Mach Learn (2017) 106:359–386
DOI 10.1007/s10994-016-5610-8

Nearest neighbors distance ratio open-set classifier

Pedro R. Mendes Júnior[1] · Roberto M. de Souza[2] · Rafael de O. Werneck[1] · Bernardo V. Stein[1] · Daniel V. Pazinato[1] · Waldir R. de Almeida[1] · Otávio A. B. Penatti[1,3] · Ricardo da S. Torres[1] · Anderson Rocha[1]

Received: 18 October 2015 / Accepted: 9 November 2016 / Published online: 15 December 2016

Abstract In this paper, we propose a novel multiclass classifier for the open-set recognition scenario. This scenario is the one in which there are no a priori training samples for some classes that might appear during testing. Usually, many applications are inherently open set. Consequently, successful closed-set solutions in the literature are not always suitable for real-world recognition problems. The proposed open-set classifier extends upon the Nearest-Neighbor (NN) classifier. Nearest neighbors are simple, parameter independent, multiclass, and widely used for closed-set problems. The proposed Open-Set NN (OSNN) method incorporates the ability of recognizing samples belonging to classes that are unknown at training time, being suitable for open-set recognition. In addition, we explore evaluation measures for open-set problems, properly measuring the resilience of methods to unknown classes during testing. For validation, we consider large freely-available benchmarks with different open-set recognition regimes and demonstrate that the proposed OSNN significantly outperforms their counterparts in the literature.

Keywords Open-set recognition · Nearest neighbor classifier · Open-set nearest-neighbor classifier · Nearest neighbors distance ratio · Open-set evaluation measures

Editor: Hendrik Blockeel.

✉ Pedro R. Mendes Júnior
pedrormjunior@gmail.com

[1] RECOD Lab., Institute of Computing (IC), University of Campinas (UNICAMP), Av. Albert Einstein, 1251, Campinas, SP 13083-852, Brazil

[2] Faculty of Electrical Engineering and Computing (FEEC), University of Campinas (UNICAMP), Av. Albert Einstein, 400, Campinas, SP 13083-852, Brazil

[3] SAMSUNG Research Institute, Advanced Technologies Group, Av. Cambacica, 1200, Bloco 1, Campinas, SP 13097-160, Brazil

1 Introduction

Typical *pattern classification* refers to the problem of assigning a test sample to one or more known classes. For instance, classifying an image of a digit as one out of 10 possible digits (0…9). We know, by definition, that this problem has 10 classes. On the other hand, *recognition* is the task of verifying whether a test sample belongs to one of the known classes and, if so, finding to which of them the test sample belongs. In the recognition problem, the test sample can belong to none of the classes known by the classifier during training. The recognition scenario is more similar to what we call an *open-set scenario*, in which the classifier cannot be trained with all possible classes because the classes are ill-sampled, not sampled, or *unknown* (Scheirer et al. 2013).

In some problems, all classes are known a priori, leading to a closed-set scenario. For example, suppose that inside an aquarium there are only three species of fish and biologists are interested in training a classifier for distinguishing these three classes. In this application, all test samples are assigned to one out of those classes because it is known that all fish species that could be tested at the aquarium belong to one of those three classes. The same classifier, however, is unsuitable for being used in a new larger aquarium containing the same three species and some new ones, i.e., in an open-set scenario in which new species are unknown. In this case, the trained classifier will always classify an unknown sample as belonging to a known class because it was developed and trained to be used in the closed-set scenario (first aquarium), leading to an undesired misclassification.

Open-set classification problems are typically a multiclass problem. The classifier must assign the label of one of the training classes or an unknown label to test samples. Approaches aiming at tackling this problem must avoid the following errors:

Misclassification The test sample belongs to one of the known training classes but the classifier assigns it to a wrong class;

False unknown The test sample belongs to one of the known training classes but the classifier assigns it to the unknown label; and

False known The test sample is unknown but the classifier assigns it to one of the known training classes (e.g., the aforementioned fish species recognition error).

In a closed-set classification scenario, only the first kind of error is possible.

Common classifiers for closed-set setups are usually optimized to minimize the *empirical risk*[1] measured on training samples. In an open-set recognition scenario, the objective is also to minimize the *risk of the unknown*[2] by minimizing the *open space risk* instead of minimizing only the empirical risk. The *open space* is all the region of the feature space outside the support of the training samples. The *positively labeled open space* (PLOS), in the level of binary classification, is "the open space representing the learned recognition function f" (Scheirer et al. 2013), i.e., the intersection of the open space and the *positively labeled region* (the region of the feature space in which a sample would be classified as positive). In this vein, the open space risk measures the PLOS. When combining binary classifiers with the multiclass-from-binary one-vs-all approach, the union of the PLOSs would be called *known labeled open space* (KLOS). The same applies to inherently multiclass classifiers: all the region of the feature space, outside the support of the training samples, in which a sample would be classified as belonging to one of the known classes, is the KLOS.

[1] Approximation of the actual risk. It is calculated by averaging the loss function on the training set.

[2] Risk of the unknown from insufficient generalization or specialization of a recognition function f (Scheirer et al. 2013).

A common approach to partially handling the open-set scenario relies on the use of threshold-based classification schemes (Phillips et al. 2011). Establishing a threshold on the *similarity score* means *rejecting distant* samples from the training samples. Basically, those methods verify whether the similarity score is greater than or equal to a previously defined threshold.

Another trend relies on modifying the classification engine or objective function of Support Vector Machines (SVM) classifiers (Costa et al. 2014; Scheirer et al. 2013, 2014). The one-vs-all multiclass-from-binary SVM (SVM$^{\text{MCBIN}}$)[3] (Rocha and Goldenstein 2009, 2014) can be considered an open-set classifier, as all the one-vs-all binary SVMs used in the SVM$^{\text{MCBIN}}$ procedure are able to classify a test sample as negative and, in this case, the test sample could be considered unknown. Those recent works aim at decreasing the positively labeled region for each binary SVM, such that, when combining them using the multiclass-from-binary technique, the risk of the unknown is minimized.

In this work, we address the open-set recognition problem by introducing an open-set version of the Nearest Neighbor (NN; Bishop 2006) classifier, hereinafter referred to as Open-Set NN (OSNN). Our approach extends upon the traditional closed-set NN and introduces a modification to verify whether or not a test sample can be classified as unknown. Instead of using a threshold on the similarity score for the most similar class, the proposed method uses the ratio of similarity scores to the two most similar classes by applying a threshold on it. One advantage of the proposed approach, compared to other existing methods for open-set scenarios, is that it is inherently multiclass, i.e., the efficiency of the OSNN is not affected as the number of *available classes* for training increases.

Our main hypothesis herein is that bounding the KLOS is key to design effective open-set classifiers. We assume this because we do not know in what region of the feature space a test sample belonging to an unknown class would appear. For open-set scenarios, we cannot assume we know how "open" the scenario is, i.e., how many classes could appear at testing or system usage. Another major characteristic of the proposed solution is that it can create a bounded KLOS (limited open space risk) therefore gracefully protecting the classes of interest and rejecting unknown classes.

In addition to the proposed open-set solution, we have designed a special experimental protocol for benchmarking open-set methods. In many works in the literature, in spite of the explicit discrimination between classification and recognition, authors perform tests on recognition problems considering a closed-set scenario instead of an open-set one (Bartlett and Wegkamp 2008; Chew et al. 2012; Jayadeva and Chandra 2007; Malisiewicz et al. 2011). Consequently, the observed results are not similar to what is observed in real-world open-set applications. Another limitation of existing experimental protocols is the lack of appropriate measures to assess the quality of the open-set classifiers. Therefore, another contribution of this work is the discussion of two measures adapted to the open-set setup: the *normalized accuracy* (NA) and the *open-set f-measure* (OSFM). The purpose of such adapted measures is to evaluate the performance of classifiers when handling both known and unknown test samples.

For the experiments and validation, we considered a diverse set of recognition problems, such as object recognition (Caltech-256, ALOI, and Ukbench), scene recognition (15-Scenes), letter recognition (Letter), and sign language recognition (Auslan). The number of classes in such problems varies from 15 to 2550 and the number of examples vary from a few thousands to hundreds of thousands. The experiments were performed by considering training setups with three, six, nine, and twelve classes of interest, and testing scenarios with

[3] The MCBIN in the name denotes the classifier is a multiclass one extended from binary classifiers.

samples of the remaining classes as possible unknown. We compared the proposed OSNN with recent methods proposed for open-set scenarios (one-vs-all One-Class SVM of Pritsos and Stamatatos 2013; 1-vs-Set Machine of Scheirer et al. 2013; Decision Boundary Carving of Costa et al. 2014; Weibull-calibrated SVM of Scheirer et al. 2014). In addition, we compared the proposed method with the traditional NN and the NN using threshold on the similarity score. The proposed open-set solution outperformed all existing solutions with statistical significance.

We organized the remainder of this paper into the following sections. In Sect. 2, we present related work in open-set recognition. In Sect. 3, we describe the proposed method. In Sect. 4, we show experiments and validation while, in Sect. 5, we conclude the paper.

2 Related work

In this section, we present previous approaches that somehow deal with open-set classification scenarios. Those approaches can be divided into two main categories: approaches resulting from adaptations of similar problems (Sect. 2.1) and approaches that explicitly deal with open-set scenarios (Sect. 2.2).

2.1 Approaches resulting from adaptations of similar problems

One-class classifiers, such as the One-Class SVM (OCSVM; Schölkopf et al. 1999), seem promising for open-set scenarios, as it tries to focus on the known class and ignore everything else. The OCSVM finds the best margin with respect to the origin and kernels can be applied, creating a bounded positive region around the samples of the known classes. This is the most reliable approach in cases in which the access to a second class is very difficult or even impossible.

Heflin et al. (2012) and Pritsos and Stamatatos (2013) used OCSVMs in a multiclass fashion: as the OCSVM is similar to binary classifiers (its output is the positive or the negative class), they compose OCSVMs using the multiclass-from-binary approach. For the cases in which no OCSVM classifies as positive, the test sample is rejected, i.e., it is classified as unknown. As Heflin et al. (2012) was dealing with a multiple class problem, for the cases whereby two or more OCSVMs classify as positive, all these positive labels are considered valid. Differently, Pritsos and Stamatatos (2013) choose the OCSVM that are more confident about its decision: the one with the highest positive distance to the hyperplane. In our work, we use the method of Pritsos and Stamatatos (2013), hereinafter referred to as one-vs-all One-Class SVM (SVM^{MCOC}), for comparison.

Zhou and Huang (2003), however, mention that the OCSVM has a limited use because it does not provide good generalization nor specialization. Several works dealing with OCSVM have tried to overcome the problem of lack of generalization (Jin et al. 2004; Cevikalp and Triggs 2012; Wu and Ye 2009; Manevitz and Yousef 2002). All of these works can be applied to the multiclass and open-set scenario in the same way the OCSVM can be applied.

Although one-class classifiers are inherently suitable for open-set classification problems, binary classifiers also hold potential. For example, binary classifiers can be applied to the open-set scenario (which is multiclass) using the one-vs-all (Rocha and Goldenstein 2014) approach. The binary classifier which classifies as positive is chosen to decide the final class of the multiclass classifier. When two or more binary classifiers return positive for the test sample, the one most confident about its classification is chosen to decide the final class. When no binary classifier classifies as positive, then the test sample is classified as unknown.

In this vein, all the variations of the SVM (Bartlett and Wegkamp 2008; Malisiewicz et al. 2011; Jayadeva and Chandra 2007; Chew et al. 2012, which are also binary classifiers) can be applied using the one-vs-all approach.

