Weakly Supervised Classification of Objects in Images Using Soft Random Forests

Riwal Lefort[1,2], Ronan Fablet[2], and Jean-Marc Boucher[2]

[1] Ifremer/STH, Technopole Brest Iroise, 29280 Plouzane
[2] Telecom-Bretagne/LabSticc, Technopole Brest Iroise, 29280 Plouzane
{riwal.lefort,ronan.fablet,jm.boucher}@telecom-bretagne.eu

Abstract. The development of robust classification model is among the important issues in computer vision. This paper deals with weakly supervised learning that generalizes the supervised and semi-supervised learning. In weakly supervised learning training data are given as the priors of each class for each sample. We first propose a weakly supervised strategy for learning soft decision trees. Besides, the introduction of class priors for training samples instead of hard class labels makes natural the formulation of an iterative learning procedure. We report experiments for UCI object recognition datasets. These experiments show that recognition performance close to the supervised learning can be expected using the propose framework. Besides, an application to semi-supervised learning, which can be regarded as a particular case of weakly supervised learning, further demonstrates the pertinence of the contribution. We further discuss the relevance of weakly supervised learning for computer vision applications.

1 Introduction

The paper focuses on weakly supervised learning that includes both the supervised and semi-supervised learning. It considers probability vectors that indicate the prior of each class for each sample of the training set. The same notations are used throughout the paper. Let $\{x_n, \pi_n\}_n$ be the training dataset, where x_n is the feature vector for sample n and $\pi_n = \{\pi_{ni}\}_i$ is the prior vector for sample n, i indexing the classes. π_{ni} gives the likelihood for the example x_n to belong to class i. A supervised case corresponds to priors π_{ni} set to 0 or 1 whether or not sample n belongs to class i. The semi-supervised learning is also a specific case where training priors π_{ni} are given as 0 or 1 for the subset of the fully labelled training samples and as uniform priors for the remaining unlabelled samples.

Weakly supervised learning covers several cases of interest. Especially, in image and video indexing issue, object recognition dataset may involve training images labelled with the presence or the absence of each object category [1] [2] [3] [4] [5] [6]. Again such presence/absence dataset can be regarded as specific cases of training priors. As an other example, one can imagine an annotation by experts with some uncertainty measure [7]. This situation would be typical from photo-interpretation applications especially remote sensing applications [8].

K. Daniilidis, P. Maragos, N. Paragios (Eds.): ECCV 2010, Part IV, LNCS 6314, pp. 185–198, 2010.

A further generalization can be issued from expert-driven or prior automated analysis providing some confidence or uncertainty measure of the classification of objects or group of objects. This is typical from remote sensing applications. For instance, in the acoustics sensing of the ocean for fisheries management [9], images of fish schools are labelled with class proportions that lead to individual priors for each fish school. Such training dataset provided with class priors instead of hard class labels could also be dealt with when a cascade of classifiers or information could be processed before the final decision. In such cases, one could benefit from soft decisions to keep all relevant information in the cascade until the final decisions. This is typical of computer vision applications where multiple sources of information may be available [10] [11].

In this paper, we address weakly supervised learning using decision trees. The most common probabilistic classifiers are provided by generative models learnt using Expectation Maximization algorithm [12] [13]. These generative probabilistic classifiers are also used in ensemble classifiers [14] as in boosting schemes or with iterative classifiers [11]. In contrast, we investigate the construction of decision trees with weakly supervised dataset. Decision tree and random forest are among the most flexible and efficient techniques for supervised object classification [15]. However to our knowledge, previous works only deal with decision trees that consider hard input and probabilistic output [16]. The second contribution of this work is to develop an iterative procedure for weakly supervised learning. The objective is to iteratively refine the class priors of the training data in order to improve the classification performance of the classifier at convergence. We report different experiments to demonstrate the relevance of these contributions. The focus is given to examples demonstrating the genericity of the proposed weakly supervised framework, including applications to semi-supervised learning. In all cases, the proposed approach favourably compares to previous work, especially hard decision trees, generative classification models, and discriminative classification models.

This paper is organized as follows. In section 2, we present the weakly supervised learning of decision trees. In section 3, the iterative procedure for weakly supervised learning is detailed. The application to semi-supervised learning is presented in section 4 while experiments and conclusions are given in sections 5 and 6.

2 Decision Trees and Random Forest

2.1 Supervised Decision Trees and Random Forests

In supervised learning, the method consists in splitting the descriptor space into sub-sets that are homogeneous in terms of object classes. More precisely, the feature space is split based on the maximization of the gain of information. Different split criteria have been proposed such as the Gini criterion [17], the Shannon entropy [18] [19], or other on statistical tests such as ANOVA [20] or χ^2 test [21]. All of these methods have shown to lead to rather equivalent classification performances.

