Towards Open Set Recognition

Walter J. Scheirer, *Member, IEEE,*
Anderson Rocha, *Member, IEEE,*
Archana Sapkota, *Student Member, IEEE,*
and Terrance E. Boult, *Member, IEEE*

Abstract—To date, almost all experimental evaluations of machine learning-based recognition algorithms in computer vision have taken the form of “closed set” recognition, whereby all testing classes are known at training time. A more realistic scenario for vision applications is “open set” recognition, where incomplete knowledge of the world is present at training time, and unknown classes can be submitted to an algorithm during testing. This article explores the nature of open set recognition, and formalizes its definition as a constrained minimization problem. The open set recognition problem is not well addressed by existing algorithms because it requires strong generalization. As a step towards a solution, we introduce a novel “1-vs-Set Machine,” which sculpts a decision space from the marginal distances of a 1-class or binary SVM with a linear kernel. This methodology applies to several different applications in computer vision where open set recognition is a challenging problem, including object recognition and face verification. We consider both in this work, with large scale cross-dataset experiments performed over the Caltech 256 and ImageNet sets, as well as face matching experiments performed over the Labeled Faces in the Wild set. The experiments highlight the effectiveness of machines adapted for open set evaluation compared to existing 1-class and binary SVMs for the same tasks.

Index Terms—Open Set Recognition, 1-vs-Set Machine, Machine Learning, Object Recognition, Face Verification, Support Vector Machines.

1 Introduction

Both recognition and classification are common terms in computer vision. What is the difference? In classification, one assumes there is a given set of classes between which we must discriminate. For recognition, we assume there are some classes we can recognize in a much larger space of things we do not recognize. A motivating question for our work here is: What is the general object recognition problem? This question, of course, is a central theme in vision. According to Duin and Pekalska [1], how one should approach multi-class recognition is still an open issue. Should it be performed as a series of binary classifications, or by detection, where a search is performed for each of the possible classes? What happens when some classes are ill-sampled, not sampled at all or undefined?

The general term recognition (and the specific terms object recognition and face verification that we consider in this article) suggests that the representation can handle different patterns often defined by discriminating features. It also suggests that the patterns to be recognized will be in general settings, visually mixed with many classes. For some problems, however, we do not need, and often cannot have, knowledge of the entire set of possible classes (Fig. 1). For instance, in a recognition application for biologists, a single species of fish might be of interest. However, the classifier must consider the set of all other possible objects in relevant settings as potential negatives. Similarly, verification problems for security-oriented face matching constrain the target of interest to a single claimed identity, while considering the set of all other possible people as potential impostors. In addressing general object recognition, there is a finite set of known objects in myriad unknown objects, combinations and configurations – labeling something new, novel or unknown should always be a valid outcome. This leads to what is sometimes called “open set” recognition, in comparison to

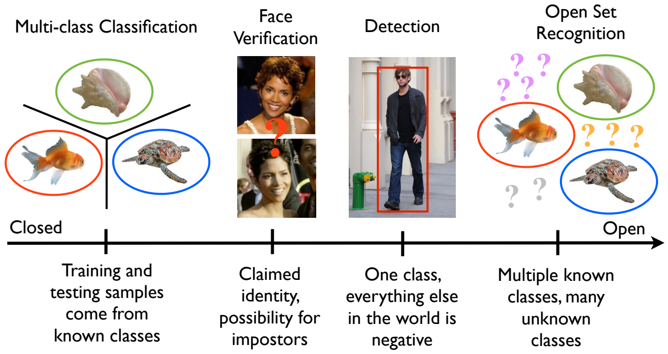

Multi-class Classification
Face Verification
Detection
Open Set Recognition
? ?
? ?
? ? ?
? ?
Closed
Open
Training and testing samples come from known classes
Claimed identity, possibility for impostors
One class, everything else in the world is negative
Multiple known classes, many unknown classes

Fig. 1. Vision problems arranged in order of “openness”. For some problems, we do not have knowledge of the entire set of possible classes during training, and must account for unknowns during testing. In this article, we develop a deeper understanding of those open cases.

- *Walter Scheirer is with Harvard University, Cambridge, MA, 02138.* *E-mail:* wscheirer@fas.harvard.edu
- *Archana Sapkota and Terrance Boult are with the University of Colorado, Colorado Springs, Colorado Springs, CO, 80918.* *E-mail:* lastname@vast.uccs.edu
- *Anderson Rocha is with the Institute of Computing, University of Campinas (Unicamp), Campinas, Brazil.* *Email:* anderson.rocha@ic.unicamp.br

systems that make closed world assumptions or use "closed set" evaluation.