As we mentioned before, the trivial approach to handle the open-set scenario is to define a threshold on the similarity score of the classifiers. Also, one would be interested in rejecting doubtful or ambiguous samples. Fukunaga (1990) describes the *reject option* and present it as a form of postponing the decision-making process to further evaluate the test sample by other means (e.g., other classifiers). Note that in open-set scenarios, we want to classify a test sample as one of the known classes or as none of the known classes (unknown) without postponing the decision making. Chow (1970) presented a method for rejecting doubtful test samples, i.e., to avoid classifying the test sample as one of the known classes when the classifier has good similar scores for more than one class. Later, Dubuisson and Masson (1993) extended the *ambiguity reject option* of Chow (1970) and presented the *distance reject option* in the context of statistical pattern recognition. The distance reject option is to avoid classifying the test sample "far from" the training ones in the feature space. Muzzolini et al. (1998) further extended upon this idea to define better distance rejection thresholds adapted for each training class.

Works dealing with distance reject option can be applied to the open-set classification scenarios because if one ensures that faraway test samples are rejected (i.e., classified as unknown), then the classifier creates a bounded KLOS in the feature space. The problem for most of the methods dealing with rejection by thresholding the similarity score is the difficulty to define such threshold. Our work differs from these works because we are not defining thresholds directly on the similarity score but rather we obtain the ratio of similarity scores and perform the decision based on this ratio. Another key difference of our work is that we simulate an open-set regime during training thus better defining the parameters for such decision later on in the testing. According to our experiments, this approach is more appropriate to cope with in the feature space.

Note that recognition in an open-set scenario also differs from *classification with abstention* (Pietraszek 2005). In an open-set scenario, a test sample can belong to none of the known classes, consequently it must be classified as unknown. Regarding the works on *abstaining classifiers*, they want to abstain the classification when the classifier is not sure about its decision. In those works they do not assume the test sample can belong to an unknown class never seen at training phase. Even postponing the classification, with those methods, and without a proper treatment of the open-set setup later on, the test sample would be classified as one of the known classes.

Some *outlier detection* methods can be also applied to open-set scenarios. As presented by Hodge and Austin (2004), there are three fundamental approaches for the problem of outlier detection:

Type 1 "Determine the outliers with no prior knowledge of the data. This is essentially a learning approach analogous to *unsupervised clustering*. [...]"

Type 2 "Model both normality and abnormality. This approach is analogous to *supervised classification* and requires pre-labelled data, tagged as normal or abnormal. [...]"

Type 3 "Model only normality or in a very few cases model abnormality. [...] Authors generally name this technique novelty detection or novelty recognition. It is analogous to a *semi-supervised recognition or detection* task and can be considered semi-supervised as the normal class is taught but the algorithm learns to recognise abnormality."

Among these approaches, the Type-3 ones are more appropriate in an open-set scenario. Type-1 approaches do not make sense in an open-set setup because, for each class, we do *not* want to find out the outliers of that class. Furthermore, using this kind of approach with all training data available for open-set scenario, i.e., samples of all the n classes available for training, is not appropriate because, as the training data represent several classes, these data do not fit a single distribution.

Type-2 methods can be applied for open-set recognition, but they are not very convenient. Those approaches do not take into account that the samples used to represent the outliers do not represent all possible "outliers" in the open-set scenario. In open-set scenarios we assume we are not able to train the classifier with all possible classes. The SVM using the one-vs-all approach, evaluated in our work, fits this kind of outlier detector.

Finally, Type-3 methods, in fact, could be used for open-set recognition, as they seek to learn what is "normality". As mentioned by Hodge and Austin (2004), a Type-3 method "aims to define a boundary of normality". As the open-set recognition problem is a multiclass problem, one can train an outlier detector for each of the available classes. The work of Pritsos and Stamatatos (2013) implements this idea using the OCSVM as an outlier detector for each known class.

2.2 Approaches proposed for open-set problems

Some recent approaches have turned the attention to open-set problems directly and extended upon the SVM classifier to deal with the modified constraints of open-set scenarios (Scheirer et al. 2013; Costa et al. 2014). As the original SVM's *risk minimization* is based only on the known classes (empirical risk), it can misclassify the unknown classes that can appear in the testing phase. Differently, possible open-set solutions need to minimize the risk of the unknown (Scheirer et al. 2013).

Costa et al. (2012, 2014) presented a source camera attribution algorithm considering the open-set scenario and developed an extension called Decision Boundary Carving (DBC) upon the traditional SVM classifier. For the multiclass open-set scenarios, the authors proposed a binary classifier and used the one-vs-all approach. The extension in the binary classifier (the DBC) is to move the decision hyperplane found by the traditional SVM by a value ϵ inwards (possibly outwards) the positive class. The value of ϵ is defined by an exhaustive search to minimize the *training data error*. In our work, we use a multiclass-from-binary version of the DBC also using the one-vs-all approach for comparison purposes, hereinafter referred to as multiclass-from-binary DBC (DBC$^{\text{MCBIN}}$).

Scheirer et al. (2013) presented the concept of *positively labeled open space* (PLOS) from which we extended the concept of *known labeled open space* (KLOS) (Sect. 1) we use herein. Therein, the authors define the open space risk in a binary problem to measure the PLOS. The open space risk is the ratio of the volume of the PLOS to the volume of a sphere containing both the PLOS and the training samples. The risk of the unknown is the risk of classifying as positive (or one of the known classes, in the multiclass point of view) a test sample that is actually unknown. The open space risk measures the PLOS, i.e., the risk of the unknown. Scheirer et al. (2013) formalized the open-set recognition problem of finding the recognition function f as a minimization of the open space risk $R_{\mathcal{O}}$ and the empirical risk $R_{\mathcal{E}}$, as follows.

$$\arg\min_{f \in \mathcal{H}} \{R_{\mathcal{O}}(f) + \lambda_r R_{\mathcal{E}}(f)\}, \tag{1}$$

in which λ_r is a regularization constant.

For handling the open-set scenario, Scheirer et al. (2013) introduced the 1-vs-Set Machine (1VS) with a linear kernel formulation that can be applied to both binary and one-class SVMs. Their proposed extension lies in the binary classifier level. Similar to Costa et al. (2012, 2014), Scheirer et al. (2013) also move the original SVM hyperplane inwards the positive class, but now adding a parallel hyperplane "after" the positive samples, making the positively labeled region be the region between the two hyperplanes. The hyperplanes are initialized to contain all the positive samples between them. Then, a refinement step is performed to adjust the hyperplanes to generalize or specialize the classifier according to user-defined *parameter pressures*. As noted by the authors, better results are usually obtained when the original SVM hyperplane is near the positive boundary seeking a specialization, and the added hyperplane is adjusted seeking generalization. Although the PLOS is minimized by adding the second hyperplane, it remains unbounded. Consequently, from the multiclass point of view (when applying one-vs-all approach), the KLOS also remains unbounded. For the experiments with the 1VS method, we used the code kindly provided by the authors.

Recently, Scheirer et al. (2014) introduced a new open-set recognition formulation called Compact Abating Probability (CAP) in which the probability of pertinence to each class decreases as a test sample gets far from the training samples of the class. By using a CAP model, one can establish a threshold on the probability value and reject test samples faraway from the training ones. Scheirer et al. (2014) then defined a CAP model based on the OCSVM. If this model accepts the test sample (as belonging to the class under consideration), then the final decision is accomplished by using another CAP model based on the binary SVM and Extreme Value Theory (EVT). The whole process is dubbed Weibull-calibrated SVM (WSVM). The idea is to construct two independent estimates: one based on the positive training samples and another based on the negative ones. In the binary classification level, the test sample is classified as positive when the product of the probability that the test sample is from the positive class times the probability that the test sample is *not* from the negative class is above a threshold. In the multiclass-from-binary level, the class is the one in which the product of the probabilities is the largest. If the one-class CAP model rejects or the product of the probabilities is under the threshold for all binary classifiers, then the test sample is classified as unknown. For the experiments using WSVM, we used the code kindly provided by the authors.

3 Proposed new open-set solution

In this section, we introduce two inherently multiclass open-set extensions for the NN classifier. We refer to the first open-set extension, called Class Verification (CV), as OSNN$^{\text{CV}}$. We refer to the second open-set extension, called Nearest Neighbor Distance Ratio (NNDR), simply as OSNN.

Comparing the CV and NNDR extensions, the NNDR has the ability to classify test samples faraway from the training ones as unknown while the CV does not. As shown below, the NNDR is able to bound the KLOS. In this work, we use the CV extension as a baseline for the method we are proposing, i.e., the OSNN.

Class Verification The OSNN$^{\text{CV}}$ is based on the agreement of the labels of the two nearest neighbors with respect to a test sample. The training phase of the OSNN$^{\text{CV}}$ is the same of the NN, i.e., it only requires the storage of the training samples. In the prediction phase, it selects the two nearest neighbors from the test sample s. If both nearest neighbors have the same label, this label is assigned to the test sample. Otherwise, s is classified as unknown.

Nearest Neighbor Distance Ratio Similarly, the OSNN obtains the nearest neighbor t of the test sample s and then obtains the nearest neighbor u of s such that $\theta(u) \neq \theta(t)$, in which $\theta(x) \in \mathcal{L} = \{\ell_1, \ell_2, \dots, \ell_n\}$ represents the class of a sample x and $\mathcal{L}$ is the set of training labels. Then we calculate the ratio

$$R = d(s,t)/d(s,u), \tag{2}$$

in which $d(x, x')$ is the Euclidean distance between samples x and x' in the feature space. If R is less than or equal to the specified threshold T, $0.0 < T < 1.0$, s is classified with the same label of t. Otherwise, it is classified as unknown, i.e.,

$$\theta(s) = \begin{cases} \theta(t) & \text{if } R \leq T \\ \ell_0 & \text{if } R > T, \end{cases} \tag{3}$$

in which ℓ_0 is the unknown label.

For the NNDR extension, a test sample s faraway from the training samples is also classified as unknown. That happens because R tends to 1 as both the best similarity score and the best similarity score of the other class increase.

The NNDR technique can be applied, effortlessly, to other classifiers in which the similarity score to the most probable class is smaller than or equal to the similarity score to the second most probable class, e.g., the Optimum-Path Forest (OPF) classifier (Papa et al. 2007, 2009). In addition, other metrics could also be used and even the feature space considered could be a transformed one (e.g., via kernels).

Parameter optimization For the OSNN, we perform a *parameter optimization* phase adapted to the open-set scenario to find the best value for T. The optimization of T is based on the accuracy on a validation set. As the learned classifier will be used in an open environment (with unknown classes), we want to tune T so that it works well in such an open environment. To that aim, a simulation of that setting is set up. Among the classes that occur in the training set, half are chosen to act as "known" classes in the simulation, the other half as "unknown" in the simulation. For this, the training set is divided into a fitting set F that contains half for the instances of the "known" classes, and a validation set V that contains the other half of the instances of the "known" classes, and all instances of the "unknown" classes. Note that, as an OSNN classifier is trained with F and evaluated on V (that contains classes unknown in F), a simulation of the open-set scenario is performed to obtain the best value of T based on the accuracy obtained on V.

Figure 1a shows that only a small part of the dataset is used to train the classifier for the experiments we present in Sect. 4.3. It is only a small part because, as represented in Fig. 1b, the testing set used to obtain the results presented in Sect. 4.3 contains all the representative samples of a great percentage of the classes in the dataset. In Fig. 1c we depict how the parameter optimization for the OSNN is accomplished by creating a validation set V (a subset of the training set) that contains representative samples of classes not used to fit the classifier to search for the best value of T on V.

Once the sets F and V are defined as described, we try out values of T in the range from 0.5 to 1.0, trying out 10 different values evenly distributed. For the best value v, e.g., say that v is the ith value in the range, we make another *grid search* procedure for values around v. We then range the threshold T from the mean of the $(i-1)$th and ith values to the mean of ith and $(i+1)$th values trying out 10 different values evenly distributed. We then repeat this refinement procedure in four levels.