We focus here on the C4.5 decision trees which are among the most popular [19]. During the training step, at a given node of the tree, the procedure chooses descriptor d and associated split value S_d that maximize information gain G:

$$\arg\max_{\{d,S_d\}} G(S_d) \tag{1}$$

where gain G is issued from the Shannon entropy of object classes [19]:

$$\begin{cases} G = \left(\sum_m E^m\right) - E^0 \\ E^m = -\sum_i p_{mi} log(p_{mi}) \end{cases}. \tag{2}$$

where E^0 indicates the entropy at the parent considered node, E^m the entropy at children node m, and p_{mi} the likelihood of class i at node m.

A test sample is passed trough the tree and follows the test rules associated with each node. It is assigned to the class of the terminal node (or descriptor subspace) that it reaches.

Random forests combine a "bagging" procedure [22] and the random selection of a subset of descriptors at each node [23]. The random forest [15] can provide probabilistic outputs given by the posterior distribution of the class votes over the trees of the forest. Additional randomization-based procedures can be applied during the construction of the tree [24]. In some cases, they may lead to improve performances. Here, the standard random forests will be considered [15].

2.2 Weakly Supervised Learning of Soft Decision Trees

In this section, we introduce a weakly supervised procedure for learning soft decision trees. Let us denote by $\{x_n, \pi_n\}$ the provided weakly supervised dataset.

In contrast to the standard decision tree, any node of the tree is associated with class priors. In the weakly supervised setting, the key idea is to propagate class priors through tree nodes rather than class labels as in the supervised case. Consequently, given a constructed decision tree, a test sample will be passed trough the tree and be assigned the class priors of the terminal it will reache.

Let us denote by p_{mi} the class priors at node m of the tree. The key aspect of the weakly supervised learning of the soft decision tree is the computation of class prior p_{mi} at any node m. In the supervised case it consists in evaluating the proportion of each class at node m. In a weakly supervised learning context, real classes are unknown and class proportions can not be easily assessed. We propose to compute p_{mi} as a weighted sum over priors $\{\pi_{ni}\}$ for all samples attached to node m. For descriptor d, denoting x_n^d the instance value and considering the children node m_1 that groups together data such as $\{x_n^d\} < S_d$, the following fusion rule is then proposed:

$$p_{m_1 i} \propto \sum_{\{n\}|\{x_n^d\}<S_d} (\pi_{ni})^\alpha \tag{3}$$

For the second children node m_2 that groups data such as $\{x_n^d\} > S_d$, the equivalent fusion rule is suggested:

$$p_{m_2 i} \propto \sum_{\{n\}|\{x_n^d\}>S_d} (\pi_{ni})^{\alpha} \tag{4}$$

The considered power α weighs low-uncertainty samples, i.e. samples such that class priors closer to 1 should contribute more to the overall cluster mean p_{mi}. An infinite exponent values resorts to assigning the class with the greatest prior over all samples in the cluster. In contrast, an exponent value close to zero withdraws from the weighted sum low class prior. In practice, for α from 0.1 to 8, performances are more or less the same accuracy. After experiments, α is set to 0.8. This setting comes to give more importance to priors close to one. If $\alpha < 1$, high class priors are given a similar greater weight compared to low class priors. If $\alpha > 1$, the closer to one the prior the greater the weight.

Considering a random forest, the output from each tree t for a given test data x is a prior vector $p_t = \{p_{ti}\}$. p_{ti} is the prior for class i at the terminal node reached for tree t. The overall probability that x is assigned to class i, i.e. posterior likelihood $p(y = i|x)$, is then given by the mean:

$$p(y = i|x) = \frac{1}{T} \sum_{t=1}^{T} p_{ti} \tag{5}$$

where $y_n = i$ denotes that sample x_n is assigned to class i. A hard classification resorts to selecting the most likely class according to posteriors (5).

In this paper, experiments are carried out to fix the mean optimal number of tree per forest. Results show that 100 trees per forests are optimal on average. Furthermore, following the random forest process, there is no pruning.

3 Iterative Classification

3.1 Naive Iterative Procedure

The basic idea of the iterative scheme is that the class priors of the training samples can be refined iteratively from the overall knowledge acquired by the trained classifier such that these class priors finally converge to the real class of the training samples. The classifier at a given iteration can then be viewed as a filter that should reduce the noise or uncertainty on the class priors of training samples. Note that this iterative method is only applied to the training dataset.

Such an iterative procedure has previously been investigated in different contexts, especially with probabilistic generative classifier [11]. Theoretical results regarding convergence properties can hardly be derived [25] [26], though good experimental performances have been reported [27]. The major drawbacks of this approach are possible over-training effects and the propagation of early classification errors [28]. Bayesian models may contribute to solve for these over-training issues.