For many vision problems, researchers have assumed one has examples from all classes, and have subsequently labeled the entire space in binary fashion as either positive $(+1)$ or negative (-1). In contrast, an open set scenario has classes, not just instances, in testing that were not seen in training. It is somewhat reasonable to assume we can gather examples of the positive class, but the number and variety of "negatives" is not well modeled. The important difference is, in the words of Zhou and Huang [2] (with a bit of inspiration from Tolstoy), "All positive examples are alike; each negative example is negative in its own way". Furthermore, even if all of the negative classes were known, from a pragmatic point of view, we generally cannot have sufficiently many positive examples to balance the required sampling of the negatives. In either case, we seek to generalize the problem from a closed world assumption to an open set.

Object detection is perhaps the most familiar vision problem that does not exist in a specific closed setting. The goal of detection is to locate an object of interest in an image. Since a negative detection is anything other than the class of interest, the problem is much more open than closed. Popular detection approaches train binary classifiers with a relatively modest sampling of positive examples and a very large sampling (often on the order of millions) of negatives from thousands of different classes. This is an appropriate strategy when a good sampling of the negative classes is possible, but with very incomplete knowledge of the possible negative classes it can lead to inaccuracies in many settings. In addition, we are generally left with a "negative set bias" [3] defined by the very large sampling of classes we do know about. In a sense, when we have very limited knowledge of the domain of possible classes, detection becomes a special case of open set recognition, with just one class of interest.

Fig. 1 depicts a few popular vision problems with varying qualitative degrees of openness. Intuitively, a problem with only a single class of interest is less open than one with many. However, the number of unknown classes we might encounter should also play a critical role. Let us formalize the "openness" of a particular problem or data universe by considering the number of target classes to be identified, the number of classes used in training, and the number of classes used in testing:

$$\text{openness} = 1 - \sqrt{\frac{2 \times |\text{training classes}|}{|\text{testing classes}| + |\text{target classes}|}} \qquad (1)$$

The above formulation yields percent openness (values between 0% and 100%), where 0% represents a completely closed problem, and larger values more open problems. For a fixed number of training classes, increasing the number of testing classes increases openness, as does increasing the number of target classes to identify. Increasing the fraction of classes available during training decreases openness. By taking the square root in Eq. 1, openness grows in a gradual manner as the number of classes increases (if linear, openness in this formulation would quickly move towards 1 with just moderate numbers of classes, which is not as meaningful). Table 1 shows values of openness for different examples considered in our work and others from the spectrum of problems in Fig. 1. The number of training instances per class is important to the accuracy of a given classifier, but is not a property of the problem itself, and thus not part of this definition. For almost any unconstrained real world problem, the number of testing classes can grow rapidly with openness approaching 100%.

TABLE 1
Examples of openness values for the vision problems of Fig. 1 as a function of the number of target classes to be identified, training classes and testing classes, calculated using Eq. 1. Multi-class classification is always 0% open.

Problem	**Targets**	**Training**	**Testing**	**Openness**
Typical Multi-class [1]	x	x	x	0%
Our work: Face Verif.	12	12	50	38%
Typical Detection [4]	1	100,000	1,000,000	55%
Our work: Obj. Recog.	88	12	88	63%
Our work: Obj. Recog.	88	6	88	74%
Our work: Obj. Recog.	212	6	212	83%

Potential solutions to the open set recognition problem must optimize for unknown classes, as well as the known classes. An important difference from typical multi-class classification is that a general open set multi-class solution must be able to label the input as one of the known classes or as *unknown*. It is not sufficient to just return the most likely class: the classifier must also support rejection. The first insight we offer here is that Support Vector Machines (SVMs) define half-spaces, and will classify data that is very far from any training sample. While we need solutions that support strong generalization, there should be a limit on how far from known data a sample associated with a given label can be.

Empirical risk, measured on training data, is what is classically defined and optimized over. However, for open set recognition it is critical to consider how to extend the model to capture the risk of the unknown from insufficient generalization or specialization. This is different from the binary classifier approach that tries to maximize the margin, which is the gap between the positive and negative decision boundary. While maximum-margins can be very effective for closed set problems, the approach generally results in overgeneralization for open set problems. For example, in Fig. 2, the space containing unknowns ("?") would likely be labeled "dog" as there is nothing to limit the positive label propagation if the decision boundary exists between birds & frogs and dogs. The SVM found a plane to separate positive and negative classes, but only by considering the known negatives. One might view the maximum-margin approach as assuming all unknown points are equally likely to be positive and negative based on what is nearest, even if that point is quite far away. For a sample coming from an unknown class, such as the raccoon, that is an incorrect assumption. We believe that good solutions to the open set recognition problem require minimizing the open space representing the learned recognition function f, outside the reasonable support of the training samples.

