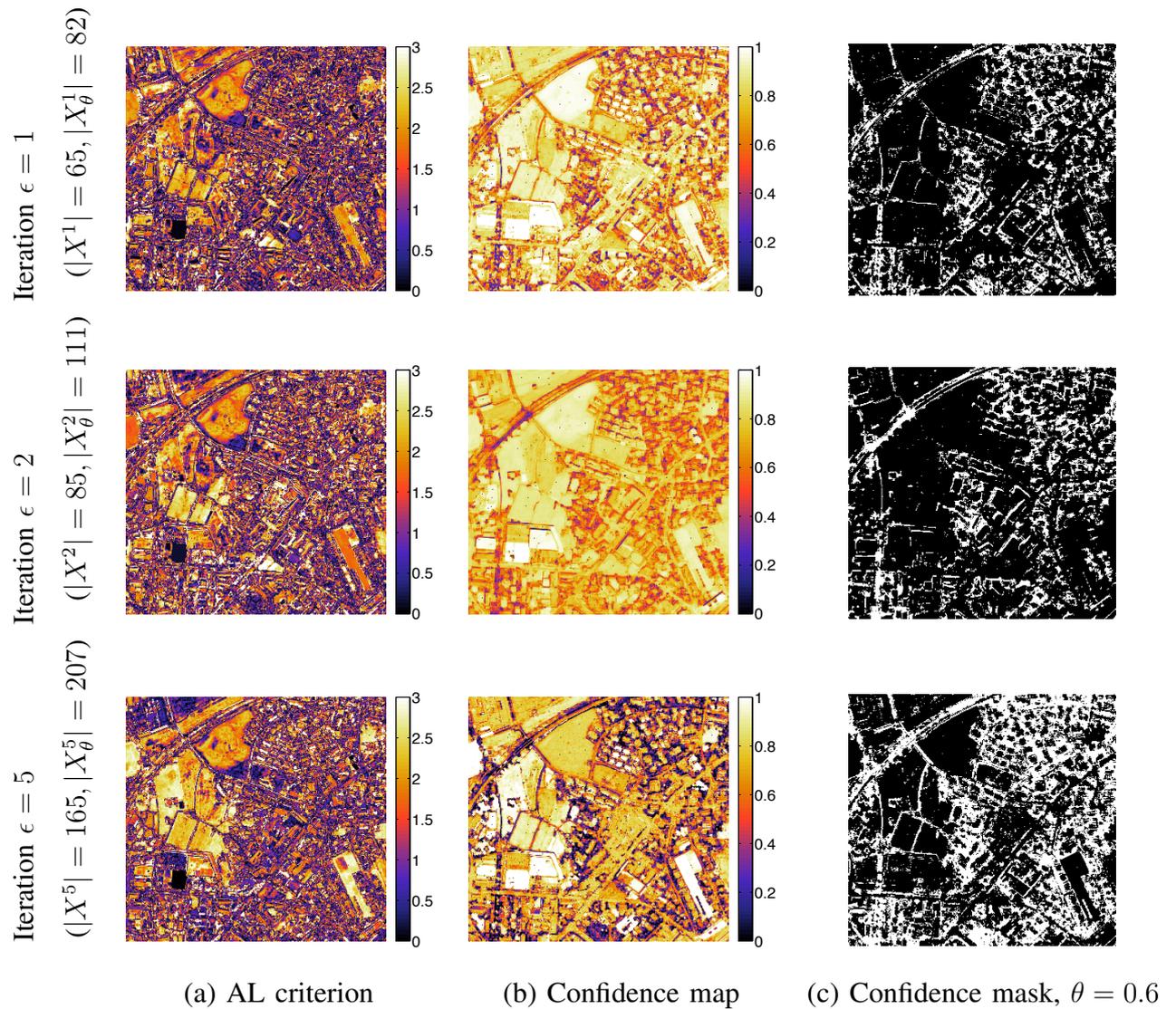
Fig. 4.   AL-UC ingredients. (a) A classical active learning heuristic: MCLU. [Uncertain samples correspond to dark colors] (b) A user's confidence estimation: SVM trained on user's responses [Easily labelizable samples correspond to light colors]. (c) Mask obtained by thresholding the confidence map at $\theta = 0.6$. Only pixels in the black areas can now be selected by minimizing the MCLU criterion.

heuristic). As stated above, the different models are compared in terms of estimated Kappa statistics on the entirety of labeled pixels shown in bottom row of Fig. 2 (quantities in Table I).