Table 1. Naive iterative procedure for weakly supervised learning (IP1)

Given an initial training data set $T_1 = \{x_n, \pi_n^1\}$ and M iterations,

1. For m from 1 to M
   - Learn a classifier C_m from T_m.
   - Apply the classifier C_m to T_m.
   - Update $T_{m+1} = \{x_n, \pi_n^{m+1}\}$ with $\pi_n^{m+1} \propto \pi_n^1 p(x_n|y_n = i, C_m)$.
2. Learn the final classifier using T_{M+1}.

The implementation of this naive iterative procedure proceeds as follows for weakly supervised learning. At iteration m given the weakly supervised dataset $\{x_n, \pi_n^m\}$, a random forest C_m can be learnt. The updated random forest could be used to process any training sample $\{x_n, \pi_n^m\}$ to provide an updates class prior π^{m+1}. This update of class prior π^{m+1} should exploit both the output of the random forest and the initial prior π^1. Here, the updated priors are given by: $\pi_n^{m+1} \propto \pi_n^1 p(x_n|y_n = i, C_m)$ where $y_n = i$ denotes the classe variable for sample n.

This algorithm is sketched in Tab. 1. In the subsequent, this procedure will be referred to as IP1 (Iterative Procedure 1).

3.2 Randomization-Based Iterative Procedure without over Training

A major issue with the above naive iterative procedure is that the random forest is repeatedly applied to the training data such that over-training effects may be expected. Such over-training effects should be avoided.

To this end, we propose a second iterative procedure. The key idea is to exploit a randomization-based procedure to distinguish at each iteration separate training and test subsets. More precisely, we proceed as follows. At iteration m, the training dataset $T_m = \{x_n, \pi_n^m\}$ is randomly split into a training dataset Tr_m and a test dataset Tt_m according to a given proportion β. Tr_m is exploited to build a weakly supervised random forest C_m. Samples in Tt_m are passed

Table 2. Randomization-based iterative procedure for weakly supervised learning (IP2)

Given a training data set $T_1 = \{x_n, \pi_n^1\}$ and M iterations,

1. for m from 1 to M
   - Randomly split T_m in two groups: $Tr_m = \{x_n, \pi_n^m\}$ and $Tt_m = \{x_n, \pi_n^m\}$ according to a split proportion β.
   - Learn a classifier C_m from subset Tr_m.
   - Apply classifier C_m to subset Tt_m.
   - Update $Tt_{m+1} = \{x_n, \pi_n^{m+1}\}$ with $\pi_n^{m+1} \propto \pi_n^1 p(x_n|y_n = i, C_m)$.
   - Update training dataset T_{m+1} as Tt_{m+1}: $T_{m+1} = \{Tr_m, Tt_{m+1}\}$.
2. Learn the final classifier using T_{M+1}.

through random forest C_m and updated class priors are issued from the same rule as previously: $\pi_n^{m+1} \propto \pi_n^1 p(x_n|y_n = i, C_m)$. β gives the proportion of training examples in the training set Tr_m while the remainder $(1 - \beta)$ training examples fall in the test set Tt_m. Setting β obeys to a trade-off: for a good assessment of random forest C_m, the number of samples in Tr_m must be high enough. But if β is too high, only very few samples will be updated at each iteration leading to a very slow convergence of the algorithm. In practice β is typically set to 0.75.

The algorithm is shown in the table 2. In the subsequent, this procedure will be denoted as IP2 (Iterative Procedure 2).

4 Application to Semi-supervised Learning

4.1 Related Work

Semi-Supervised Learning is reviewed in [28]. Four types of methods can be distinguished. The first type includes generative models often exploiting Expectation Maximization schemes that assess parameters of mono-modal Gaussian models [29] [28] or multi-modal Gaussian models [30]. Their advantages are the consistency of the mathematical framework with the probabilistic setting. The second category refers to discriminative models such as the semi-supervised support vector machine (S3VM) [31] [28]. Despite a mathematically-sound basis and good performances, S3VM are subject to local optimization issues and S3VM can be outperformed by other models depending on the dataset. Graph-based classifier is an other well known category in semi-supervised learning [32] [28]. The approach is close to the K-nearest-neighbour approach but similarities between examples are also taken in account. The principal drawback is that this classifier is mostly transductive: generalization properties are rather weak and performances decrease with unobserved data. The last family of semi-supervised models is formed by iterative schemes such as the self-training approach [33] or the co-training approach [34] that is applicable if observation features can be split into two independent groups. The advantage is the good performance reached by these methods and the simplicity of the approach. Their drawbacks mostly lie in the difficulties to characterize convergence properties.

4.2 Self Training with Soft Random Forests

A semi-supervised version of the iterative procedure proposed in the previous section can be derived. Following a self training strategy, it conststs in initially training a random forest from groundtruthed training samples only. Then, at each iteration, unlabelled data are processed by the current classifier and the K samples with the greatest class posteriors are appended to the training database to retrain the classifier. If should be stressed that in the standard implementation of semi-supervised learning with SVMs and random forest the new samples appended to training set at each iteration are assigned class labels. In contrast, we benefit from the proposed weakly supervised decision trees. This is expected to reduce the propagation of classification errors. The sketch of semi-supervised learning is given in table 3.