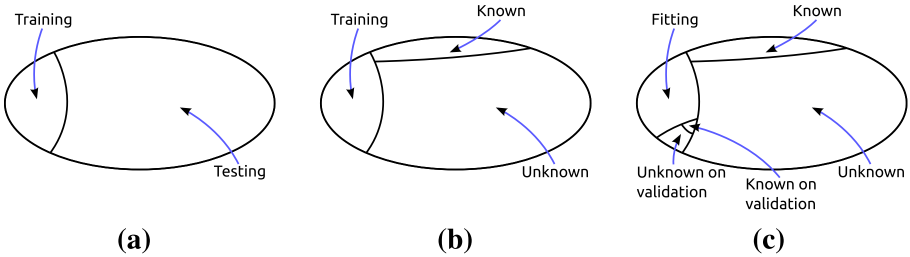


Fig. 1 Overview of data partitioning in the experiments and the parameter optimization of the Open-Set NN (OSNN). **a** A dataset is divided into training and testing sets as a real scenario would be. **b** Most of the samples in testing set are unknown but need to be properly classified during testing operation of the classifier. **c** Partitioning of the training set by simulating an open-set scenario for parameter optimization of the OSNN

The accuracy obtained on the unknown samples, during the performed simulation, can be seen as an estimation of the open space risk $R_{\mathcal{O}}$ in Eq. (1). To the best of our knowledge, our proposed method is the first to estimate $R_{\mathcal{O}}$ based on data. We present experiments regarding the parameter optimization phase in Sect. 4.4.

4 Experiments and validation

In this section, we present the evaluation measures (Sect. 4.1) and the experimental setup for validating the different methods (Sect. 4.2), including: details of the implementation of baseline classifiers, the datasets used for validation, and the experimental protocol. Then, we present the main experiments and the obtained results (Sect. 4.3) and experiments towards the evaluation of the parameter setting of the proposed method (Sect. 4.4). Finally, we further discuss the failing cases of OSNN (Sect. 4.5).

4.1 Evaluation measures

For evaluating classifiers in an open-set scenario, we should be aware of the unknown classes. Most of the existing evaluation measures, such as the macro- and micro-averaging f-measure, the average accuracy (Sokolova and Lapalme 2009), and the traditional classification accuracy (Chang and Lin 2011), do not take into account the unknown. Therefore, another contribution of this paper is the adaptation of two evaluation measures to assess the quality of open-set classifiers.

In the literature, the following classes of evaluation measures are found: (1) Measures for binary closed-set problems (traditional classification accuracy, f-measure, etc.); (2) Measures for multiclass closed-set problems (traditional classification accuracy, multiclass version of the f-measure, etc); and (3) Measures for binary open-set problems (Costa et al. 2014, the open-set version of the average accuracy). Measures in (1) and (2) are not appropriate as they do not consider open-set scenarios, which usually lead to the overestimation of the performance of evaluated classifiers. The measures adopted in (3), in turn, do not consider multiclass open-set classification problems.

In this work, we consider the open-set scenario potentially a multiclass scenario, i.e., one would want to classify a test sample as unknown or one of the known classes. Under those circumstances, the measures discussed in this work define a new class of evaluation measures, as they are suitable for *multiclass open-set* classification problems.

Here, we refer to the *known samples* as the samples belonging to one of the available classes for training. The *unknown samples* belong to classes for which no representative sample is used during training.

Normalized accuracy The *normalized accuracy* (NA) takes into account both the *accuracy on known samples* (AKS) and the *accuracy on unknown samples* (AUS). It was proposed because it avoids overestimating the performance of biased classifiers, i.e., classifiers that occasionally classify almost all samples as belonging to the most frequent class. This is important because the more open the scenario, the greater the amount of unknown samples.

The NA is defined as follows.

$$\text{NA} = \lambda_r \text{AKS} + (1 - \lambda_r)\text{AUS}, \tag{4}$$

in which λ_r, $0 < \lambda_r < 1$, is a regularization constant. Note that λ_r regulates the tradeoff of mistakes on the known and unknown cases. The NA differs from the accuracy presented by Costa et al. (2014) because the NA is the accuracy of the multiclass problem while the *final accuracy* in the work of Costa et al. (2014) is the average of the accuracies obtained in the binary problems.

Open-set f-measure Besides using the NA to assess the quality of results of classifiers in open-set scenarios, we also use extensions of the macro- and micro-averaging f-measure because these measures can give us fine-grained analysis of the behavior of the evaluated methods. The definition and the properties of f-measure are presented by Sokolova and Lapalme (2009). The following equation describes the traditional f-measure:

$$\text{f-measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \tag{5}$$

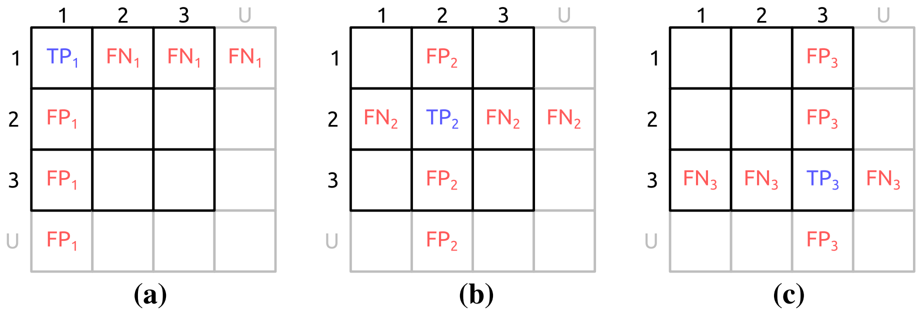
A trivial extension of f-measure to open-set scenario could be to consider the unknown as one simple class and obtain the f-measure value in the same way it is accomplished for the multiclass closed-set scenario. However, this trivial extension is not appropriate to evaluate tests in open-set scenarios because all correct classification of unknown test samples would be considered true positive classifications. These classification results cannot be considered true positive because it does not make sense to consider the unknown classes as one single positive class, since we have no representative samples of unknown classes to train the classifier.

Our open-set modifications to the f-measure are related to how precision and recall are computed. Equations (6) and (7) are used to compute the *macro-averaging open-set f-measure* (OSFM_M) and the *micro-averaging open-set f-measure* (OSFM_μ), respectively. The measures precision_M and recall_M stand for the macro precision and the macro recall, respectively. The measures precision_μ and recall_μ, in turn, stand for the micro precision and the micro recall, respectively. The following equations detail the proposed modified measures, which are the basis for the f-measure computed by Eq. (5) to define the *open-set f-measure* (OSFM):

$$\text{precision}_M = \frac{\sum_{i=1}^{l-1} \frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i}}{l-1}, \quad \text{recall}_M = \frac{\sum_{i=1}^{l-1} \frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i}}{l-1}, \tag{6}$$

$$\text{precision}_\mu = \frac{\sum_{i=1}^{l-1} \text{TP}_i}{\sum_{i=1}^{l-1} (\text{TP}_i + \text{FP}_i)}, \quad \text{recall}_\mu = \frac{\sum_{i=1}^{l-1} \text{TP}_i}{\sum_{i=1}^{l-1} (\text{TP}_i + \text{FN}_i)}, \tag{7}$$

in which $l = n + 1$ is the size of the confusion matrix (represented in Fig. 2) and n is the number of available classes for training. TP, FP, and FN stand for true positive, false positive,



	1	2	3	U
1	TP_1	FN_1	FN_1	FN_1
2	FP_1			
3	FP_1			
U	FP_1			

(a)

	1	2	3	U
1		FP_2		
2	FN_2	TP_2	FN_2	FN_2
3		FP_2		
U		FP_2		

(b)

	1	2	3	U
1			FP_3	
2			FP_3	
3	FN_3	FN_3	TP_3	FN_3
U			FP_3	

(c)

Fig. 2 Open-set confusion matrix. Example related to the computation of the precision and recall measures from an open-set confusion matrix with three available classes and unknown samples (U). The FN_i in the column U and FP_i in the row U account for false unknown and false known, respectively. The cell in the intersection of column U and row U is *not* regarded as true positive in the same way it would be considered by the multiclass closed-set f-measure

and false negative samples, respectively. For these formulas, the last column and the last row of the confusion matrix refer to the unknown.

In Eqs. (6) and (7), in spite of computing the precision and recall only for the n available classes, by taking into account the false negative and the false positive, respectively, the FN and FP also consider the false unknown and false known, respectively, as illustrated in Fig. 2.

The f-measure adopted in traditional multiclass classification is invariant to the true negative (Sokolova and Lapalme 2009), i.e., the f-measure does not take into account the samples truly rejected as belonging to the class under consideration. Similarly, the proposed OSFM for open-set scenarios is invariant to the true unknown. But both the false unknown (known samples incorrectly classified as unknown) and the false known (unknown samples incorrectly classified as known) are considered in the OSFM. We adopted this strategy for two reasons: (1) The f-measure must give importance to classified known samples and (2) The unknown is not a single class but possibly several ones.

Differently from the NA, the OSFM is sensitive to the unbalancing of the classes.

4.2 Experimental setup

In this section, we present the baselines (Sect. 4.2.1); the datasets considered in the experiments (Sect. 4.2.2); and the experimental protocol adopted (Sect. 4.2.3) in this work.

4.2.1 Baselines

We compare the OSNN classifier with the multiclass SVM with one-vs-all approach using two types of grid search procedures (SVM$^{\text{MCBIN}}$ and SVM$^{\text{MCBIN}}_{\text{ext}}$); the SVM$^{\text{MCOC}}$ proposed by Pritsos and Stamatatos (2013); the DBC$^{\text{MCBIN}}$ of Costa et al. (2012, 2014); the 1VS proposed by Scheirer et al. (2013); the WSVM method of Scheirer et al. (2014); the traditional NN; and the NN using thresholds with two types of grid search procedures to find out the threshold (TNN and TNN$_{\text{ext}}$). Besides these classifiers, we consider the OSNN$^{\text{CV}}$ as a baseline for comparison with OSNN to show the effectiveness of the ability of OSNN to reject faraway samples.

In addition to the well-known C and γ parameters of the SVM, there are other configurations that influence the behavior of the classifier. According to Chang and Lin (2011), there are two possible ways to accomplish the grid search for a multiclass binary-based classifier such as the $\text{SVM}^{\text{MCBIN}}$: (1) the *external* and (2) the *internal grid search*.[4] In the external approach, the grid search is performed in the multiclass level forcing all the binary classifiers to share the same parameters. On the other hand, in the internal grid search, each binary classifier performs its own grid search. According to Chen et al. (2005), considering the one-vs-one approach, the external approach obtains parameters not uniformly good to every binary classifier. In addition, it considers the overall accuracy of the multiclass classifier. On the other hand, the internal grid search can over-fit the classifier. However, for closed-set, they result in similar accuracy (Chen et al. 2005).

We used both external and internal grid search approaches for SVM in comparison to the proposed method herein. These approaches are implemented in $\text{SVM}^{\text{MCBIN}}_{\text{ext}}$ and $\text{SVM}^{\text{MCBIN}}$, respectively. Both classifiers use Radial Basis Function (RBF) kernel. Both 1VS and WSVM use the external grid search, according to the authors. All other SVM-based baselines use the internal grid search.

Regarding the 1VS, recently Scheirer et al. (2013) released a new version of the code (more efficient) in their website. We use this new version of the code because it already has the implementation of the one-vs-all approach.[5]

The WSVM implementation was also accomplished using the one-vs-all approach. For this baseline, the authors kindly provided an implementation containing the multiclass-from-binary version of the classifier.[6] Then, we used the C code implementation as provided. We fixed the threshold δ_τ for the CAP model in WSVM in 0.001, as specified by the authors (Scheirer et al. 2014), and we performed a grid search in $\{2^{-7}, 2^{-6}, \ldots, 2^{0}\}$ for the threshold δ_R. Regarding the SVM parameters, we performed grid search for $C \in \{2^{-5}, 2^{-3}, \ldots, 2^{15}\}$ and $\gamma \in \{2^{-15}, 2^{-13}, \ldots, 2^{3}\}$ following the standard protocol in the literature (Chang and Lin 2011). In summary, we performed a tridimensional grid search on the parameters C, γ, and δ_R.