The primary goal of this work is to develop a thorough understanding of open set recognition in a supervised learning setting. We construct the first formalization of this problem, and provide an empirical case expanding existing 1-class and binary SVMs with linear kernels to address open set recognition. The resulting *1-vs-Set Machine* is a step towards a solution. Specifically, we revisit the ideas of the 1-class and binary SVM for open set recognition problems, and address the generalization/specialization issues through a novel learning technique. Instead of tackling the generalization/specialization problem as an error minimization of the training function of the SVM, we introduce a concept of open space risk and then minimize an error function combining empirical risk over training data with the risk model for the open space. The known class training data represents the "Set" of 1-vs-Set.

To improve the overall open set recognition error, our 1-vs-Set formulation balances the unknown classes by obtaining a core margin around the decision boundary A from the base SVM, specializing the resulting half-space by adding another plane Ω and then generalizing or specializing the two planes (shown in Fig. 2) to optimize empirical and open space risk. This process uses the open set training data and the risk model to define a new "open set margin". The second plane Ω allows the 1-v-Set machine to avoid the over generalization that would misclassify the raccoon in Fig. 2. The overall optimization can also adjust the original margin with respect to A to reduce open space risk, which can avoid negatives such as the owl.

We organize the rest of this article as follows. First, we formalize the open set recognition problem in Sec. 2. In Sec. 3, we take a look at the related work in open set recognition and machine learning across vision and pattern recognition. In Sec. 4, we formalize our theoretical model of margin generalization and specialization to develop the 1-vs-Set Machine. We apply this model, as well as common SVM models for comparison, to the problems of object recognition and face verification and present results in Sec. 5. We conclude and discuss some ideas for future work in Sec. 6.

2 Open Set Recognition Formalization

Assume images of objects from various classes are processed into d-dimensional representations, *i.e.* we measure feature vectors $x \in \mathbb{R}^d$. We assume we have countably many classes y labeled by $\mathbb{N}$, and that there exists a probability measure $P(x,y)$ over $(x,y) \subset \mathbb{R}^d \times \mathbb{N}$. For simplicity we will focus on open set recognition of a single class, and without loss of generality we assume the label of this class of interest is 1. Further, we assume a sample can be either positive or negative, but not both (no nested classes). Let $\mathcal{P} \subset \mathbb{R}^d$ represent the positive input space where $x \in \mathcal{P}$ if $P(x,1) = \sup_y P(x,y)$, *i.e.* inputs where the class of interest is the most likely class. Recognition here can be viewed as finding an efficient approximation of $\mathcal{P}$.

Let $f : \mathbb{R}^d \mapsto \mathbb{N}$ be a measurable recognition function for some class $\mathcal{P}$, mapping measurements x to labels y. Following Smola [5], our overall goal is to find a function f that minimizes our expected error. More precisely, consider a loss function $L(x,y,f(x))$ that defines the penalty for incorrectly labeling a vector x:

$$L(x,y,f(x)) \geq 0 \quad \text{and} \quad L(x,y,y) = 0 \tag{2}$$

The fundamental multi-class recognition problem would be to find a recognition function f that minimizes the ideal risk $R_{\mathcal{I}}$:

$$\underset{f}{\operatorname{argmin}} \left\{ R_{\mathcal{I}}(f) := \int_{\mathbb{R}^d \times \mathbb{N}} L(x,y,f(x))P(x,y) \right\} \tag{3}$$

Unfortunately, since we are not given the joint distribution $P(x,y)$, we cannot directly minimize Eq. 3, and the problem is unsolvable in the fundamental formulation. The traditional approach at this point is to change the problem to use only things we do know. As Smola notes in [5] (Sec. 1.2.1), *"The only way out is to approximate* $[P(x,y)]$ *by the empirical probability density function..."*. Hence minimizing the ideal risk is switched to minimizing the empirical risk. Unfortunately, even minimizing empirical risk is, in general, ill-posed [5], [6]. So prior work ([6], [5], [7], among others) exploits other knowledge, such as assuming that the label space is at least locally smooth and regularizing the empirical risk minimization to make it well-posed. For example, assuming that f is from a particular Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$ is a way of adding a smoothness constraint, and minimizing empirical risk over $f \in \mathcal{H}$ (with a regularization term) is then well-posed.