Table 3. Soft self-training procedure for semi-supervised learning

Given an initial training data set $T = \{T_L, T_U\}$, where T_L contains labelled data and T_U unlabelled data, and M iterations.

1. For m from 1 to M
   - Learn a classifier C_m from T_L.
   - Apply classifier C_m to T_U.
   - For each classes, transfer from T_U to T_L the most confident examples, with weak label, according to the probabilistic classification.
2. Generate the final classifier using T_L.

5 Experiments

5.1 Simulation Protocol

In this section, we compare four classification models: IP1 using soft random forests, IP2 using soft random forests, soft random forests alone, and the generative model proposed in [2] for weakly labelled data.

Given a supervised dataset, a weakly supervised training dataset is built. We distribute all the training examples in several groups according to predefined target class proportions (table 4). All the instances in a given group are assigned the class proportion of the group. In table 4, we show an example of target class proportions for a three-class dataset. In this example, we can create groups containing from one class (supervised learning) to three classes. For each case of class-mixture, different mixture complexities can be created: from one class dominating the mixture, i.e. the prior of one class being close to one, to equiprobable class, i.e. equal values for non-zeros class priors.

To evaluate the performances of the proposed weakly supervised learning strategies, we consider different reference datasets of the UCI machine learning repository so that reported experiments could be reproduced. The three considered datasets have been chosen to provide representative examples of the datasets to be dealt with in computer vision applications. We do not consider datasets with two classes because they do not allow us to generate complex class-mixtures. D1 is an image segmentation dataset containing 7 classes of texture and 330 instances per class. Each sample is characterized by a 19-dimensional real feature vector. D1 is a typical computer vision dataset drawn from a database of 7 outdoor images (brickface, sky, foliage, cement, window, path, grass). D2 is the classical Iris dataset containing 3 classes and 50 instances per class. Each objectis characterized by geometric features, i.e. length and width of the petals. D3 is the Synthetic Control Chart Time Series dataset, containing 6 classes of typical line evolutions, 100 instances per classes, and 5 quantified descriptors. An interesting property of this dataset is that the distribution of the features within each class is not unimodal and cannot be easily modelled using a parametric approach. This is particularly relevant for computer vision applications where objects classes often lead non-linear manifolds in the feature space. Dataset D3

Table 4. Example of training class priors for a dataset with 3 classes. Different cases are carried out: from the supervised labelling to the high complexity mixture.

Dataset with 3 classes, 1-class mixture labels:
$\begin{pmatrix}1\\0\\0\end{pmatrix}$ $\begin{pmatrix}0\\1\\0\end{pmatrix}$ $\begin{pmatrix}0\\0\\1\end{pmatrix}$ (supervised learning)
Dataset with 3 classes, 2-class mixture labels:
$\begin{pmatrix}0.8\\0.2\\0\end{pmatrix}$ $\begin{pmatrix}0.2\\0.8\\0\end{pmatrix}$ $\begin{pmatrix}0.6\\0.4\\0\end{pmatrix}$ $\begin{pmatrix}0.4\\0.6\\0\end{pmatrix}$ $\begin{pmatrix}0.8\\0\\0.2\end{pmatrix}$ $\begin{pmatrix}0.2\\0\\0.8\end{pmatrix}$ $\begin{pmatrix}0.6\\0\\0.4\end{pmatrix}$ $\begin{pmatrix}0.4\\0\\0.6\end{pmatrix}$ $\begin{pmatrix}0\\0.8\\0.2\end{pmatrix}$ $\begin{pmatrix}0\\0.2\\0.8\end{pmatrix}$ $\begin{pmatrix}0\\0.6\\0.4\end{pmatrix}$ $\begin{pmatrix}0\\0.4\\0.6\end{pmatrix}$
Dataset with 3 classes, 3-class mixture labels:
$\begin{pmatrix}0.8\\0.1\\0.1\end{pmatrix}$ $\begin{pmatrix}0.1\\0.8\\0.1\end{pmatrix}$ $\begin{pmatrix}0.1\\0.1\\0.8\end{pmatrix}$ $\begin{pmatrix}0.4\\0.2\\0.2\end{pmatrix}$ $\begin{pmatrix}0.2\\0.4\\0.2\end{pmatrix}$ $\begin{pmatrix}0.2\\0.2\\0.4\end{pmatrix}$

was also chosen to investigate the classification of time series depicting different behaviours (cyclic, increasing vs. decreasing trend, upward vs. downward shift). This is regarded as a mean to evaluate the ability to discriminate dynamic contents particularly relevant for video analysis, i.e. activity recognition, trajectory classification, event detection.