IV. RESULTS

This section reports the experimental results obtained on the two case studies.

A. Zurich Brüttisellen

Figure 4 illustrates the basic components of the proposed AL-UC on the Brüttisellen dataset: in the left column, the AL heuristic is reported at iterations 1, 2 and 5. Even if during the iterations there is an increase in confidence on large spatially smooth areas, the heuristic remains fragmented in complex areas and many minima can be seen. Central column of Fig. 4 illustrates the posterior probability of the confidence classifier: during the iterations, this classifier specializes in detecting areas that are easily recognized by the user and not only large, smooth areas (for example, note how the roads become more and more confident, even if they are thin elongated structures).

In Fig. 5, we show two examples of how the confidence map works: in the top example, the railway is easily labeled by the user and result in high confidence, while the train (in white/yellow) is not among the classes to be detected, so it is handled as a bad state with low confidence. In the bottom example, one can appreciate how the linear structures such as roads receive high confidence, as well as the parking lots, which are characterized by high variance (parked cars).

Finally, both sets of information are fused by creating a confidence mask. The confidence
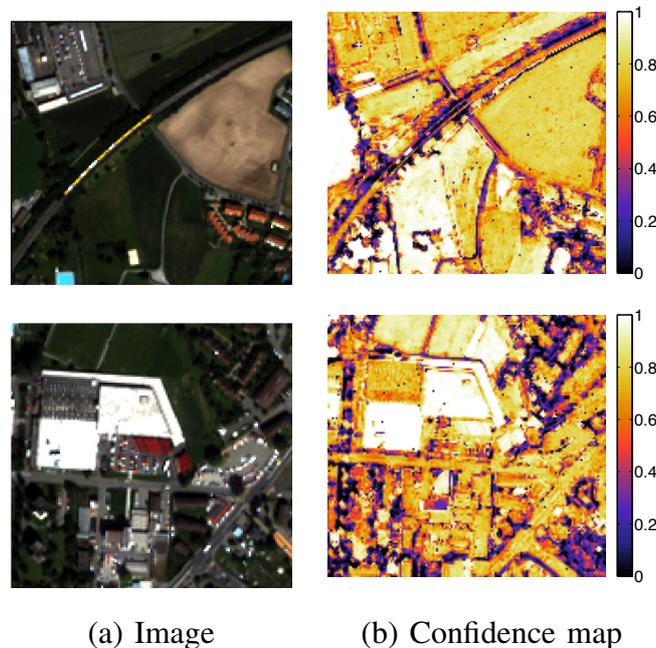