The TNN classifier is simply the NN with a threshold T on the distance to the nearest neighbor. When the distance to the nearest neighbor is greater than T, the test sample is classified as unknown. In the grid search for T, we try values from 0 to $\sqrt{D}$ with 100 linearly separated values, in which D is the number of features of the dataset. In our experiments, all feature values of the dataset were normalized between 0 and 1 (the same normalization for all classifiers).

In Table 1, we summarize the evaluated methods. We say that a method is open set when it somehow allows the classification of a test sample as unknown.

4.2.2 Datasets

We performed experiments considering the following six datasets.

- In the 15-Scenes (Lazebnik et al. 2006) dataset, with 15 classes, the 4485 images were represented by a bag-of-visual-word vector created with soft assignment (van Gemert et al. 2010) and max pooling (Boureau et al. 2010), based on a codebook of 1000 Scale Invariant Feature Transform (SIFT; Lowe 2004) codewords.
- The 26 classes of the Letter (Frey and Slate 1991; Michie et al. 1994) dataset represent the letters of the English alphabet (black-and-white rectangular pixel displays). The 20,000 samples contain 16 attributes.

[4] We defined these names.

[5] The code is available in http://www.metarecognition.com/openset/ (as of October 2016).

[6] The code is available in http://www.metarecognition.com/open-set-with-kernels/ (as of October 2016).

Table 1 General characteristics of the classifiers used in the experiments

Method	Approach	Open-set	Kernel	Grid search
$\text{SVM}^{\text{MCBIN}}$	One-vs-all	✓	RBF	Internal
$\text{SVM}^{\text{MCBIN}}_{\text{ext}}$	One-vs-all	✓	RBF	External
SVM^{MCOC}	One-class based	✓	RBF	Internal
$\text{DBC}^{\text{MCBIN}}$	One-vs-all	✓	RBF	Internal
1VS	One-vs-all	✓	Linear	External
WSVM	One-vs-all	✓	RBF	External
NN	Multiclass	✗	–	–
TNN	Multiclass	✓	–	Internal
TNN_{ext}	Multiclass	✓	–	External
OSNN^{CV}	Multiclass	✓	–	–
OSNN	Multiclass	✓	–	External

- The Auslan (Kadous 2002) dataset contains 95 classes of Australian Sign Language (Auslan) signs collected from a volunteer native Auslan signer (Kadous 2002). Data was acquired using two Fifth Dimension Technologies (5DT) gloves hardware and two Ascension Flock-of-Birds magnetic position trackers. There are 146,949 samples represented with 22 features (x, y, z positions, bend measures, etc).
- The Caltech-256 (Griffin et al. 2007) dataset comprises 256 object classes. The feature vectors consider a bag-of-visual-words characterization approach and contain 1000 features, acquired with dense sampling, SIFT descriptor for the points of interest, hard assignment (van Gemert et al. 2010), and average pooling (Boureau et al. 2010). In total, there are 29,780 samples.
- The ALOI (Geusebroek et al. 2005) dataset has 1000 classes and 108 samples for each class (108,000 in total). The features were extracted with the Border/Interior Pixel Classification (BIC; Stehling et al. 2002) descriptor and contain 128 dimensions.
- The Ukbench (Nist and Stew 2006) dataset comprises 2550 classes of four images each. In our work, the images were represented with BIC descriptor (128 dimensions).

In Table 2, we present the overall characteristics of the datasets we used in the experiments. Note that we did not try to find the best characterization approach for each dataset since this is not the focus of this work. We relied on characteristics that presented good results according to prior work in the literature. In addition, all of the used features are available on https://dx.doi.org/10.6084/m9.figshare.1097614.

4.2.3 Experimental protocol

We performed experiments on all of these datasets by training each classifier with $n = 3$, 6, 9, and 12 classes available for training, among the total number of classes of each dataset. Each experiment consists of a combination of a classifier, a dataset, and a set of n available classes. For each experiment, we

1. randomly choose n available classes for training;
2. consider half of the known samples in each of the n classes for testing and half for training;

Table 2 General characteristics of the datasets used in the experiments

Dataset	No. of samples	No. of classes	No. of features	No. of samples/class
15-Scenes	4485	15	1000	299
Letter	20,000	26	16	769
Auslan	146,949	95	22	1546
Caltech-256	29,780	256	1000	116
ALOI	108,000	1000	128	108
Ukbench	10,200	2550	128	4

3. consider the samples of the other classes as unknown for testing; and
4. acquire results based on the previously mentioned measures (Sect. 4.1).

In this work, we also adopt the concept of *openness* of a problem[7] as defined by Scheirer et al. (2013).

$$\text{openness} = 1 - \sqrt{\frac{|\text{training classes}|}{|\text{testing classes}|}}. \tag{8}$$

Note that the more available classes for training, the less open the classification problem. The considered classification scenarios are "very open". For example, for 3 available classes for training, the openness for the 15-Scenes, Letter, Auslan, Caltech-256, ALOI, and Ukbench are 0.55, 0.66, 0.82, 0.89, 0.94, and 0.96, respectively. The exception is for the 15-Scenes and the Letter datasets, for which, considering 12 available classes, the openness is 0.11 and 0.32, respectively.

To verify the effectiveness of the proposed classifier, we performed a Pairwise Wilcoxon test with Bonferroni's correction (Demšar 2006) with 95% of confidence. For each experiment (a combination of classifier, dataset, and number of available classes), we repeated the experiments 10 times with different sets of available classes.

4.3 Results

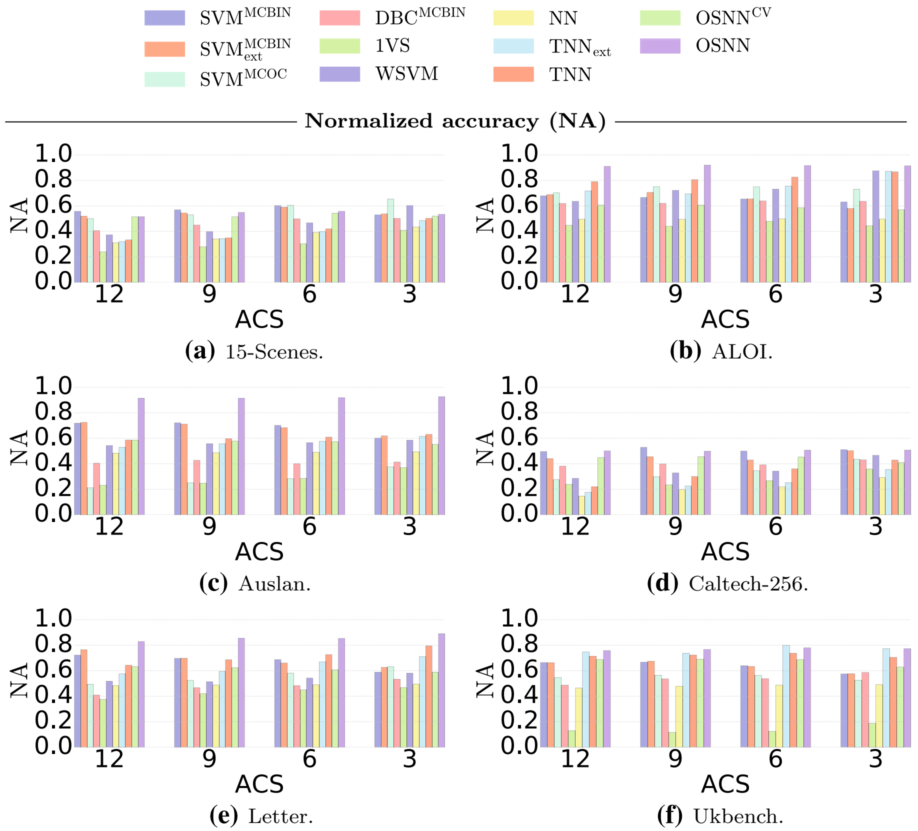
In this section, we show a comparison of the classification performance of the chosen eleven methods in the six datasets (Sect. 4.3.1) and we analyze the *decision boundaries* of the classifiers in synthetic datasets (Sect. 4.3.2).

4.3.1 Classification performance

Figures 3, 4, 5, 6 and 7 depict the results considering the NA (with $\lambda_r = 0.5$), OSFM$_M$, OSFM$_\mu$, AKS, and AUS, respectively, for all datasets considering the classifiers trained with 3, 6, 9, and 12 classes and comparing SVM$^{\text{MCBIN}}$, SVM$^{\text{MCBIN}}_{\text{ext}}$, SVM$^{\text{MCOC}}$, DBC$^{\text{MCBIN}}$, 1VS, WSVM, NN, TNN, TNN$_{\text{ext}}$, OSNN$^{\text{CV}}$, and OSNN classifiers.

In Figs. 3, 4 and 5, we can see that for the ALOI, Auslan, Letter, and Ukbench, the OSNN obtained the best results in general, for the evaluated measures. For the 15-Scenes and Caltech-256, the OSNN does not perform as well as the other classifiers, when considering,

[7] The *openness* measure serves only for evaluating open-set solutions in academic terms since in practice it might not be possible to even estimate the actual number of classes. The measure presented here is an adaptation of the measure presented in Scheirer et al. (2013).



(a) 15-Scenes.

(b) ALOI.

(c) Auslan.

(d) Caltech-256.

(e) Letter.

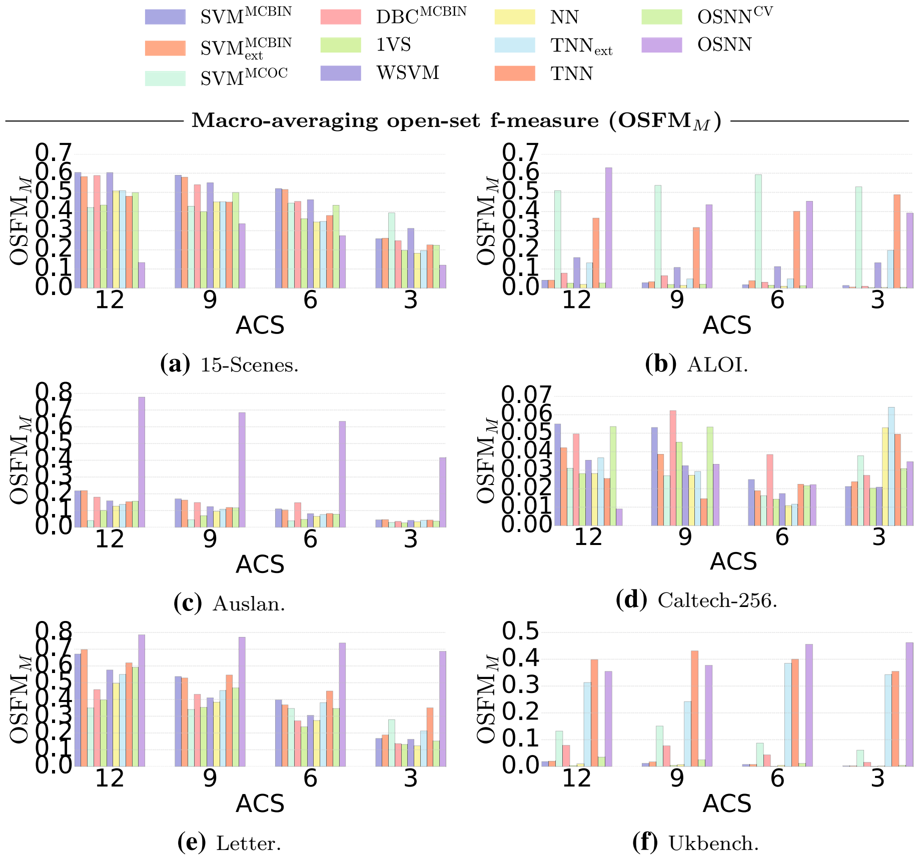
(f) Ukbench.

Fig. 3 Results for all datasets regarding *normalized accuracy* (NA) considering 3, 6, 9, and 12 *available classes* (ACS) (Color figure online)

for instance, the OSFM in Fig. 5. However, we can see in Table 3 that, overall, the OSNN obtained results statistically better than the other classifiers regarding these measures.