Cross validation over 100 tests allows assessing a mean classification rate. 90% of data are used to train classifier while the 10% remainders are used to test. Dataset is randomly split every test and the procedure that affects weak labels to the training data is carried out every test. A mean correct classification rate is extracted over the cross validation.

5.2 Experiments on Weakly Supervised Dataset

We report weakly supervised experiments in table 5 for the tree datasets. Results are provided as a function of the training dataset and as a function of the class mixture complexity, from the supervised learning (1-class mixture) to the maximum complexity mixture (mixture with all classes). Results are reported for the iterative procedures IP1 (section 3.1) and IP2 (section 3.2), the weakly supervised learning of soft random forests, the generative and discriminant models previously proposed in [2] and [9]. The later respectively exploit Gaussian mixtures and a kernel Fisher discrimination technique.

Overall, the iterative process IP2 outperforms the other models. Even if random forests alone are outperformed by the generative model with D2, the iterative procedure leads to improved classification. The explanation is that class priors are iteratively refined to finally resort to less fuzzy priors. Experiments with dataset D3, particularly stress the relevance of the introduction of soft decision trees. Due to the interlaced structure of the feature distribution for each class of dataset D3, the generative and discriminative models perform poorly. In contrast the weakly supervised random forests reach correct classification rates close to the supervised reference even with complex 6-class training mixtures. For instance, considering D3 and 6-classes mixture labels, the iterative procedure IP2 combined to soft forest reach 98.8% (vs. 100% in the supervised case) where the generative end discriminative models only reach 58.3% and 59.8% of correct classification.

Table 5. Classification performances for datasets D1, D2, and D3: the mean correct classification rate (%) is reported as a function of the complexity of the mixture label for the 5 classification models IP1 + soft trees, IP2+ soft trees, soft trees and random forest alone, a EM-based generative algorithm [2], and a discriminative-based algorithm [9]

Dataset, type of mixtures	IP1 + soft trees	IP2 + soft trees	soft trees	Naive bayes [2]	Fisher + K-pca [9]
D1, 1 classes mixture	-	-	**96.1%**	83.7%	89.7%
D1, 2 classes mixture	90.7%	**96.1%**	92.3%	83.6%	89.2%
D1, 3 classes mixture	88.7%	**95.9%**	91.2%	84.4%	89.5%
D1, 4 classes mixture	88.3%	**94.4%**	88.4%	83.7%	89.1%
D1, 5 classes mixture	85.0%	**94.1%**	88.8%	83.8%	89.1%
D1, 6 classes mixture	75.2%	**92.7%**	84.6%	83.1%	89.1%
D1, 7 classes mixture	55.1%	81.4%	62.6%	75.1%	**85.9%**
D2, 1 classes mixture	-	-	**97.3%**	94.6%	96.0%
D2, 2 classes mixture	**97.3%**	**97.3%**	90.6%	95.3%	87.3%
D2, 3 classes mixture	84.0%	**92.6%**	81.3%	85.3%	76.6%
D3, 1 classes mixture	-	-	**100%**	77.1%	66.8%
D3, 2 classes mixture	90.5%	**100%**	90.0%	62.2%	63.6%
D3, 3 classes mixture	91.3%	**99.5%**	89.3%	62.1%	61.5%
D3, 4 classes mixture	82.1%	**98.1%**	75.6%	45.5%	62%
D3, 5 classes mixture	74.6%	**97.3%**	82.1%	47.3%	59.1%
D3, 6 classes mixture	94.0%	**98.8%**	88.6%	58.3%	59.8%

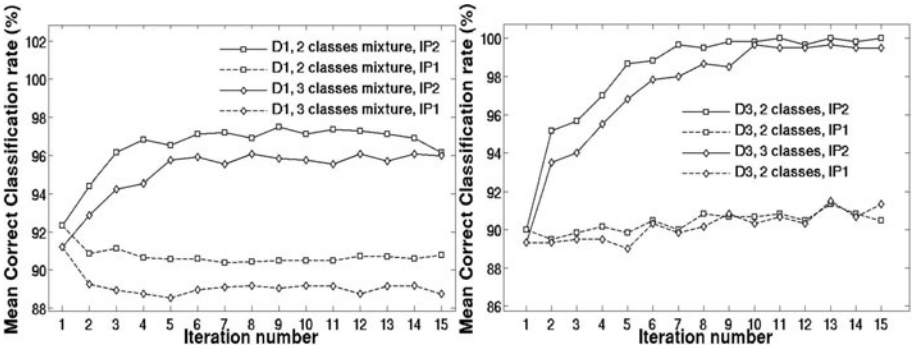

Fig. 1. Evolution of the performances of the iterative procedures IP1 and IP2 through iteration: dataset D1 (left), dataset D3 (right)

To further illustrate the behaviour of the iterative procedures IP1 and I2, we report in figure 1 the evolution of the mean correct classification rate (for the test set) as a function of the iteration for dataset D1 and D2 and several types of class mixtures. These plots state the relevance of the procedure IP2 compared to procedure IP1. Whereas the later does not lead to significant improvement over iterations, the gain in the correct classification rate can be up to 10% after the convergence of the IP2 procedure, for instance for dataset D2 and 2-class mixtures. The convergence is typically observed on a ten of iterations. These results can be explained by the fact that the IP2 procedure distinguishes at each iteration separate training and test set to update the random forests and the class priors.