(a) Image   (b) Confidence map

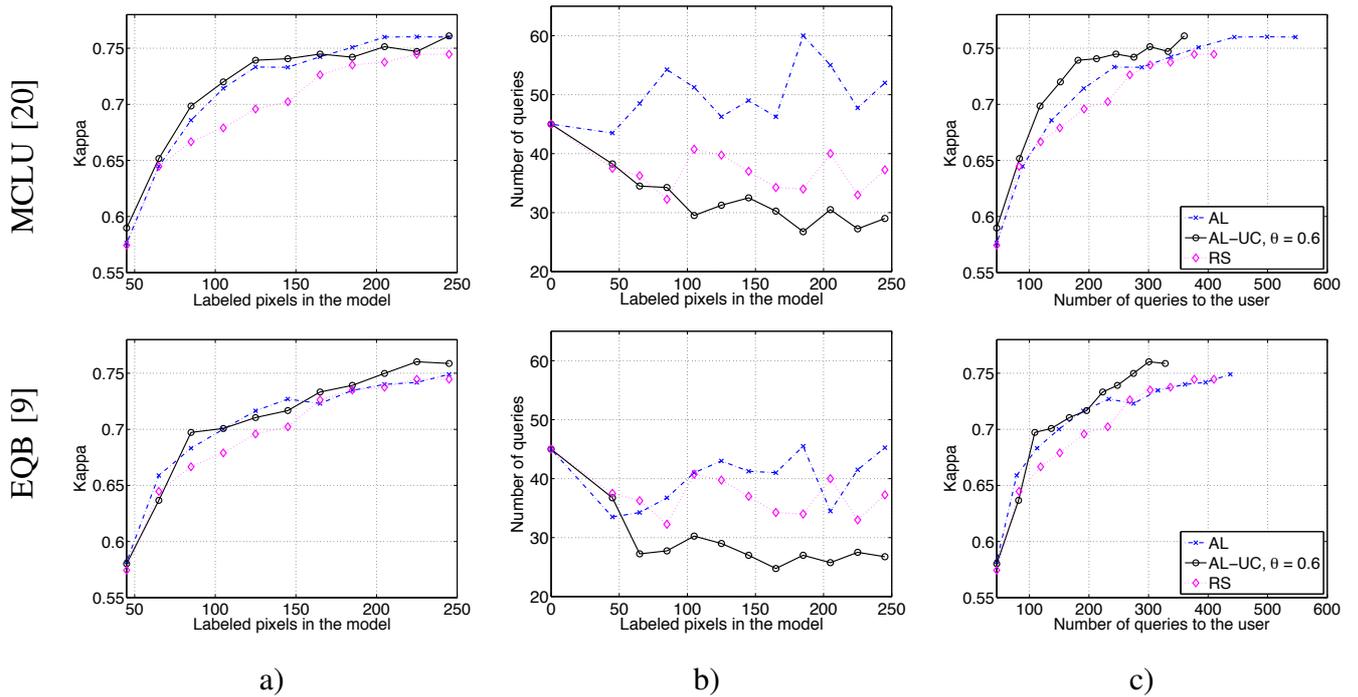Fig. 5.   Zooms into the confidence map of Fig. 4b at iteration $\epsilon = 5$.

Fig. 6.   Numeric results for the Brüttisellen dataset. a) Kappa statistic with traditional active learning setting; b) Number of queries per iteration involving 20 valid labeled samples; c) Kappa statistic related to average real effort provided by the user.

maps of Fig. 4(b) are simply thresholded at a level of confidence θ and only the pixels for those $p((y_{\theta,i} = +1|\mathbf{x})) > \theta$ become presentable to the user. This way, the model continues to rank the pixels according to the active learning criterion, but only areas supposed to be easily understood by the user are made visible (in red in Fig. 4(c)).

Numerical results on the Brüttisellen image are reported in Fig. 6 for the three methods considered and the two heuristics tested. The left-hand panel (Fig. 6(a)) illustrates the numerical performance at the end of each iteration. At first glance, the proposed AL-UC seems not to outperform the standard AL.

These observations would be correct considering an omniscient user who can always label the pixels queried by the model (i.e. if 20 labeled pixels could be obtained with 20 queries per iteration). However, as illustrated in Fig. 1, the traditional active learning method often highlights border pixels that the user cannot label: after initialization, where 45 pixels are queried by each method, the user needs to consider, on average, 45 to 50 pixels to provide 20 labels at every iteration (Fig. 6(b)). Random sampling is much more efficient in this sense, since the user needs

around 35 queries to label 20 pixels (40% of the queries were feasible for the user). Finally, the proposed AL-UC was the most efficient, since only 25 to 30 queries are necessary to obtain the 20 labels. This shows that the second classifier has learned the confidence of the user, as it queries more useful pixels than the random strategy.

This leads to a rescaling of the curves of Fig. 6(a) into a more realistic picture, that is illustrated in Fig. 6(c): the performance is plotted as a function of the real effort provided by the user, i.e., the total number of queries (both successful and unsuccessful). This figure shows that the proposed method has a steeper learning phase than the traditional active learning method, since the latter wastes several queries on pixels too difficult to label. When taking a performance objective of 0.75 in κ, the proposed method needed on the average 432 queries to reach it, while traditional active learning needed one hundred additional queries.

By looking back at the confidence maps in Fig. 4(b), we can also observe that during the iterations the confidence is spread in all the areas that are easy to label: regular areas, like buildings, bare soil or very contrasted structures, such as roads. These maps also illustrate that the threshold of confidence θ must not be chosen too high, since a risk of getting stuck on a single object in the first iterations increase: in this case, setting $\theta = 0.9$ would have focused all the sampling in a few areas such as the commercial area in the bottom-left corner, thus trapping the solution in a local maximum.