This begs the question if "the only way" to approximate the ideal risk formulation is empirical risk, or if there are other things that are known that could/should be added as we move from the ideal risk minimization of Eq. 3 to our formulation of open set recognition. We prefer to make minimal assumptions about f, but intuitively, there is risk in labeling the open space as "positive" for any known class. The insight for open set

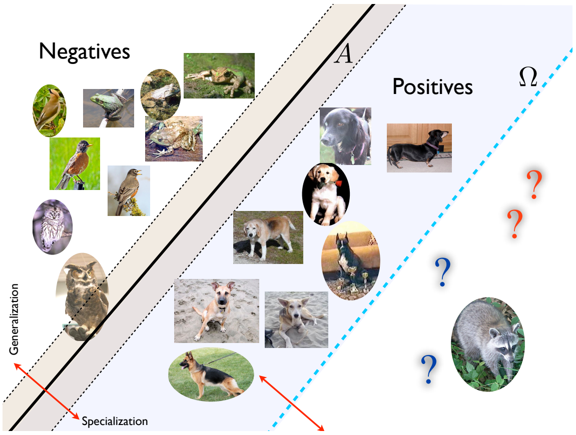


Fig. 2. The Open Set Recognition Problem explicitly assumes not all classes are known *a priori*. Square images are from training, oval images are from testing. The class of interest ("dog") is surrounded by other classes, which can be known ("frog", "birds"), or unknown ("owl","raccoon", "?"). Plane A maximizes the SVM margin making "dog" a half-space – which is mostly open space. The 1-vs-Set machine adds a second plane Ω and defines an optimization to adjust A and Ω to balance empirical and open space risk.

recognition is to recognize that we do know something else: we know where positive training samples exist *and* we know that in "open space" (the space far from known data) we do not have a good basis for assigning a label for the class of interest.

Before formalizing open space risk, we note that the maximum-margin concept can be viewed as using weak knowledge on open spaces, wherein we expect there to be errors near the decision boundary. Thus, these algorithms seek to maximize the distance between the known data and the decision boundary. This maximum-margin assumption does well for the space between the classes, but does not really address the remaining open space. There is, in general, still infinite amounts of space far from any known samples and often there is not even a point on the "other side" of such open space that could be used to define a margin. We seek to formalize and then manage such risk.

What information does open space provide? If an oracle provides the function $\psi(x) = 1$ for open space, where none of the known classes exist, a weak recognition system $\neg\psi(x) = 1$ can be built, even with no training samples. Combining training data with ψ, the estimation could be even better. Ideally, one might hope to define open space as the subspace $\mathbb{R}^d - \mathcal{P}$, but that just reduces the definition back to the problem of recognition. Estimating open space from positive data leads directly to a one-class formulation such as the 1-class SVM we examine in this article. Note, however, that the open space for a linear 1-class SVM is still a half-space. Our approach to open space estimation is similar in spirit, but the difference is that we reduce the labeled space to less than a half space and include other training data in the definition of open space, as well as in the subsequent recognition function.

While we do not know the joint distribution $P(x, y)$ in Eq. 3, one way to look at the open space risk is as a weak assumption: far from the known data the Principle of Indifference [8] suggests that if there is no known reason to assign a probability, alternatives should be given equal probability. In our case this means that at all points in open space, all labels (both known and unknown) are equally likely, and risk should be computed accordingly. However, we cannot have constant value probabilities over infinite spaces – the distribution must be integrable and integrate to 1. We must formalize open space differently, *e.g.* by ensuring the problem is well posed and then assuming the probability is proportional to relative Lebesgue measure [9]. Thus, we can consider the measure of the open space to the full space, and define our risk penalty proportional to such a ratio.

Consider an example with a large ball S_o containing both the positively labeled open space $\mathcal{O}$ and all of the positive training examples, as well as a given measurable recognition function f where $f(x) = 1$ for recognition of the class y of interest and $f(x) = 0$ when y is not recognized. *Open Space Risk* $R_\mathcal{O}(f)$ can be defined as

$$R_\mathcal{O}(f) = \frac{\int_\mathcal{O} f(x)dx}{\int_{S_o} f(x)dx} \tag{4}$$

where open space risk is considered to be the fraction (in terms of Lebesgue measure) of positively labeled open space compared to the overall measure of positively labeled space (which includes the space near the positive examples). The more we label open space as positive, the greater our open space risk. Eq. 4 is only one theoretical possibility. Other definitions can also capture the notion of open space risk, and some may do so in more a precise manner. This example does not include a loss function, class conditional densities, or class priors, but it is possible to define open space risk models that do. Such alternatives may allow for more precise estimations and/or simplify multi-class formulations, but since the unknown classes have unknown priors and unknown joint distributions, they would need to introduce more assumptions and complexity. A specific open space risk model for linear-kernels is introduced in Sec. 4.2.