5.3 Application to Fish School Classification in Sonar Images

Weakly supervised learning is applied to fisheries acoustics data [9] formed by a set of fish schools automatically extracted in sonar acoustics data. Fish schools are extracted as connected components using a thresholding-based algorithm. Each school is characterized by a X-dimensional feature vector comprising geometric (i.e. surface, width, height of the school) and acoustic (i.e. backscattered energy) descriptors. At the operation level, training samples would issue from the sonar in trawled areas, such any training school would be assigned the relative priors of each class. With a view to performing a quantitative evaluation, such weakly supervised situations are simulated from a groundtruthed fish school dataset. The later has been built from sonar images in trawled regions depicting only one species.

From results given in table 6, we perform a comparative evaluation based on the same methods than in section 5.2 as shown in table 5. Class proportions have been simulated as presented in table 4. Similar conclusions can be drawn. Overall the iterative procedure with soft random forests (IP2-SRF) outperforms the other techniques including the generative and discriminative models presented in [2] [9], except for the four-class mixture case where soft random forests alone perform better (58% vs. 55%). The operational situations typically involve mixtures between two or three species and the reported recognition performances (between 71% and 79%) are relevant w.r.t. ecological objectives in terms of species biomass evaluation and the associated expected uncertainty levels.

Table 6. Classification performances for sonar image dataset D4: the mean correct classification rate (%) is reported as a function of the complexity of the mixture label for the 5 classification models IP1 + soft trees, IP2+ soft trees, soft trees and random forest alone, a EM-based generative algorithm [2], and a discriminative-based algorithm [9].

Dataset, type of mixtures	**IP1 + soft trees**	**IP2 + soft trees**	**soft trees**	**Naive bayes** [2]	**Fisher + K-pca** [9]
D4, 1 classes mixture	-	-	**89.3%**	66.9%	69.9%
D4, 2 classes mixture	72.3%	**79.4%**	71.9%	52%	71.7%
D4, 3 classes mixture	62.9%	**70%**	68.3%	51.2%	65.9%
D4, 4 classes mixture	45.3%	55%	**58.7%**	47.9%	56.2%

5.4 Semi-supervised Experiments

Semi-supervised experiments have been carried out using a procedure similar to the previous section. Training and test sets are randomly built for a given dataset. Each training set is composed of labelled and unlabelled samples. We here report results for datasets D2 and D3 with the following experimental setting. For dataset D3 the training dataset contains 9 labelled examples (3 for each class) and 126 unlabelled examples (42 for each class). For dataset D3, we focus on a two-class example considering only samples corresponding to normal and cyclic pattern. Training datasets contain 4 labelled samples and 86 unlabelled samples per class. This particular experimental setting is chosen to illustrate the relevance of the semi-supervised learning when only very fulled labelled training

samples are available. In any case, the upper bound of the classification performances of a semi-supervised scheme is given by the supervised case. Therefore, only weak gain can be expected when a representative set of fully labelled samples is provided to the semi-supervised learning.

Five semi-supervised procedure are compared: three based on self-training (ST) strategies [28], with soft random forests (ST-SRF), with standard (hard) random forests (ST-RF), with a nave Bayes classifier (ST-NBC), a EM-based nave Bayes classifier (EM-NBC) [2] and the iterative procedure IP2 to soft random forest (IP2-SRF). Results are reported in figure 2.

These semi-supervised experiments first highlight the relevance of the soft random forests compared to their standard versions. For instance, when comparing both to a self-training strategy, the soft random forests lead to a gain of 5% of correct classification with dataset D3. This is regarded as a direct consequence of a reduced propagation of initial classification errors with soft decisions. The structure of the feature space for dataset D3 further illustrates as previously the flexibility of the random forest schemes compared to the other ones, especially generative models which behave poorly.