Regarding comparison with the EQB heuristic (reported in the bottom row of Fig. 6) similar trends are observed. The learning curve is less steep, showing the adequacy of MCLU for active learning with SVM (the heuristic uses the decision function directly, while EQB works with

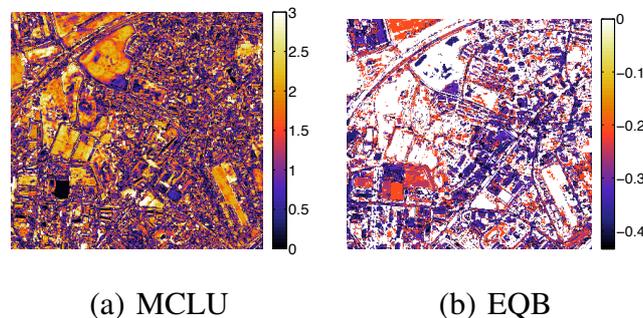(a) MCLU   (b) EQB

Fig. 7. Comparison of the uncertainty maps returned by MCLU and EQB (multiplied by a -1 factor). Dark areas correspond to uncertain areas.

committees of models), but in general both reach a similar performance after 200 valid queries. However, EQB seems to require less queries to find the 20 valid samples, at each iteration and independently from the approach tested. This can be explained by the differences in the nature of the two heuristics (Fig. 7): MCLU ranks pixels according to the (continuous) decision function obtained on the two most probable classes; this means that there are potentially as many different values as candidates in U^ϵ. We saw that, on the average, 50% of the samples minimizing such function are easy to label. On the contrary, the EQB heuristic depends on a committee of predictions and the heuristic only has a limited number of possible entropy values (depending on the number of classes Ω, the number of models k and the number of classes predicted by the committee N_i (see Eq. (3)). Therefore, the EQB function is a quantized function with few distinct values[1]. As a consequence, several pixel candidates receive the maximal entropy value and the choice is then done randomly among those with maximal entropy. Consequently with EQB, the chances to query a pixel that the user can label increase.

Figure 8 illustrates the importance of finding a good model to describe the confidence. It shows the crossvalidation surface of the parameter space of the confidence classifier at iteration $\epsilon = 4$ for a given run of the AL-UC algorithm. The green circle highlights the area of solutions capable of describing the confidence of the user. A good selection of parameters results in confidence maps that successfully constrain the AL heuristic (in green/solid), whereas a bad choice of parameters leads to confidence maps that do not constrain the AL heuristic (in red/dashed).

B. Zurich highway

Experimental results on the Highway dataset are reported in Fig. 9 for the three methods considered. Given the higher resolution, the difference between the compared methods is expected to be lower, since a higher proportion of uncertain samples are expected to be within the objects of interest (for instance chimneys on roofs). This is due to the increased intraclass variance of the classes in the Highway dataset, that has a spatial resolution four times higher (0.6 m for "Highway" vs. 2.4 m for "Brüttisellen"). As a consequence, the ambiguous areas are more