While we want to minimize risk of the unknown in open space, we also need to balance it against the empirical risk $R_\mathcal{E}$ (the data error measure) over the training data. This empirical risk combines data errors via some type of performance metric (empirical probability of error and loss functions). Researchers have looked at SVMs and other learning models that optimize a more general data error measure [10]. While the presentation herein applies to many measures, and our implementation can optimize for multiple different empirical risk models, the one we consider most appropriate for the open set problem is the inverse of the F-measure. We look at this score in more detail in Sec. 5. Empirical risk can also include the specification of hard constraints, *e.g.*, meeting at least a particular false accept or false reject rate, which we discuss below.

In summary, our goal is to balance the risk of the unknown in open space with the empirical (known) risk. In this sense, we formally define the open set recognition problem as follows:

Definition 1. (The Open Set Recognition Problem) *Let samples $\hat{V} = \{v_1, \ldots, v_m\}$ from $\mathcal{P}$ be our positive training data and samples $\hat{K} = \{k_1, \ldots, k_n\}$ from other known classes $\mathcal{K}$ be our negative training data. Let $\mathcal{U}$ be the larger universe of allowed unknown (negative) classes which appear only in testing and let $\mathcal{T} = \{t_1, \ldots, t_z\}, t_i \in \mathcal{P} \cup \mathcal{K} \cup \mathcal{U}$ be our test data, where problem* openness *in Eq. 1 is* > 0.

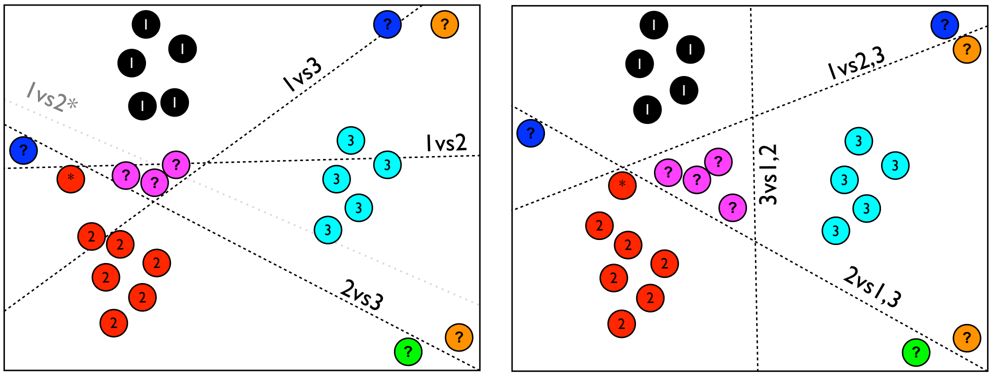
Given the training data $\hat{V} \cup \hat{K}$, an open space risk function $R_\mathcal{O}$, and an empirical risk function $R_\mathcal{E}$, open set recognition is to find a measurable recognition function $f \in \mathcal{H}$, where $f(x) > 0$ implies positive recognition, and f is defined by minimizing the **Open Set Risk***:*

$$\operatorname*{argmin}_{f \in \mathcal{H}} \left\{ R_\mathcal{O}(f) + \lambda_r R_\mathcal{E}(f(\hat{V} \cup \hat{K})) \right\} \tag{5}$$

where λ_r is a regularization constant.

In Eq. 5, we have defined open set recognition as minimizing the *open set risk*, which combines the *open space risk* and *empirical risk*, over the space of allowable recognition functions. Given what is assumed about the function $f \in \mathcal{H}$, this definition balances what is known via $\hat{V} \cup \hat{K}$, and the open space risk $R_\mathcal{O}$ associated with unknown classes $\mathcal{U}$.