Regarding the AKS in Fig. 6, we also present two kinds of error included in the error on known samples (EKS $= 1 - $AKS): the misclassification (MIS) and the false unknown (FU). The MIS refers to the samples classified as belonging to the wrong class. The FU refers to the samples wrongly classified as unknown. We can see in those figures that the main reason of the smaller AKS for the OSNN is due false unknown classifications. When the OSNN classifies a test sample as one of the known classes, it is surer about the classification compared to the other classifiers. The other methods, in general, have a bias for classifying as known, as evinced by the high MIS in Fig. 6 and the small AUS in Fig. 7, compared to the OSNN. Furthermore, regarding the NA, the OSNN is stable to the variation of openness.

In Table 3, we present the statistical comparison between OSNN and the baselines using the non-parametric Pairwise Wilcoxon test with Bonferroni's correction (Demšar 2006). We see that OSNN obtained better results with statistical difference in most of the cases. Regarding the NA, OSFM$_M$, OSFM$_\mu$, and AUS, the OSNN did not get worse results in any case. Despite having worse results regarding AKS in some cases, the OSNN obtained better results in all cases regarding AUS.



(a) 15-Scenes.

(b) ALOI.

(c) Auslan.

(d) Caltech-256.

(e) Letter.

(f) Ukbench.

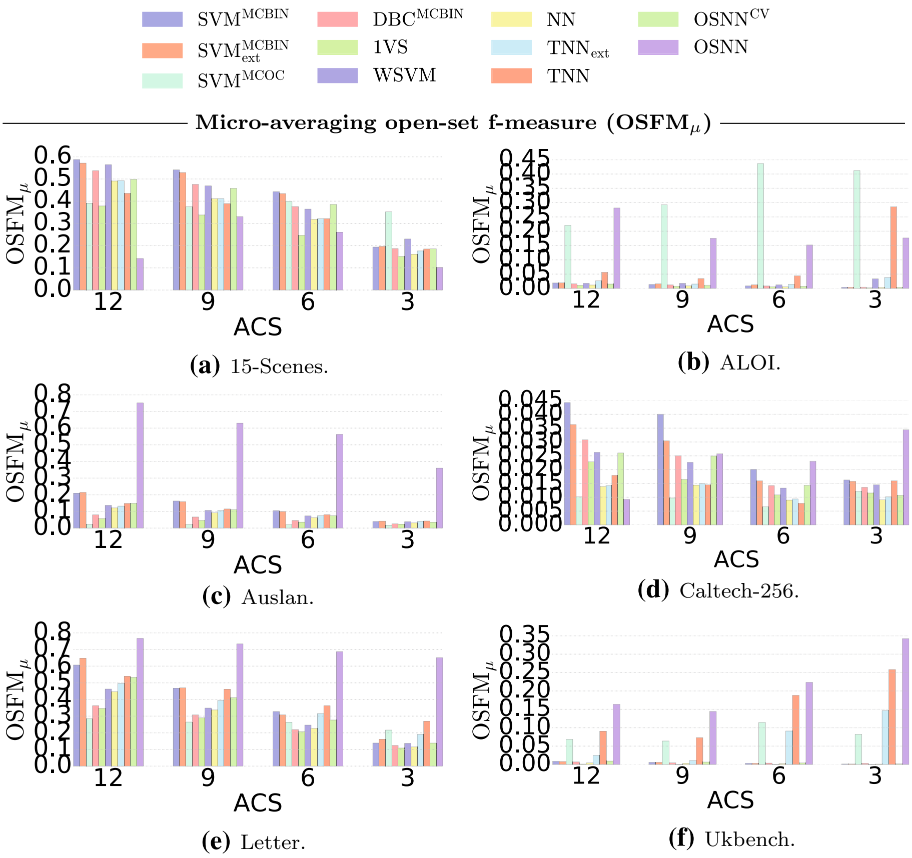
Fig. 4 Results for all datasets regarding *macro-averaging open-set f-measure* (OSFM_M) considering 3, 6, 9, and 12 *available classes* (ACS) (Color figure online)

We do not present the WSVM classifier in Table 3 because it is not able to run on Ukbench dataset; note in Table 2 that Ukbench has only four samples per class (two of them are for training) and the WSVM is not able to generate the model with less than three samples per class (Scheirer et al. 2014). Comparing the OSNN with the WSVM, the OSNN obtains better results, with statistical difference, for NA, OSFM_M, OSFM_μ, and AUS for 3, 6, 9, and 12 available classes. Regarding the AKS, WSVM obtained better results in all cases.

4.3.2 Analysis of decision boundaries

Aiming at visually understanding the different behavior of the classifiers, we also performed tests on a 2-dimensional synthetic dataset using the Four-gauss (Kuncheva and Hadjitodorov 2004) dataset. We trained the classifiers using all samples of the dataset to plot the decision boundaries for each class. The decision boundary of a class defines the region in which a possible test sample will be classified as belonging to that class.

In Fig. 8, we present the decision boundaries for the Four-gauss dataset. Based on these figures, we can note that OSNN (Fig. 8k) successfully bounds the KLOS. While the $\mathrm{SVM}^{\mathrm{MCBIN}}$ (Fig. 8a, b) is able to classify as unknown only the doubtful samples among the available



(a) 15-Scenes.

(b) ALOI.

(c) Auslan.

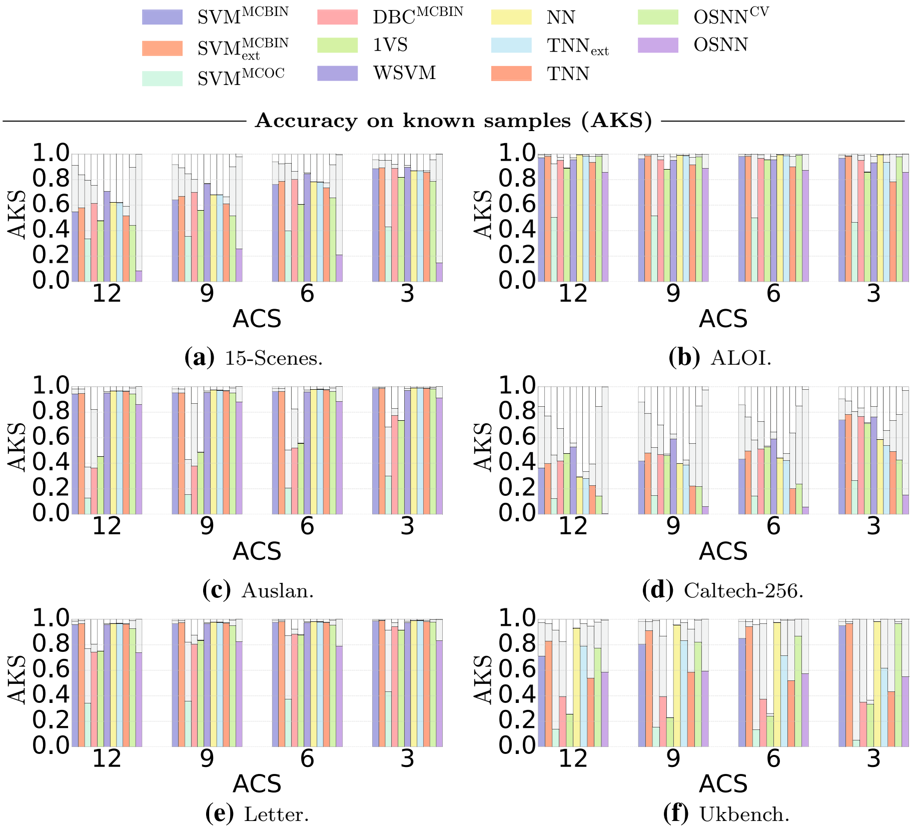
(d) Caltech-256.

(e) Letter.

(f) Ukbench.

Fig. 5 Results for all datasets regarding *micro-averaging open-set f-measure* (OSFM$_\mu$) considering 3, 6, 9, and 12 *available classes* (ACS) (Color figure online)

classes, OSNN also avoids recognizing the faraway samples. As expected, we can see that SVM^{MCOC} (Fig. 8c) is very specialized. It makes the classifier obtain a good AUS, however the performance regarding the AKS is affected. Overall, its NA (Fig. 3) is also affected. The DBC^{MCBIN} (Fig. 8d) has a behavior similar to the SVM^{MCBIN}. As it changes the original position of the hyperplane independently for each binary classifier, we can see, in Fig. 8d, that the final decision of some binary classifiers predominates over others. As expected, the 1VS (Fig. 8e) bounds each class with two hyperplanes, however it maintains an unbounded KLOS. As the WSVM (Fig. 8f) is based on the OCSVM, we can see a specialized behavior in these figures, however for the high-dimensional datasets, the AUS (Fig. 7) is not as good. For the NN (Fig. 8g) classifier, there is no white region, i.e., no test sample is classified as unknown. The TNN and TNN$_{ext}$ (Fig. 8h, i) also presented interesting behaviors in the 2-dimensional dataset, however it is well known that in high-dimensional spaces it is difficult to obtain a reasonable threshold (Phillips et al. 2011). Finally, we can see that the OSNN^{CV} (Fig. 8j) is able to classify, as unknown, only doubtful test samples, consequently it is not able to bound the KLOS. In these figures, we can see that most of the baseline classifiers, including some of the proposed for the open-set scenario (DBC^{MCBIN} and 1VS), leave an unbounded KLOS.



(a) 15-Scenes. **(b)** ALOI. **(c)** Auslan. **(d)** Caltech-256. **(e)** Letter. **(f)** Ukbench.

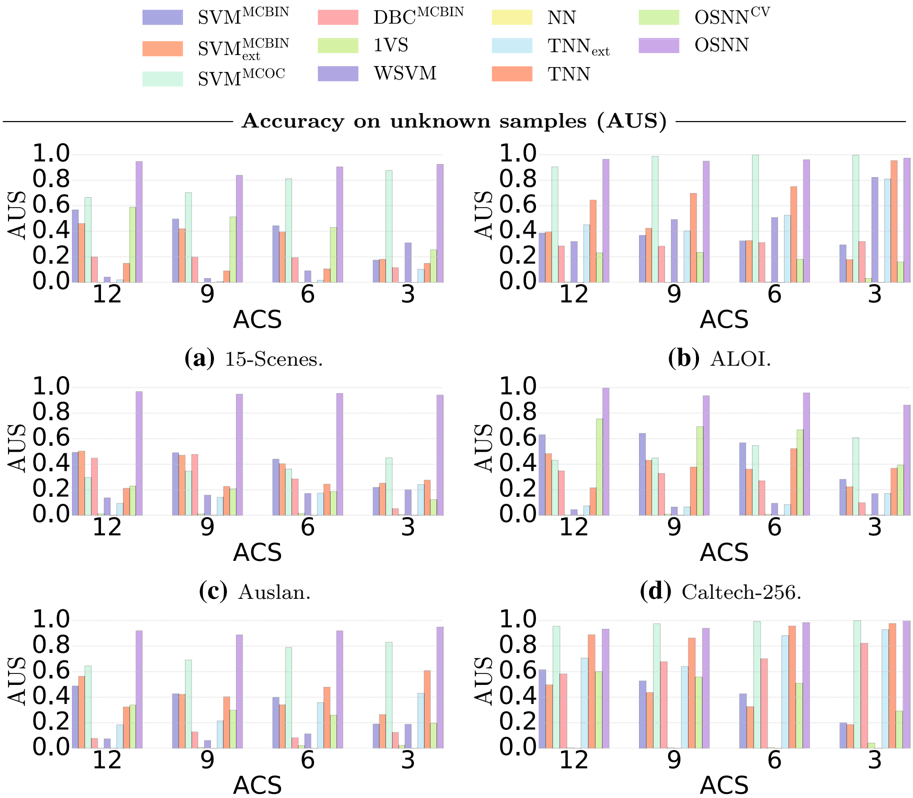
Fig. 6 Results for all datasets regarding *accuracy on known samples* (AKS) considering 3, 6, 9, and 12 *available classes* (ACS). The *graphs* depicting AKS also depicts two kinds of error included in the error on known samples (EKS $= 1 - $ AKS): the misclassification (MIS) and the false unknown (FU). The MIS and the FU are depicted in the *white* and *gray bars*, respectively (Color figure online)

4.4 Remarks on the OSNN's parameter optimization

The parameter optimization of the OSNN, described in Sect. 3, performs a simulation of the open-set scenario based on the available samples for training to better estimate the threshold for rejecting unknown samples. In this section, we evaluate some details about the parameter optimization process. First, in Sect. 4.4.1, we evaluate the influence on the behavior of the OSNN when using a regularization constant λ_r other than 50% for the NA during the parameter optimization process. Then, in Sect. 4.4.2, we evaluate the influence of the amount of classes considered as unknown during the parameter optimization process.