---

[1]In the case of the experiments reported, the number of partitions P (corresponding to different entropy values) achievable by a committee of n models predicting K classes is $P(n, K) = \sum_{k=1}^{K} P(n, k)$. To compute this quantity, we use the following three properties: i) $P(n, k) = P(n - 1, k - 1) + P(n - k, k)$; ii) $P(n, k) = 0$ if $n < k$; iii) $P(n, n) = P(n, 1) = 1$. Using the recursive formula in i), the maximum number of entropy values is $P(10, 9) = \sum_{k=1}^{9} P(10, k) = 41$.
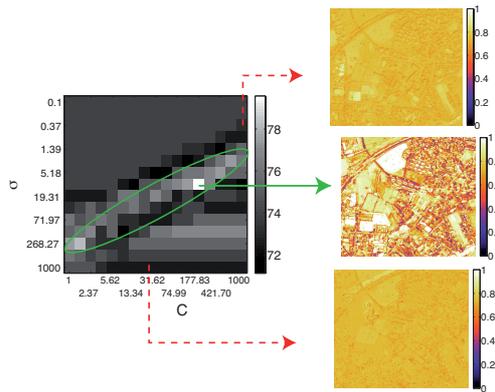
Fig. 8. Crossvalidation surface showing the overall accuracy of the confidence classifier for the Brüttisellen dataset. On the right, confidence maps corresponding to the maximum (in green/solid) and two minima (in red/dashed).
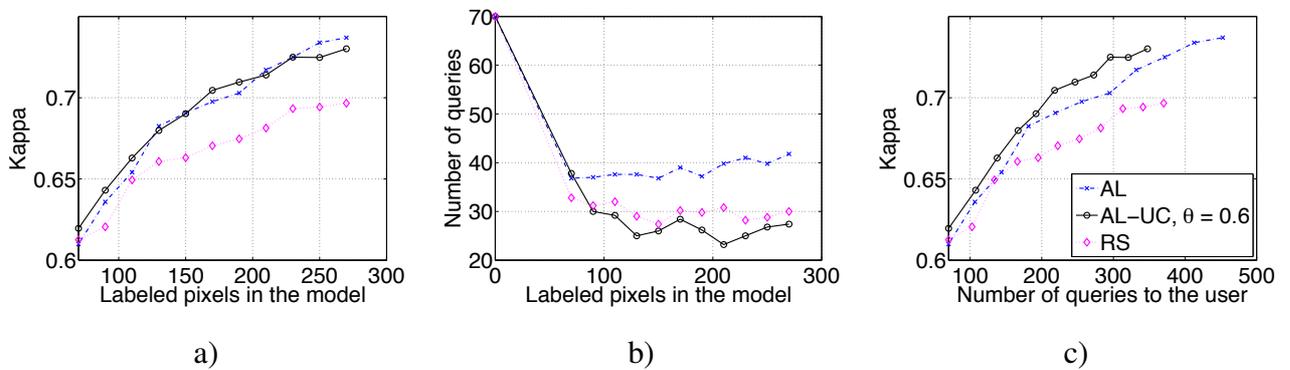

a)

b)

c)

Fig. 9. Numeric results for the Highway dataset using the MCLU heuristic. a) Kappa statistic with traditional active learning setting; b) Number of queries per iteration involving 20 valid labeled samples; c) Kappa statistic related to average real effort provided by the user.

limited in space (Fig. 10), since the traditional AL tends to ask pixels that are anomalous, but within the urban objects: in this case, the user can respond more easily to the query, since the object itself is easily identifiable.

Curves reported in Fig. 9 confirm this intuition: after initialization, where 70 pixels are queried for each method, AL-UC performs similarly to classical active learning, but AL requires on average 10 additional queries per iteration to obtain the 20 labeled pixels. On average, AL-UC needs 25 to 30 queries (similar to RS), while AL requires about 40. The right-hand plot illustrates the effort demanded to the used against the performance and, as for the previous dataset, AL-UC allows to retrieve higher accuracy with less queries.
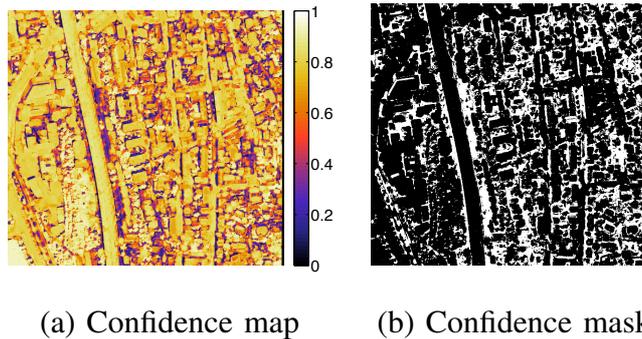
(a) Confidence map   (b) Confidence mask

Fig. 10.   (a) Confidence map and (b) corresponding binary mask for the Highway dataset at iteration $\epsilon = 5, |X^5| = 170, |X_\theta^5| = 218$.
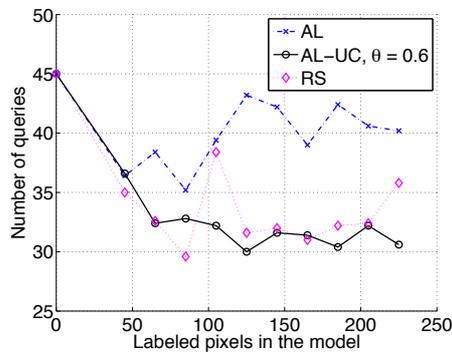

Fig. 11.   Average number of queries for a committee of different users performing a single run of each model.