We can also allow for explicit hard constraints on the training error (empirical risk). This is useful in some applications where one type of error may be constrained for operational



(a) 1-vs-1 (b) 1-vs-All

Fig. 3. The trouble with binary (1-vs-1) and multi-class (1-vs-All) classification for open set problems. In a 1-vs-1 scenario (a), good separation can be achieved between two classes during training, but this establishes margins that need not separate additional known or unknown classes. For instance, considering the margin between class 1 and class 2 above (labeled 1vs2), examples from class 3 and unknown classes fall indiscriminately across the margin. Similarly, in a 1-vs-All scenario (b), we see the same problem for unknown classes. In both cases, when considering just an additional training example (the red circle with a star in each figure), the results can be even worse, as the margins re-adjust for maximum separation. Far from the training classes this produces very significant margin plane movement, which can be seen in the light gray new margin separation plane 1vs2* in (a).

use, *e.g.* a maximally allowable false accept rate. Satisfying such constraints is not easily specified in the minimization formulation of Eq. 5. We can add this by making Eq. 5 subject to a constraint on the fraction of errors observed in the training set:

$$m\alpha \le \sum_{i=1}^{m}\phi(f(v_i)) \quad \text{and} \quad n\beta \ge \sum_{j=1}^{n}\phi(f(k_j)) \tag{6}$$

where $v_i \in \hat{V}$, m is the number of positive training samples, $k_j \in \hat{K}$, n is the number of negative training samples, $\alpha \ge 0$ and $\beta \ge 0$ allow a prescribed limit on true positive and/or false positive rates, and $\phi(z)$ is a given loss function, *e.g.*, the classic soft margins hinge loss $\phi(z) = max(0, 1-z)$ or squared hinge loss $\phi(z) = max(0, 1-z)^2$ functions. For a prescribed α and β it is possible there is no solution (100% classification of true positives with no false negatives is often not achievable). In a practical setting with operational constraints, we can fix either α or β and then choose our empirical risk term $R_{\mathcal{E}}$, requiring the system to effectively optimize the other parameter. This can allow us to set minimum precision or recall rates in an object recognition scenario, or bound the false accept rate in a face verification scenario.

In defining an open set problem, the evaluation methodology must sample some of the "unknown" classes in $\mathcal{U}$. Thus they are actually "known" but excluded from training. This is similar in spirit to general machine learning evaluation, where we must have "known data" that is considered unknown in training. One can do hold-out type cross validation, or simply have separate testing data. Similarly, open set recognition can hold out some classes for testing. Note that the formal definition does not precisely define the space of unknown classes – we do not assume they are enumerated, let alone modeled. It is, however, important to define an evaluation paradigm that does include the unknown classes. If we never test on "unknown" classes, the solution may seem overly constrained. Thus, testing on some set $\mathcal{T}$ where problem openness is > 0 is a requirement for open set recognition evaluation. Ideally, evaluation should consider test sets with multiple levels of openness and multiple sizes of training and testing data.

3 Related Work

Open set recognition has received only limited treatment in the literature, and almost all prior work focuses on evaluation. We are unaware of any prior formal definitions outside of evaluation protocols. In a study on evaluation methods for face recognition by Phillips et al. [11], a typical framework for open set identity recognition is described. The key to evaluation in open set recognition in the context described by Phillips et al. is the definition of an operating threshold τ, which all classification scores must meet or exceed to be considered matches. An open set recognition system incorporating a threshold does not naïvely accept a top score as a match, allowing it to handle the cases where a sample does not correspond to a known class. Of course, the choice of τ remains dependent on the requirements of the recognition system and its operating environment.

A series of thresholds can be considered to build a full performance curve (CMC, DET, PR, etc.), with parameters for matching instances selected by choosing one point on the curve. This idea is not just constrained to face recognition, and is familiar to many researchers working in recognition areas across vision. In [12], Fayin and Wechsler again view open set face recognition from just an evaluation perspective, describing it as a variation of the watch-list formulations in earlier face recognition testing at the National Institute of Standards and Technology (NIST). They state: "Open Set recognition operates under the assumption that not all the probes have mates in the gallery and it thus requires the reject option."

Given our formal definition of open set recognition from Sec. 2, we briefly discuss related work in recognition techniques that might satisfy that definition. A natural inclination towards solving the open set problem may be to consider binary and multi-class learning approaches with a representative sampling of negative training data to generalize the classifiers as much as possible. However, the nature of binary classification inhibits the controlled generalization needed for the open set problem. Consider the two examples in Fig. 3. 1-vs-1 classifiers [13] are trained by using positive examples from one class and negative examples from another. In a 1-vs-1 scenario, good separation can be achieved between the two classes during training, but this does not establish margins that separate additional known or unknown classes. 1-vs-All classifiers [13] are trained by using examples of a single class as the positive training set, and examples from all of the remaining (known) classes as the negative training set. In a 1-vs-All scenario, we can see the same problem that is present for 1-vs-1 for unknown classes. In both cases, when considering just an additional training example, the results can be even worse, as the margins re-adjust for maximum separation between the known data, while not taking other potential classes into account.