These experiments also demonstrate the relevance of the weakly supervised learning IP2-SRF in a semi-supervised context. The later favourably compares to the best self-training strategy (i.e. 90% vs 82.5% of correct classification for dataset D2 after 10 iterations). This can be justified by the relations between the two procedures. As mentioned in section 4, the self training procedure with soft random forests can be regarded as a specific implementation of the iterative procedure IP2. More precisely, the self-training strategy consists in iteratively appending unlabelled samples to the training dataset. At a given iteration, among the samples not yet appended to the training set, those with the greatest measures of the confidence in the classification are selected. Hence the classification decisions performed for the samples in the training set are never re-evaluated. In contrast

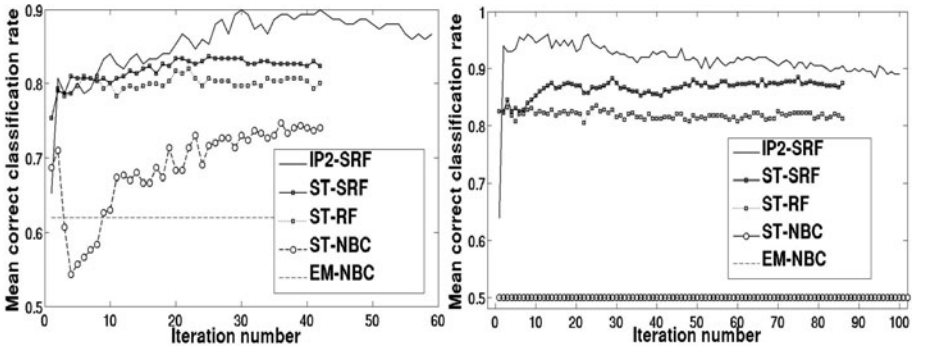

Fig. 2. Classification performances in semi-supervised contexts: dataset D2 (left) and dataset D3 (right) restricted to classes "standard patterns" and "cyclic pattern". Five approaches are compared: ST-SRF, ST-RF, ST-NBC, EM-NBC, and IP2-SRF (cf. text for details).

to this deterministic update of the training set, the weakly supervised iterative procedure IP2 exploits a randomization-based strategy where unlabelled sample are randomly picked to build at each iteration a training set. Therefore, soft classification decisions are repeatedly re-evaluated with respect to the updated overall knowledge. Then, the proposed entropy criterion (equation (3)) implies that fully labelled samples are also implicitly given more weight as in the soft training procedures. These different features support the better performances reported here for the iterative procedure IP2 combined to soft random forests.

6 Conclusion

We have presented in this paper methodological contributions for learning object class models in a weakly supervised context. The key feature is that classification information in the training datasets are given as the priors of the different classes for each training sample. This weakly supervised setting covers the supervised situations, the semi-supervised situations, and computer vision applications as the object recognition scheme that only specifies the presence or absence of each class in each image of the training dataset.

Soft random forests for weakly supervised learning: From a methodological point of view, a first original contribution is a procedure to learn soft random forests in a weakly supervised context. Interestingly the later is equivalent to the C4.5 random forest [15] if a supervised dataset is available such that recognition performances for low uncertainty priors can be granted. The second methodological contribution is an iterative procedure aimed at iteratively improving the learning model.

The experimental evaluation of these methodological contributions for several reference UCI datasets demonstrate their relevance compared to previous work including generative and discriminative models [2] [9]. We have also shown that these weakly supervised learning schemes are relevant for semi-supervised datasets for which they favourably compare to standard iterative techniques such as self-training procedures. The experiments support the statement widely acknowledged in pattern recognition that, when relevantly iterated, soft decisions perform better than hard decisions.

Weakly supervised learning for computer vision applications: The reference UCI datasets considered in the reported experiments are representative of different types of computer vision datasets (i.e. patch classification, object recognition, trajectory analysis). For these datasets, we have shown that recognition performances close to upper bounds provides by the supervised learning could be reached by the proposed weakly learning strategy even when the training set mostly high uncertainty class priors.

This is of particular importance as it supports the relevance of the weakly supervised learning to process uncertain or contradictory expert or automated preliminary interpretations. In any case, such a learning scheme should make in computer vision applications the construction of training datasets which is task

often a very tedious task. The later observation initially motivated the introduction of the semi-supervised learning and of the weakly supervised case restricted to presence and absence information. Our work should be regarded as a further development of these ideas to take into account more general prior information. As illustrated for instance by the fisheries acoustics dataset, we believe that this generalization may permit reaching relevant classification performances when the knowledge of presence and/or absence information only leads to unsatisfactory classification rates [35] [2].

Given the central role of the randomization-based sampling in the iterative procedure, future work will focus on its analysis both form experimental and theoretical points of view. The objectives will be to characterize the convergence of this procedure as well to evaluate different random sampling beyond the uniform sampling tester in the reported experiments. A stopping criteria might also be considered.