C. Considering a committee of users

In this section, we consider a committee of five users, each one performing one experiment on the Brutisellen image with the three models. Three of them are trained remote sensing analysts, while the others two are not familiar with labeling tasks. Figure 11 illustrates the average number of queries required by the four users. The tendency observed in the single-user experiment presented above are confirmed. This shows that the proposed method efficiently evaluates the confidence of the user in labeling, avoiding bad states and significantly reducing the number of queries.

V. CONCLUSIONS

In this paper, we studied a way to learn the confidence of a photointerpreter to make active learning routines effective in real-life scenarios. For the first time, an active learning method is

assessed on the entirety of an image and shortcomings related to the uncertainty of a signal are put in relation with the capacity of a user to provide a reliable label for the pixel. By assessing the probability of having a bad state (a pixel that the photointerpreter is unable to label), the most uncertain pixels ranked by active learning are filtered, thus presenting to the user a set that he/she is capable of labeling. Experiments on two QuickBird images at different spatial resolutions showed the efficiency of the method, which significantly decreased the number of queries that the user must provide to fulfill his task.

This work opens a wide range of possible studies for operative active learning: the effects of the user must be studied, as well as the effect of adapting the threshold θ along the iterations (see [27]). If the latter is more a technical question to be tackled in the future, the first opens avenues related to crowdsourcing [39], [40] and community-based online learning of surface signatures.

ACKNOWLEDGEMENTS

REFERENCES

[1] P. J. Pinter, J. L. Hatfield, J. S. Schepers, E. M. Barnes, S. Moran, C. S. T. Daughtry, and D. R. Upchurch, "Remote sensing for crop management," *Photogramm. Eng. Rem. S.*, vol. 69, no. 6, pp. 647–664, 2003.

[2] W. A. Dorigo, R. Zurita-Milla, A.J.W. de Wit, J. Brazile, R. Singh, and M. Schaepman, "A review on reflective remote sensing and data assimilation techniques for enhanced agroecosystem modeling," *Int. J. Appl. Earth Obs. Geoinf.*, vol. 9, pp. 166–193, 2007.

[3] H. Taubenbock, T. Esch, M. Wiesner, A. Roth, and S. Dech, "Monitoring urbanization in mega cities from space," *Remote Sensing of Environment*, vol. 117, pp. 162–176, 2012.

[4] G. Camps-Valls, D. Tuia, L. Gómez-Chova, S. Jimenez, and J. Malo, *Remote Sensing Image Processing*, Synthesis Lectures on Image, Video, and Multimedia Processing. Morgan and Claypool, 2011, Available at: *http://www.morganclaypool.com/toc/ivm/5/1*.

[5] S. Prasad, L. Bruce, and J. Chanussot, *Optical Remote Sensing*, Advances in Signal Processing and Exploitation Techniques. Springer, 2011.

[6] G. M. Foody and A. Mathur, "Toward intelligent training of supervised image classifications: directing training data acquisition for SVM classification," *Remote Sensing of Environment*, vol. 93, no. 1-2, pp. 107–117, 2004.

[7] G. M. Foody and A. Mathur, "The use of small training sets containing mixed pixels for accurate hard image classification: training on mixed spectral responses for classification by a SVM," *Remote Sensing of Environment*, vol. 103, no. 2, pp. 179–189, 2006.

[8] D. Tuia, M. Volpi, L. Copa, M. Kanevski, and J. Muñoz-Marí, "A survey of active learning algorithms for supervised remote sensing image classifications:," *IEEE J. Sel. Topics Signal Proc.*, vol. 5, no. 3, pp. 606–617, 2011.

[9] D. Tuia, F. Ratle, F. Pacifici, M. Kanevski, and W.J. Emery, "Active learning methods for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2218–2232, 2009.

[10] W. Di and M. M. Crawford, "Active learning via multi-view and local proximity co-regularization for hyperspectral image classification," *IEEE J. Sel. Topics Signal Proc.*, vol. 5, no. 3, pp. 618–628, 2011.

[11] W. Di and M. M. Crawford, "View generation for multiview maximum disagreement based active learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, in press.

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

[13] E. Pasolli, F. Melgani, and Y. Bazi, "SVM active learning through significance space construction," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 3, pp. 431 – 435, 2011.

[14] S. Patra and L. Bruzzone, "A fast cluster-assumption based active-learning technique for classification of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 5, pp. 1617–1626, 2011.