Another issue for any open set problem is that the training is both highly unbalanced and very incomplete (especially in the case of detection). Unbalanced data generally leads to overspecialization on the negative side. Resampling does not really solve the problem and the inherent imbalance in open set recognition presents issues that binary classifiers cannot easily overcome [14]. Thus, we turn to other methodologies that compensate for these deficiencies in our work.

In this article, we consider the open set recognition problem using the 1-class and binary SVM as a basis and introduce a new formulation to solve the problem with respect to generalization/specialization. While it is possible that a density estimator (such as [15], [16], [17], [18]) could be used instead of the SVM, we restrict our focus to linear kernel machines. SVM has a number of desirable traits for this work: its solutions are global and unique; it has a simple geometric interpretation; and it does not depend on the dimensionality of the input space. And it has been considered for open set recognition before.

3.1 SVM Approches to Open Set Recognition

The 1-class SVM introduced by Schölkopf et al. [19] adapts the familiar SVM methodology to the open set recognition problem. With the absence of a second class in the training data, the origin defined by the kernel function serves as the only member of a "second class". The goal then becomes to find the best margin with respect to the origin. The resulting function f after training takes the value $+1$ in a region capturing most of the training data points, and -1 elsewhere.

Let $p(x)$ be the probability density function estimated from the training data $\{x_1, x_2, \ldots, x_m \mid x_i \in X\}$, where X is a single class. A kernel function $\Psi : X \rightarrow H$ transforms the training data into a different space. To separate the training data from the origin, the algorithm solves the following quadratic programming problem for w and ρ to learn f:

$$min \frac{1}{2} \| w \|^2 + \frac{1}{\nu m} \sum_{i=1}^{l} \xi_i - \rho \tag{7}$$

subject to

$$(w \cdot \Psi(x_i)) \geq \rho - \xi_i \quad i = 1, 2, \ldots, m \quad \xi_i \geq 0 \tag{8}$$

where ρ is an offset that parameterizes the hyperplane in the feature space defined by the kernel Ψ, and ξ_i are slack variables. The kernel function Ψ impacts density estimation and smoothness. The regularization parameter $\nu \in (0, 1]$ controls the trade-off between training classification accuracy and the smoothness term $\| w \|$, and also impacts the choice and number of support vectors. In the 1-class SVM, $p(x)$ is cut by the margin plane minimizing Eq. 7 and satisfying Eq. 8. Regions of $p(x)$ above the margin plane define positive classification and capture most of the training data. As some researchers have pointed out in the literature [20], the 1-class SVM does not provide particularly good generalization or specialization ability, which has limited its use.

While not as much of an issue for binary SVMs, using Radial Basis Function (RBF) kernels, especially with a large γ, can also lead to over specialization. This can occur when "abusing" a 1-class SVM by performing grid search over the parameters and then testing with all available positive and negative examples for a given data set. While still formally a 1-class SVM, since only positive data is used for fitting, the optimization of class parameters to avoid negative training examples from the entire data set is not appropriate.

The 1-class SVM has received some (albeit limited) attention in the computer vision literature – mostly in the areas of image retrieval and face recognition. The application of 1-class SVMs to problems in computer vision was first made by Chen et al. [21] a decade ago. For binary classification, equal treatment is usually given to positive and negative training examples. However, Chen et al. argue that for image retrieval, while it is reasonable to assume that positive training examples cluster in a certain way, the same cannot be said about negative examples, since they can belong to *any* class. Thus, for an open set problem, it seems natural to consider a 1-class SVM, which is trained using only positive examples for a target class. The feasibility of this approach was shown by Chen et al. [21] (and in subsequent works [22] [23]), but with a caveat: kernel and parameter selection. Zhou and Huang note [20] that RBF and other Gaussian kernels are commonly used for 1-class SVMs, often leading to an "over-fitting" of the training data, with kernel parameters selected in an *ad hoc* manner, resulting in an overall lack of generalization to many classes. We believe the lack of generalization and specialization, combined with the common practice of closed set testing, are the primary reasons that the 1-class SVM did not gain much traction in vision.