4.4.1 Influence of the regularization constant

Despite training the OSNN using the NA with λ_r set to 50%, as shown in the experiments in Sect. 4.3, we also evaluated the performance of the classifier for setting λ_r to 10, 30, 70, and 90%. In Table 4, we compare the OSNN trained with λ_r set to 50% (simply referred to



(a) 15-Scenes. **(b)** ALOI.

(c) Auslan. **(d)** Caltech-256.

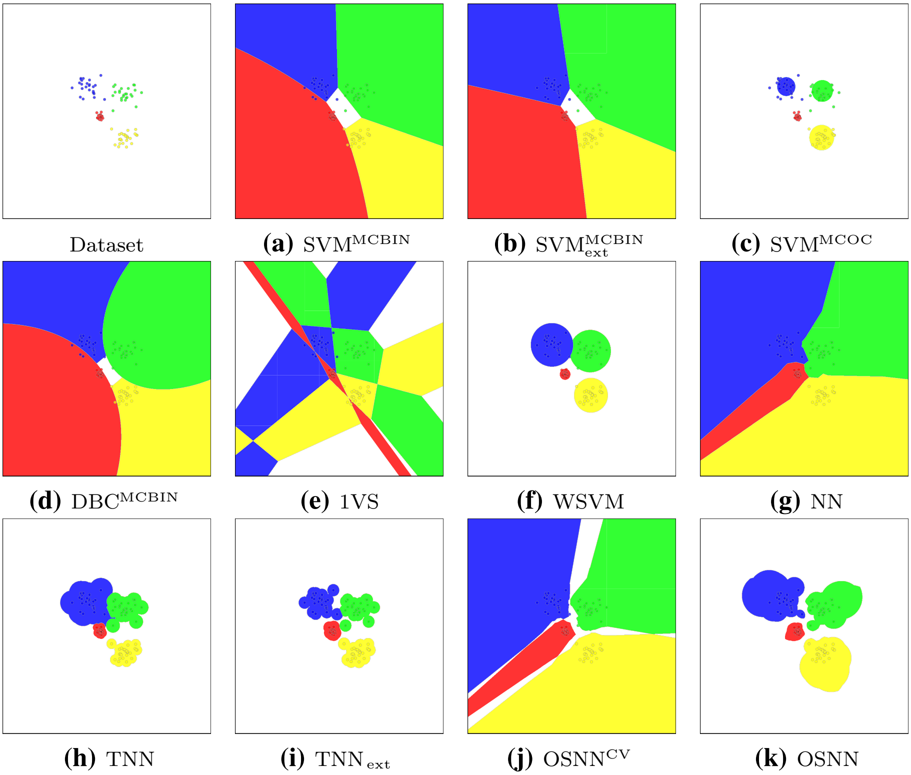
(e) Letter. **(f)** Ukbench.

Fig. 7 Results for all datasets regarding *accuracy on unknown samples* (AUS) considering 3, 6, 9, and 12 *available classes* (ACS) (Color figure online)

Table 3 Statistical tests for all datasets (15-Scenes, ALOI, Auslan, Caltech-256, Letter, and Ukbench) for the measures *normalized accuracy* (NA), *macro-averaging open-set f-measure* (OSFM$_M$), *micro-averaging open-set f-measure* (OSFM$_\mu$), *accuracy on known samples* (AKS), and *accuracy on unknown samples* (AUS), comparing the OSNN with the other methods

Measure	SVM^{MCBIN}	SVM$^{MCBIN}_{ext}$	SVM^{MCOC}	DBC^{MCBIN}	1VS	NN	TNN$_{ext}$	TNN	OSNN^{CV}
NA	wwww	wwww	wwww	wwww	wwww	wwww	wwww	wwww	wwww
OSFM$_M$	wwww	wwww	.www	wwww	wwww	wwww	www.	.ww.	wwww
OSFM$_\mu$	www.	www.	..ww	wwww	wwww	wwww	wwww	.ww.	wwww
AKS	1111	1111	wwww	1...	....	1111	1111	1111	1111
AUS	wwww	wwww	wwww	wwww	wwww	wwww	wwww	wwww	wwww

Each cell presents the comparison for 3, 6, 9, and 12 available classes. w indicates that the OSNN has better results with statistical difference and 1 indicates that the OSNN has worse results with statistical difference. The dot indicates there is no statistical difference between OSNN and the method in the column


Dataset

(a) SVM$^{\text{MCBIN}}$

(b) SVM$^{\text{MCBIN}}_{\text{ext}}$

(c) SVM$^{\text{MCOC}}$

(d) DBC$^{\text{MCBIN}}$

(e) 1VS

(f) WSVM

(g) NN

(h) TNN

(i) TNN$_{\text{ext}}$

(j) OSNN$^{\text{CV}}$

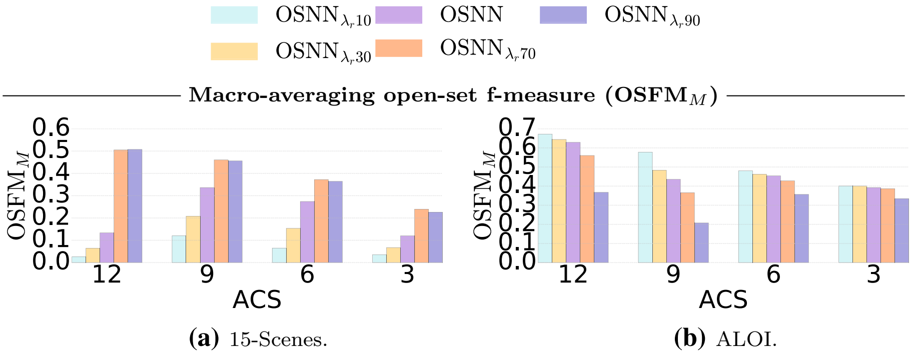
(k) OSNN

Fig. 8 Decision boundaries for the Four-gauss dataset (depicted in the *top-left image*). The *non-white regions* represent the region in which a test sample would be classified as belonging to the same class of the samples with the *same color*. All samples in the *white regions* would be classified as unknown (Color figure online)

Table 4 Statistical tests for all datasets (15-Scenes, ALOI, Auslan, Caltech-256, Letter, and Ukbench) for the measures *normalized accuracy* (NA), *macro-averaging open-set f-measure* (OSFM$_M$), *micro-averaging open-set f-measure* (OSFM$_\mu$), *accuracy on known samples* (AKS), and *accuracy on unknown samples* (AUS), comparing the OSNN with OSNN trained with regularization constant λ_r other than 50% during the parameter optimization

Measure	OSNN$_{\lambda_r 10}$	OSNN$_{\lambda_r 30}$	OSNN$_{\lambda_r 70}$	OSNN$_{\lambda_r 90}$
NA	.www	.w.w	..w.	wwww
OSFM$_M$	....	....	....	www.
OSFM$_\mu$	....	....	....	www.
AKS	wwww	wwww	llll	llll
AUS	llll	llll	wwww	wwww

Each cell presents the comparison for 3, 6, 9, and 12 available classes. w indicates that the OSNN has better results with statistical difference and l indicates that the OSNN has worse results with statistical difference. The dot indicates there is no statistical difference between OSNN and the method in the column



(a) 15-Scenes. **(b)** ALOI.

Fig. 9 Results for 15-Scenes and ALOI regarding *macro-averaging open-set f-measure* (OSFM$_M$) considering 3, 6, 9, and 12 *available classes* (ACS) comparing the OSNN trained with different regularization constant λ_r during the parameter optimization phase (Color figure online)

as OSNN) with the OSNN trained with other values of λ_r (referred to as OSNN$_{\lambda_r n}$, in which n is the percentage set for λ_r).

As expected, the OSNN obtains better results, with statistical difference in several cases regarding the NA with λ_r set to 50%. Interestingly, only the OSNN$_{\lambda_r 90}$ classifier obtains worse results with statistical difference for OSFM$_M$ and OSFM$_\mu$ measures. We can observe the influence of λ_r during training at the AKS and AUS performances. Remember that λ_r is proportional to the AKS in Eq. (4). We can see that OSNN obtains better AKS than OSNN$_{\lambda_r 10}$ and OSNN$_{\lambda_r 30}$ and worse AKS than OSNN$_{\lambda_r 70}$ and OSNN$_{\lambda_r 90}$. Regarding the AUS, it is the inverse: OSNN obtains worse results than OSNN$_{\lambda_r 10}$ and OSNN$_{\lambda_r 30}$ and better results than OSNN$_{\lambda_r 70}$ and OSNN$_{\lambda_r 90}$. Those results on AKS and AUS show that OSNN is sensitive to λ_r used for training and it can be properly set for training for specific problems, in case the openness of the problem can be estimated a priori. This is confirmed in Fig. 9, in which OSNN$_{\lambda_r 70}$ and OSNN$_{\lambda_r 90}$ have better results in the dataset with small openness (15-Scenes in Fig. 9a; 15 classes) and OSNN$_{\lambda_r 10}$ has better results in the dataset with high openness (ALOI in Fig. 9b; 1000 classes). The results are presented therein for OSFM$_M$, but a similar trend can be observed for those classifiers when considering OSFM$_\mu$ as well.

4.4.2 Amount of unknown classes in the simulation

The parameter optimization performed by the OSNN, for the results presented in Sect. 4.3, relies on a simulation of the open-set scenario by selecting half of the available classes for training and considering them as unknown in the fitting set, making it to appear only in the validation set used to estimate the better threshold T. Herein, we verify the influence on the classifier for considering other amounts of the available classes as unknown during the parameter optimization. These experiments were performed only for the open-set scenario considering 12 available classes for training. We trained different versions of the OSNN, each of which selects from 9 to 0 the number of available classes to be considered as unknown. Recall that "0" in this case refers to a closed-set training setup. We refer to those versions as OSNN$_n$, for $n = 0, \dots, 9$, in which n is the amount of the available classes considered as unknown during the parameter optimization. According to our experiments, for all six datasets used in this work, there is no statistical difference among OSNN$_n$ for $n = 1, \dots, 9$. However, the OSNN$_0$ (that does not simulate the open-set scenario) has worse results with statistical difference compared to OSNN$_n$ for $n = 1, \ldots, 9$ for NA, OSFM$_M$, and OSFM$_\mu$ in almost all cases. The exception is for OSNN$_9$, that has no statistical difference compared to OSNN$_0$ for OSFM$_M$ and OSFM$_\mu$. Regarding the AKS and AUS, the OSNN$_n$ for $n = 1, \ldots, 9$ have statistical difference among them in few cases. The OSNN$_0$, however, obtains better AKS and worse AUS compared to all OSNN$_n$ for $n = 1, \ldots, 9$ in all cases with statistical difference. In summary, the take-home lesson here is that it is important to consider at least one of the available classes as unknown during the parameter optimization stage of the OSNN.