[15] J. Li, J.M. Bioucas-Dias, and A. Plaza, "Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 11, pp. 4085 –4098, 2010.

[16] D. Tuia, E. Pasolli, and W. J. Emery, "Using active learning to adapt remote sensing image classifiers," *Remote Sensing of Environment*, vol. 115, pp. 2232–2242, 2011.

[17] Jun Li, J.M. Bioucas-Dias, and A. Plaza, "Hyperspectral image segmentation using a new bayesian approach with active learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3947 –3960, 2011.

[18] D. Tuia, J. Muñoz-Marí, and G. Camps-Valls, "Remote sensing image segmentation by active queries," *Pattern Recognition*, vol. 45, no. 6, pp. 2180–2192, 2012.

[19] J. Muñoz-Marí, D. Tuia, and G. Camps-Valls, "Semisupervised segmentation of remote sensing images with active queries," *IEEE Transaction on Geoscience and Remote Sensing*, in press.

[20] B. Demir, C. Persello, and L. Bruzzone, "Batch mode active learning methods for the interactive classification of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 3, pp. 1014–1032, 2011.

[21] M. Volpi, D. Tuia, and M. Kanevski, "Memory-based cluster sampling for remote sensing image classification," *IEEE Transactions on Geoscience and Remote Sensing*, in press.

[22] C. Persello and L. Bruzzone, "A novel active learning strategy for domain adaptation in the classification of remote sensing images," in *Proc. IEEE IGARSS 2011, Vancouver, Canada*, July 2011, pp. 3720–3723.

[23] K. Judah, A. Fern, and T. Diettrich, "Active imitation learning via state queries," in *Intl. Conf. Mach. Learn. ICML, Workshop on Combining Learning Strategies to Reduce Label Cost*, Bellevue, WA, USA, 2011.

[24] T. Inamura, M. Inaba, and H. . Inoue, "Acquisition of probabilistic behavior decision model based on the interactive teaching method," in *Proceedings of the Ninth International Conference on Advanced Robotics*, 1999, pp. 523–528.

[25] M. N. Nicolescu, *A framework for learning from demonstration, generalization and practice in human-robot domains*, Ph.D. thesis, University of Southern California, 2003.

[26] D. Grollman and O. Jenkins, "Dogged learning for robots," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 2483–2488.

[27] S. Chernova and M. Veloso, "Interactive policy learning through confidence-based autonomy," *J. Artificial Intelligence Res.*, vol. 2009, pp. 1–25, 34.

[28] D. Cohn, L. Atlas, and R. Ladner, "Improving generalization with active learning," *Mach. Learn.*, vol. 15, no. 2, pp. 201–221, 1994.

[29] B. Settles, "Active learning literature survey," Computer Sciences Technical Report 1648, University of Wisconsin-Madison, 2010.

[30] B. Boser, I. Guyon, and V. Vapnik, "A training algorithm for optimal margin classifiers," in *5th ACM Workshop on Computational Learning Theory*, Pittsburgh, USA, 1992, pp. 144–152.

[31] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in large margin classifiers*, pp. 61–74. MIT press, 1999.

[32] D. Fasbender, J. Radoux, and P. Bogaert, "Bayesian data fusion for adaptable image pansharpening," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 6, pp. 1847–1857, 2008.

[33] P. Soille, *Morphological image analysis*, Springer-Verlag, Berlin-Heidelberg, 2004.

[34] J.A. Benediktsson, J. A. Palmason, and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 480–490, 2005.

[35] G. Licciardi, F. Pacifici, D. Tuia, S. Prasad, T. West, F. Giacco, J. Inglada, E. Christophe, J. Chanussot, and P. Gamba, "Decision fusion for the classification of hyperspectral data: Outcome of the 2008 GRS-S data fusion contest," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 11, pp. 3857–3865, 2009.

[36] N. Longbotham, C. Chaapel, L. Bleiler, C. Padwick, W. J. Emery, and F. Pacifici, "Very high resolution multiangle urban classification analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 4, pp. 1155–1170, 2012.

[37] M. Dalla Mura, J. Atli Benediktsson, B. Waske, and L.; Bruzzone, "Morphological attribute profiles for the analysis of very high resolution images," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 10, pp. 3747–3762, 2010.