References

1. Crandall, D.J., Huttenlocher, D.P.: Weakly supervised learning of part-based spatial models for visual object recognition. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 16–29. Springer, Heidelberg (2006)
2. Ulusoy, I., Bishop, C.M.: Generative versus discriminative methods for object recognition. In: CVPR, vol. 2, pp. 258–265 (2005)
3. Ponce, J., Hebert, M., Schmid, C., Ziserman, A.: Toward Category-Level Object Recognition. LNCS, vol. 4170. Springer, Heidelberg (2006)
4. Weber, M., Welling, M., Perona, P.: Unsupervised learning of models for object recognition. In: Vernon, D. (ed.) ECCV 2000. LNCS, vol. 1842, pp. 18–32. Springer, Heidelberg (2000)
5. Fergus, R., Perona, P., Zisserman, A.: Weakly supervised scaled-invariant learning of models for visual recognition. IJCV 71, 273–303 (2006)
6. Schmid, C.: Weakly supervised learning of visual models and its application to content-based retrieval. IJCV 56, 7–16 (2004)
7. Rossiter, J., Mukai, T.: Bio-mimetic learning from images using imprecise expert information. Fuzzy Set and Systems 158, 295–311 (2007)
8. van de Vlag, D., Stein, A.: Incorporing uncertainty via hierarchical classification using fuzzy decision trees. IEEE Transaction on GRS 45, 237–245 (2007)
9. Lefort, R., Fablet, R., Boucher, J.M.: Combining image-level and object-level inference for weakly supervised object recognition. Application to fisheries acoustics. In: ICIP (2009)
10. Maccormick, J., Blake, A.: A probabilistic exclusion principle for tracking multiple objects. IJCV 39, 57–71 (2000)
11. Neville, J., Jensen, D.: Iterative classification in relational data. In: AAAI workshop on learning statistical models from relational data, pp. 42–49 (2000)
12. Neal, R., Hinton, G.: A view of the EM algorithm that justifies incremental, sparse and other variants. Kluwer Academic Publishers, Dordrecht (1998)
13. Lachlan, G.M., Krishnan, T.: The EM algorithm and extentions. Wiley, Chichester (1997)
14. Kotsiantis, P., Pintelas, P.: Logitboost of simple bayesian classifier. Informatica Journal 29, 53–59 (2005)

15. Breiman, L.: Random forests. Machine Learning 45, 5–32 (2001)
16. Calonder, M., Lepetit, V., Fua, P.: Keypoint signatures for fast learning and recognition. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 58–71. Springer, Heidelberg (2008)
17. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classification and regression trees. Chapman and Hall, Boca Raton (1984)
18. Quinlan, J.: Induction of decision trees. Machine Learning 1, 81–106 (1986)
19. Quinlan, J.: C4.5: Programs for machine learning. Morgan Kaufmann Publishers, San Francisco (1993)
20. Loh, W.Y., Shih, Y.Y.: Split selection methods for classification trees. Statistica Sinica 7, 815–840 (1997)
21. Kass, G.V.: An exploratory technique for invesgating large quantities of categorical data. Journal of applied statistics 29, 119–127 (1980)
22. Breiman, L.: Bagging predictors. Machine Learning 26, 123–140 (1996)
23. Ho, T.K.: Random decision forest. ICDAR (1995)
24. Geurts, P., Ernst, D., Wehenkel, L.: Extremely randomized trees. Machine Learning 36, 3–42 (2006)
25. Culp, M., Michailidis, G.: An iterative algorithm for extending learners to semi-supervised setting. In: The 2007 Joint Statistical Meetings (2007)
26. Haffari, G., Sarkar, A.: Analysis of semi-supervised learning with the yarowsky algorithm. In: 23rd Conference on Uncertainty in Artificial Intelligence (2007)
27. Macskassy, S.A., Provost, F.: A simple relational classifier. In: Proceedings of the second workshop on multi-relational data mining, pp. 64–76 (2003)
28. Chapelle, O., Schölkopf, B., Zien, A.: Semi-supervised learning. MIT Press, Cambridge (2006)
29. Nigam, K., McCullum, A., Thrun, S., Mitchel, T.: Learning to classify text from labeled and unlabeled documents. In: AAAIJ (1998)
30. Nigam, K., McCallum, A., Thrun, S., Mann, G.: Text classification from labeled and unlabeled documents using em. Machine Learning 39, 103–134 (2000)
31. Joachims, T.: Transductive inference for text classification using support vector machines. In: ICML, pp. 200–209 (1999)
32. Zhu, X., Ghahramani, Z., Lafferty, J.: Semi-supervised learning using gaussian fields and harmonic functions. In: ICML, pp. 912–919 (2003)
33. Rosenberg, C., Hebert, M., Schneidermann, H.: Semi-supervised self-training of object detection models. In: WACV (2005)
34. Blum, A., Mitchel, T.: Combining labeled and unlabeled data with co-training. In: WCLT, pp. 92–100 (1998)
35. Fablet, R., Lefort, R., Scalabrin, C., Mass, J., Boucher, J.M.: Weakly supervised learning using proportion based information: an application to fisheries acoustic. In: International Conference on Pattern Recognition (2008)