With these results, one could wonder whether the OSNN's effectiveness is only due to the open-set simulation performed in its parameter optimization stage and that using the same parameter optimization for a simple classifier such as TNN would make it perform similarly to OSNN. To clarify this point, we implemented the parameter optimization stage for TNN$_{\text{ext}}$ and compared to the OSNN. We chose TNN$_{\text{ext}}$ over TNN because its parameter optimization is designed for external grid search as in OSNN. The results show that OSNN outperforms TNN$_{\text{ext}}$, with statistical difference, in virtually all cases considering NA, OSFM$_M$, OSFM$_\mu$, and AUS. Only regarding NA for 3 available classes there is no statistical significance. If we consider AKS, their performance is not statistically significant in the majority of the cases (for 3, 9, and 12 available classes). These results clearly show that the OSNN's performance is not only due to the parameter optimization stage as one could think at a first glance.

4.5 Failing cases of OSNN

As we could see in Fig. 6, in general, the OSNN obtained slightly smaller AKS compared to the baselines. This is due to its FU rate, because the OSNN also classifies as unknown a doubtful test sample, i.e., a test sample in the overlapping region of two or more classes. It happens because the ratio R also approaches 1 in such cases. To overcome this problem, we must identify when a test sample is being classified as unknown by doubt or because it is faraway from the training samples. In this section, we show some direct approaches trying to differentiate these two cases.

A first direct approach for this verification is to obtain the *typical distance* for each class. In this case, when the OSNN classifies as one of the known classes, that casts its final decision. Then when the OSNN classifies as unknown, we compare the distance from the test sample to the nearest neighbor with the typical distance of the nearest neighbor's class aiming to identify if the test sample is being classified as unknown by doubt or because it is faraway from the training samples and probably belongs to an unknown class. If the distance to the nearest neighbor is smaller than or equal to the typical distance of the class, then probably the classification will be as unknown by doubt. In this case, instead of classifying as unknown, we classify as belonging to the most probable class, i.e., the class of the nearest neighbor. If the distance is greater than the typical distance of the class, it is really considered unknown.

We need to define what would be the typical distance for each class. Below we list some possible reasonable definitions:

Max-Min For each class, the typical distance is the *maximum* of the minimum distances, i.e., for each training sample of the class, we get the distance to its nearest neighbor and then the *maximum* of those distances.

Mean-Min For each class, the typical distance is the *mean* of the minimum distances, i.e., for each training sample of the class, we get the distance to its nearest neighbor and then we calculate the *mean* of those distances.

Median-Min For each class, the typical distance is the *median* of the minimum distances, i.e., for each training sample of the class, we get the distance to its nearest neighbor and then we calculate the *median* of those distances.

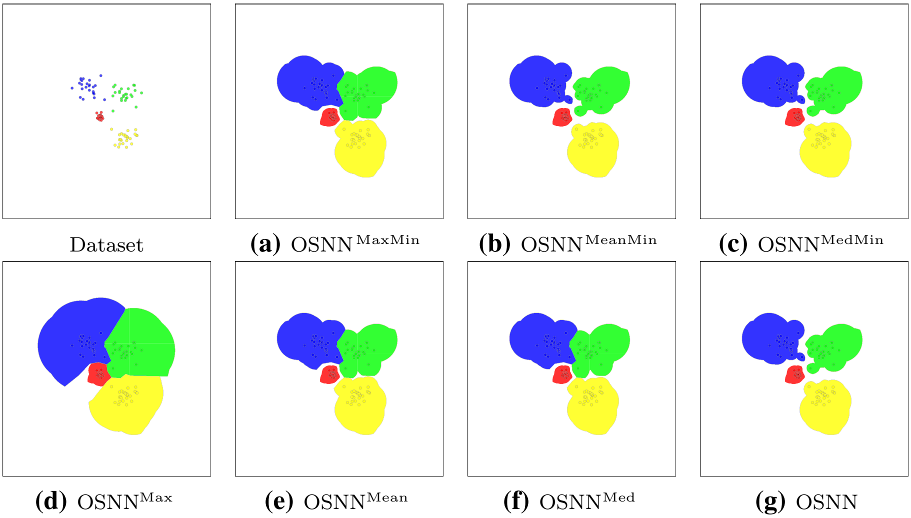


Fig. 10 Decision boundaries for the Four-gauss dataset (depicted in the *top-left image*). The *non-white regions* represent the region in which a test sample would be classified as belonging to the same class of the samples with the *same color*. All samples in the *white regions* would be classified as unknown (Color figure online)

Max For each class, the typical distance is the *maximum* of the distances among the samples, i.e., we calculate the distance from each sample of the class to each other sample and then we calculate the *maximum* of those distances.

Mean For each class, the typical distance is the *mean* of the distances among the samples, i.e., we calculate the distance from each sample of the class to each other sample and then we calculate the *mean* of those distances.

Median For each class, the typical distance is the *median* of the distances among the samples, i.e., we calculate the distance from each sample of the class to each other sample and then we calculate the *median* of those distances.

We refer to these extensions of the OSNN classifier that implement the typical distances listed above as OSNN$^{\text{MaxMin}}$, OSNN$^{\text{MeanMin}}$, OSNN$^{\text{MedMin}}$, OSNN$^{\text{Max}}$, OSNN$^{\text{Mean}}$, and OSNN$^{\text{Med}}$, respectively.

We can see in Table 5 that those modifications are able to improve the AKS with statistical difference but its performance regarding AUS is worse with statistical difference. We also see that OSNN$^{\text{MeanMin}}$ and OSNN$^{\text{MedMin}}$ have statistical equivalence with OSNN regarding the NA, OSFM$_M$, and OSFM$_\mu$ in most of the cases. The typical distances considered for the OSNN$^{\text{MeanMin}}$ and OSNN$^{\text{MedMin}}$ are the smallest ones compared to the typical distances of the other modifications (OSNN$^{\text{MaxMin}}$, OSNN$^{\text{Max}}$, OSNN$^{\text{Mean}}$, and OSNN$^{\text{Med}}$). We can see in Fig. 10 that the decision boundaries defined by the OSNN$^{\text{MeanMin}}$ (Fig. 10b), OSNN$^{\text{MedMin}}$ (Fig. 10c), and OSNN (Fig. 10g) are similar while the other modifications have greater changes in their behavior. It means that the additional verification, compared to OSNN, takes effect only in a few cases.

One could say that then the OSNN$^{\text{MeanMin}}$ or the OSNN$^{\text{MedMin}}$ are better than OSNN because the results presented in Table 5 are not clear regarding the comparison among these classifiers. However, if we compare one of these methods, e.g., the OSNN$^{\text{MedMin}}$, with the baselines used in the paper, we can see that it is *not* as good as the OSNN in general. In Table 6, we compare the OSNN$^{\text{MedMin}}$ with the baselines. The same comparison between the OSNN and

Table 5 Statistical tests for all datasets (15-Scenes, ALOI, Auslan, Caltech-256, Letter, and Ukbench) for the measures *normalized accuracy* (NA), *macro-averaging open-set f-measure* (OSFM$_M$), *micro-averaging open-set f-measure* (OSFM$_\mu$), *accuracy on known samples* (AKS), and *accuracy on unknown samples* (AUS), comparing the OSNN with the variations of OSNN aiming at avoiding false unknown classifications

Measure	OSNN$^{\text{MaxMin}}$	OSNN$^{\text{MeanMin}}$	OSNN$^{\text{MedMin}}$	OSNN$^{\text{Max}}$	OSNN$^{\text{Mean}}$	OSNN$^{\text{Med}}$
NA	wwww	....	....	wwww	wwww	wwww
OSFM$_M$	www.	....	...l	wwww	www.	www.
OSFM$_\mu$	wwww	....	....	wwww	wwww	wwww
AKS	llll	llll	llll	llll	llll	llll
AUS	wwww	wwww	wwww	wwww	wwww	wwww

Each cell presents the comparison for 3, 6, 9, and 12 available classes. w indicates that the OSNN has better results with statistical difference and l indicates that the OSNN has worse results with statistical difference. The dot indicates there is no statistical difference between OSNN and the method in the column

Table 6 Statistical tests for all datasets (15-Scenes, ALOI, Auslan, Caltech-256, Letter, and Ukbench) for the measures *normalized accuracy* (NA), *macro-averaging open-set f-measure* (OSFM$_M$), *micro-averaging open-set f-measure* (OSFM$_\mu$), *accuracy on known samples* (AKS), and *accuracy on unknown samples* (AUS), comparing the OSNN$^{\text{MedMin}}$ with the other methods

Measure	SVM$^{\text{MCBIN}}$	SVM$^{\text{MCBIN}}_{\text{ext}}$	SVM$^{\text{MCOC}}$	DBC$^{\text{MCBIN}}$	1VS	NN	TNN$_{\text{ext}}$	TNN	OSNN$^{\text{CV}}$
NA	wwww	wwww	wwww	wwww	wwww	wwww	wwww	wwww	wwww
OSFM$_M$	wwww	wwww	wwww	wwww	wwww	wwww	wwww	wwww	wwww
OSFM$_\mu$	wwww	wwww	...w	wwww	wwww	wwww	wwww	wwww	wwww
AKS	llll	llll	wwww	....	....	llll	llll	....	llll
AUS	wwww	wwww	....	wwww	wwww	wwww	wwww	wwww	wwww

Each cell presents the comparison for 3, 6, 9, and 12 available classes. w indicates that the OSNN$^{\text{MedMin}}$ has better results with statistical difference and l indicates that the OSNN$^{\text{MedMin}}$ has worse results with statistical difference. The dot indicates there is no statistical difference between OSNN$^{\text{MedMin}}$ and the method in the column

the baselines was presented in Table 3. Comparing Tables 6 and 3, we observe that OSNN and OSNN$^{\text{MedMin}}$ have exactly the same performance compared to the baselines regarding the AKS in most of the cases. Regarding the other measures, the comparison of the two tables indicates the OSNN has more cases with better results with statistical difference compared to the baselines. Therefore, we conclude that the OSNN outperforms OSNN$^{\text{MedMin}}$ for open-set scenarios.

This conclusion does not mean the OSNN$^{\text{MedMin}}$ and the OSNN$^{\text{MeanMin}}$ are not suitable for open-set scenarios. Depending on the application, one would require an open-set classifier that performs slightly better regarding the AKS regardless of some performance drop in the AUS and the other measures. Summing up, those modifications, in fact, improve OSNN's performance on AKS, however at the cost of substantially decreasing the AUS. We believe the main reason for the worse performance, in general, was because we started working with the distance itself (the typical distance comparison) in the Euclidean space instead of only the ratio of distances. Therefore, avoiding the problem of the false unknown of the OSNN is a research topic worth pursuing in the future.

5 Conclusions

Only in the last few years open-set recognition has received the proper attention and formalization. Usually, experiments in literature are performed considering that all classes of the problem are available for training, i.e., a closed-set scenario. However, in real-world situations, the amount of classes during test is many times larger than the known classes. That means that real systems must be able to deal with unknown elements that appear only during the system use and not during its development. In this work, we have two main contributions:

– The introduction of the Nearest Neighbor Distance Ratio (NNDR)-based Open-Set NN (OSNN) classifier; and
– Two new evaluation measures, *normalized accuracy* (NA) and *open-set f-measure* (OSFM), to assess the quality of methods in multiclass open-set recognition problems.

The proposed open-set classifier has the advantage of being inherently multiclass (non-binary-based), while the state-of-the-art methods are multiclass from binary. As more classes are available, multiclass-from-binary classifiers loose some efficiency, while the proposed classifier is not affected by the number of classes.

Based on the results that we presented, we showed that the proposed OSNN outperforms the baseline classifiers evaluated ($\text{SVM}^{\text{MCBIN}}$, $\text{SVM}^{\text{MCBIN}}_{\text{ext}}$, SVM^{MCOC}, $\text{DBC}^{\text{MCBIN}}$, 1VS, WSVM, NN, TNN, TNN_{ext}, and OSNN^{CV}) in most of the cases for several datasets: 15-Scenes, Letter, Auslan, Caltech-256, ALOI, and Ukbench. We confirmed the superiority of the proposed method using the non-parametric Pairwise Wilcoxon test with Bonferroni's correction (Demšar 2006). As we can see in Fig. 8, the proposed OSNN can gracefully limit the *known labeled open space* (KLOS).