[38] R. Collobert, S. Bengio, and J. Mariéthoz, "Torch: a modular machine learning software library," Tech. Rep. RR 02-46, IDIAP, 2002.

[39] J. Abernethy and R. Frongillo, "A collaborative mechanism for crowdsourcing prediction problems," in *Advances in Neural Information Processing Systems (NIPS)*, 2011.

[40] R. Gomes, P. Welinder, A. Krause, and P. Perona, "Crowdclustering," in *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2011.

Devis Tuia (SM'07, M'09) was born in Mendrisio, Switzerland, in 1980. He received a diploma in Geography at the University of Lausanne (UNIL) in 2004, a Master of Advanced Studies in Environmental Engineering at the Federal Institute of Technology of Lausanne (EPFL) in 2005 and a Ph.D in environmental sciences at UNIL in 2009. He was a postdoc researcher at both the University of Valéncia, Spain and the Univeristy of Colorado at Boulder under a Swiss National Foundation program. He is now senior research associate at the LaSIG laboratory, EPFL. His research interests include the development of algorithms for information extraction and classification of very high resolution remote sensing images using machine learning algorithms. Visit http://devis.tuia.googlepages.com/ for more information.

Jordi Muñoz-Marí (M'11) was born in València, Spain in 1970, and received a B.Sc. degree in Physics (1993), a B.Sc. degree in Electronics Engineering (1996), and a Ph.D. degree in Electronics Engineering (2003) from the Universitat de València. He is currently an Associate Professor in the Electronics Engineering Department at the Universitat de València, where teaches Electronic Circuits and, Programmable Logical Devices, Digital Electronic Systems and Microprocessor Electronic Systems. His research interests are mainly related with the development of machine learning algorithms for signal and image processing. Visit http://gpds.uv.es/~jordi/ for more information.

Algorithm 1 AL-UC algorithm

Inputs

- Initial training set $X^\epsilon = \{\mathbf{x}_i, y_i\}_{i=1}^{l}$

  ($X \in \mathbb{R}^d$, $Y \in [1, ..., \Omega]$, $\epsilon = 1$, $\epsilon = $ iteration).

- Initial confidence training set $X_\theta^\epsilon = \{\mathbf{x}_i, \mathbf{1}_i\}_{i=1}^{l}$

  ($X \in \mathbb{R}^d$, $\epsilon = 1$).

- Pool of candidates $U^\epsilon = \{\mathbf{x}_i\}_{i=l+1}^{l+u}$ ($U \in \mathbb{R}^d$, $\epsilon = 1$).

- Number of pixels m to add at each iteration

  (defining the size of the batch of selected pixels S).

1: **repeat**

2:     Train the classifier with current training set X^ϵ;

3:     Train the confidence classifier with current X_θ^ϵ;

4:     **for** each candidate in U^ϵ **do**

5:         Evaluate the active learning *heuristic*;

6:         Assess assignment's confidence $p(y_\theta = +1|X_\theta^\epsilon)$;

7:     **end for**

8:     Rank the candidates in U^ϵ according to the score of the heuristic, obtain ranking r;

9:     **repeat**

10:         Select next candidate in r, $\mathbf{x}_r$;

11:         **if** $p(y_{\theta,r} = +1|X_\theta^\epsilon) > \theta$ **then**

12:             **if** the user can provide the label y_r **then**

13:                 Add the labeled candidate to the batch $S^\epsilon = S\epsilon \cup \{\mathbf{x}_r, y_r\}$;

14:                 Add the positive example to the confidence training set $X_\theta^\epsilon = X_\theta^\epsilon \cup \{\mathbf{x}_r, 1\}$;

15:             **else**

16:                 Add the negative example to the confidence training set $X_\theta^\epsilon = X_\theta^\epsilon \cup \{\mathbf{x}_r, -1\}$;

17:             **end if**

18:         **else**

19:             Add the negative example to the confidence training set $X_\theta^\epsilon = X_\theta^\epsilon \cup \{\mathbf{x}_r, -1\}$;

20:         **end if**

21:     **until** Batch S has m candidates

22:     Add the batch to the training set $X^{\epsilon+1} = X^\epsilon \cup S^\epsilon$;

23:     Remove the batch from the pool of candidates $U^{\epsilon+1} = U^\epsilon \backslash S^\epsilon$;

24:     $\epsilon = \epsilon + 1$.

25: **until** a stopping criterion is met.