We showed that OSNN is sensitive to the regularization constant λ_r established for the NA during the *parameter optimization*, allowing it to be optimized for specific open-set problems in case the openness can be estimated a priori. We also showed that it is important to perform a simulation of the open-set scenario during OSNN's parameter optimization by leaving at least one of the available classes out of the fitting set used to fit an OSNN classifier to estimate the better threshold based on the validation set.

The main characteristic of the proposed method is the use of the ratio of similarity scores by applying a threshold on it instead of using the similarity score itself. According to our experiments, it is better for bounding the KLOS. Also, the proposed method is stable as the recognition scenario gets more open.

Many of the classifiers in the literature specially proposed for the open-set scenario (SVM^{MCOC}, $\text{DBC}^{\text{MCBIN}}$, 1VS, and WSVM) did not obtain good results in our experiments. 1VS and WSVM, for instance, performed worse than traditional SVM using the one-vs-all approach ($\text{SVM}^{\text{MCBIN}}$ and $\text{SVM}^{\text{MCBIN}}_{\text{ext}}$), even using the source code provided by their authors. We observed that the high specialization of the SVM^{MCOC} makes it obtain a low *accuracy on known samples* (AKS) and the final accuracy is impacted by this behavior.

Future work includes using the proposed parameter optimization for the OSNN as a general open-set grid search procedure and investigating whether this novel grid search procedure obtains better parameter estimation than the traditional grid search for general classifiers.

Another research topic worth pursuing in the future is to extend the NNDR technique to consider the ratio not only between the two closest classes to make the final classification decision. We could consider, for example, 3 or 4 classes and obtain the ratio of similarity score between pairs of these classes. More complex extensions could be investigated using Extreme Value Theory (EVT) (de Haan and Ferreira 2007). The idea is, instead of simply analyzing individual ratio, to create a model by fitting an extreme distribution on the smallest ratios. At testing, we could verify if the ratios produced by the test sample belongs or not to the distribution.

Finally, as we showed in Sect. 4.5, the false unknown (FU) rate obtained by the OSNN is mainly due to doubtful samples classified as unknown instead of one of the doubtful classes. We also showed that it is not trivial to identify when the test sample is being classified as unknown because it is faraway from the training samples or because it is in doubt between two or more training classes. Therefore, future research can also be devoted to investigating meta-recognition approaches (Scheirer et al. 2011, 2012) to develop a classifier to learn when the test sample is being classified as unknown by one or another reason.

Acknowledgements Part of the results presented in this paper were obtained through the project "Pattern recognition and classification by feature engineering, *-fusion, open-set recognition, and meta-recognition", sponsored by Samsung Eletrônica da Amazônia Ltda., in the framework of law No. 8248/91. The authors also thank Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq; Grants #304352/2012-8 and #304472/2015-8), Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) through the DeepEyes grant and for the scholarship provided to the first author, Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP; Grants #2010/05647-4, #2013/50169-1, and #2013/50155-0) and through the DéjàVu grant #2015/19222-9, and Microsoft Research for the financial support.

References

Bartlett, P. L., & Wegkamp, M. H. (2008). Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, *9*, 1823–1840.

Bishop, C. M. (2006). *Pattern recognition and machine learning* (1st ed.). New York: Information Science and Statistics, Springer.

Boureau, Y. L., Bach, F., LeCun, Y., & Ponce, J. (2010). Learning mid-level features for recognition. In *International conference on computer vision and pattern recognition* (pp. 2559–2566). San Francisco, CA: IEEE Press.

Cevikalp, H., & Triggs, B. (2012). Efficient object detection using cascades of nearest convex model classifiers. In *International conference on computer vision and pattern recognition* (pp. 3138–3145). Providence, RI: IEEE Press.

Chang, C. C., & Lin, C. J. (2011). LIBSVM: A library for support vector machines. *Transactions on Intelligent Systems and Technology*, *2*(3), 27:1–27:27.

Chen, P. H., Lin, C. J., & Schölkopf, B. (2005). A tutorial on nu-support vector machines. *Applied Stochastic Models in Business and Industry*, *21*(2), 111–136.

Chew, S. W., Lucey, S., Lucey, P., Sridharan, S., & Cohn, J. F. (2012). Improved facial expression recognition via uni-hyperplane classification. In *International conference on computer vision and pattern recognition* (pp. 2554–2561). Providence, RI: IEEE Press.

Chow, C. K. (1970). On optimum recognition error and reject tradeoff. *Transactions on Information Theory*, *16*(1), 41–46.

Costa, F.O., Eckmann, M., Scheirer, W. J., & Rocha, A. (2012). Open set source camera attribution. In *Conference on graphics, patterns, and images* (pp. 71–78). Ouro Preto: IEEE Press.

Costa, F. O., Silva, E., Eckmann, M., Scheirer, W. J., & Rocha, A. (2014). Open set source camera attribution and device linking. *Pattern Recognition Letters*, *39*, 92–101.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, *7*, 1–30.

de Haan, L., & Ferreira, A. (2007). *Extreme value theory: An introduction* (1st ed.), Springer Series in Operations Research and Financial Engineering. New York: Springer.

Dubuisson, B., & Masson, M. (1993). A statistical decision rule with incomplete knowledge about classes. *Pattern Recognition*, *26*(1), 155–165.

Frey, P. W., & Slate, D. J. (1991). Letter recognition using Holland-style adaptive classifiers. *Machine Learning*, *6*(2), 161–182.

Fukunaga, K. (1990). Hypothesis testing. In *Introduction to statistical pattern recognition, Chapter 3*, Computer Science and Scientific Computing Series, 2nd ed. (pp. 51–123). London: Academic Press.

Geusebroek, J. M., Burghouts, G. J., & Smeulders, A. W. M. (2005). The Amsterdam library of object images. *International Journal of Computer Vision*, *61*(1), 103–112.

Griffin, G., Holub, A., & Perona, P. (2007). *Caltech-256 object category dataset*. Tech. rep., California Institute of Technology.

Heflin, B., Scheirer, W. J., & Boult, T. E. (2012). Detecting and classifying scars, marks, and tattoos found in the wild. In *International conference on biometrics: Theory, applications and systems* (pp. 31–38). Arlington, VA: IEEE Press.

Hodge, V. J., & Austin, J. (2004). A survey of outlier detection methodologies. *Artificial Intelligence Review*, *22*(2), 85–126.

Jayadeva, K. R., & Chandra, S. (2007). Twin support vector machines for pattern classification. *Transactions on Pattern Analysis and Machine Intelligence*, *29*(5), 905–910.

Jin, H., Liu, Q., & Lu, H. (2004). Face detection using one-class-based support vectors. In *International conference on automatic face and gesture recognition* (pp. 457–462). Seoul: IEEE Press.

Kadous, M. W. (2002). Temporal classification: Extending the classification paradigm to multivariate time series. PhD Thesis, The University of New South Wales, New South Wales, Australia.

Kuncheva, L. I., & Hadjitodorov, S. T. (2004) Using diversity in cluster ensembles. In *International conference on systems, man and cybernetics* (Vol. 2, pp. 1214–1219). The Hague: IEEE Press.

Lazebnik, S., Schmid, C., & Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *International conference on computer vision and pattern recognition* (Vol. 2, pp. 2169–2178). New York, NY: IEEE Press.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, *60*(2), 91–110.

Malisiewicz, T., Gupta, A., & Efros, A. A. (2011). Ensemble of exemplar-svms for object detection and beyond. In *International Conference on Computer Vision* (pp. 89–96). Barcelona: IEEE Press.

Manevitz, L. M., & Yousef, M. (2002). One-class svms for document classification. *Journal of Machine Learning Research*, *2*, 139–154.

Michie, D., Spiegelhalter, D. J., & Taylor, C. C. (1994). *Machine learning, neural and statistical classification, Ellis Horwood Series in Artificial Intelligence*. Upper Saddle River, NJ: Prentice Hall.

Muzzolini, R., Yang, Y. H., & Pierson, R. (1998). Classifier design with incomplete knowledge. *Pattern Recognition*, *31*(4), 345–369.

Nist, D., & Stew, H. (2006). Scalable recognition with a vocabulary tree. In *International conference on computer vision and pattern recognition* (Vol. 2, pp. 2162–2168). New York, NY: IEEE Press.

Papa, J. P., Falcão, A. X., Miranda, P. A. V., Suzuki, C. T. N., & Mascarenhas, N. D. A. (2007). Design of robust pattern classifiers based on optimum-path forests. In *International symposium on mathematical morphology, MCT/INPE, Rio de Janeiro* (Vol. 1, pp. 337–348).

Papa, J. P., Falcão, A. X., & Suzuki, C. T. N. (2009). Supervised pattern classification based on optimum-path forest. *International Journal of Imaging Systems and Technology*, *19*(2), 120–131.

Phillips, P. J., Grother, P., & Micheals, R. (2011). Evaluation methods in face recognition. In S. Z. Li & A. K. Jain (Eds.), *Handbook of face recognition* (pp. 329–348). New York: Springer.

Pietraszek, T. (2005). Optimizing abstaining classifiers using ROC analysis. In *International conference on machine learning* (pp. 665–672). Bonn: ACM Press.

Pritsos, D. A., & Stamatatos, E. (2013). Open-set classification for automated genre identification. In P. Serdyukov, P. Braslavski, S. O. Kuznetsov, J. Kamps, S. Rüger, E. Agichtein, I. Segalovich, & E. Yilmaz (Eds.), *Advances in information retrieval, Lecture Notes in Computer Science* (Vol. 7814, pp. 207–217). Berlin: Springer.

Rocha, A., & Goldenstein, S. (2009). Multi-class from binary: Divide to conquer. In *International conference on computer vision theory and applications* (pp. 1–8). Lisboa: Springer.

Rocha, A., & Goldenstein, S. (2014). Multiclass from binary: Expanding one-vs-all, one-vs-one and ECOC-based approaches. *Transactions on Neural Networks and Learning Systems*, *25*(2), 289–302.

Scheirer, W. J., Rocha, A., Micheals, R. J., & Boult, T. E. (2011). Meta-recognition: The theory and practice of recognition score analysis. *Transactions on Pattern Analysis and Machine Intelligence*, *33*(8), 1689–1695.

Scheirer, W. J., Rocha, A. R., Parris, J., & Boult, T. E. (2012). Learning for meta-recognition. *Transactions on Information Forensics and Security*, *7*(4), 1214–1224.

Scheirer, W. J., Rocha, A. R., Sapkota, A., & Boult, T. E. (2013). Towards open set recognition. *Transactions on Pattern Analysis and Machine Intelligence*, *35*(7), 1757–1772.

Scheirer, W. J., Jain, L. P., & Boult, T. E. (2014). Probability models for open set recognition. *Transactions on Pattern Analysis and Machine Intelligence*, *36*(11), 2317–2324.

Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (1999). *Estimating the support of a high-dimensional distribution*. Tech. rep., Microsoft Research, Redmond, WA.

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing and Management*, *45*(4), 427–437.

Stehling, R. O., Nascimento, M. A., & Falcão, A. X. (2002). A compact and efficient image retrieval approach based on border/interior pixel classification. *International conference on information and knowledge management* (pp. 102–109). McLean, VA: ACM Press.

van Gemert, J. C., Veenman, C. J., Smeulders, A. W. M., & Geusebroek, J. M. (2010). Visual word ambiguity. *Transactions on Pattern Analysis and Machine Intelligence*, *32*(7), 1271–1283.

Wu, M., & Ye, J. (2009). A small sphere and large margin approach for novelty detection using training data with outliers. *Transactions on Pattern Analysis and Machine Intelligence*, *31*(11), 2088–2092.

Zhou, X. S., & Huang, T. S. (2003). Relevance feedback in image retrieval: A comprehensive review. *Multimedia Systems*, *8*(6), 536–544.