Detection, as noted in the introduction, is an important open set problem, and several 1-class SVM techniques have been proposed to address it in that context. An interesting approach was presented by Hongliang et al. [24], where 1-class SVMs are used for face detection. By choosing to optimize the data used to train a 1-class SVM through subset selection and inclusion of *negative examples as positive*, they improved the generalization. This partially addresses the concerns of Zhou and Huang [20] at the training stage, but is logically inconsistent with no theoretical support. Cevikalp and Triggs [25] used a slab approach to define the boundaries around positive data, and then applied a 1-class SVM as a second stage filtering of false positives for object detection. Using a 1-class SVM trained with samples from the positive class and a few outlier cases, Wu and Ye [26] attempt to maximize the margin between the positive volume defined by a Gaussian kernel and the outliers for the task of novelty detection. This situation is similar to the approach proposed in this paper, with the following key differences:

- Our training data consists of a larger sampling of known data, instead of just a few outlier cases
- We consider a balanced risk formulation after SVM training
- We pursue a linear kernel approach that applies to both the 1-class and binary SVM

Beyond computer vision, 1-class SVMs have been considered in several other areas within pattern recognition, often implicitly to address open set recognition but without formal definitions of that problem. One of the earliest and best works is that of Manevitz and Yousef [27], which considers the problem of document classification. Using a 1-class SVM and a novel variation based on more strict outlier detection, the authors show high levels of classification accuracy on a standard document classification data set (Reuters). Manevitz and Yousef, like Zhou and Huang [20], point out that accuracy is quite sensitive to the choice of kernel and parameters, which they note is not well understood for this problem. In a similar vein, our own work [28] has used 1-class SVMs for an open set analysis of literary style.

The field of speech processing has also considered 1-class SVMs for problems with unknown classes. In Shen and Yang's work [29], a novel data description kernel based on the 1-class SVM is developed for text-dependent speaker verification. Kadri et al. [30] successfully apply 1-class SVMs to audio stream segmentation to overcome the problem of overlapping speech and very short speaker changes by maximizing the generalized likelihood ratio with respect to any probability distribution of the speech windows. Rossignol and Pietquin [31] use a 1-class SVM approach for audio segmentation in the context of overlapping speech. In a follow-up work to [30], Rabaoui et al. [32] move beyond stream segmentation to consider speech classification for recognition tasks.

While the 1-class SVM is specifically designed for the open set problem, the potential of the binary SVM for this problem should not be neglected. Specifically, when a classifier is trained with positive samples from one class, and negative samples from multiple classes (as is common in detection), it is a valid solution for open set recognition. Binary SVMs attempt to learn a margin that maximizes the separation between two classes. Let w be a normal vector to a hyperplane. To separate the data in the linear binary case (which we consider in this article), the algorithm solves the following optimization problem:

$$min \frac{1}{2}||w||^2 \tag{9}$$

subject to

$$y_i(w * x_i + b) \geq 1, \forall_i \tag{10}$$

where x_i is the i-th training example from the data $\{x_1, x_2, \ldots, x_m \mid x_i \in X\}$, where X contains positive and negative samples, and $y_i \in \{-1, +1\}$ is, for the i-th training example, the correct output label.

Revisiting binary SVMs for detection tasks, Malisiewicz et al. [33] note that a large ensemble of classifiers for a particular class trained with a single positive example and millions of negative examples yields surprisingly good generalization. In this article, we look at specific instances where more limited samplings of training data are assumed to be available, especially with respect to the known classes, where an approach like [33] cannot easily be applied.

Several binary SVM-like formulations should also be mentioned. Like our algorithm, a few approaches can be found that make use of multiple hyperplanes [34], [35], but not in the context of open set recognition. With a variant of the hinge loss function, Bartlett and Wegkamp [36] introduce a form of classification with a reject option. The reject option is a third decision for a binary classifier, expressing doubt when the conditional probability for a label of an observation is close to chance. To implement such a reject option, Bartlett and Wegkamp describe a construction (somewhat akin to our own fix for the problem of overgeneralization) that uses a threshold to mark an ambiguous decision space. However, the notion of rejection here is introduced to address the problem of uncertainty with respect to specific samples – not to reject samples that are not from the class of interest.

3.2 Other Approches to Open Set Recognition

Outside of the strict SVM framework, there are several other approaches that can apply to the open set problem, though they do not specifically address it. Recently, the vision community has produced some efforts to deal with the expressiveness and learnability of object models as well as the need for increasing amounts of training data [37]. Indeed, some work has been introduced to address the problem of object classification when training and test classes are disjoint (that is, no training examples of the target classes are available). In this direction, researchers have explored knowledge transfer for object class recognition such as: hierarchical structure of the object class space imposed by a general-to-specific ordering [38]; an intermediate layer of descriptive attributes to represent object classes [39]; and direct similarity computations between known object classes [40]. In the machine learning literature there is also some work in this direction such as zero-shot [41] and one-shot [42] learning techniques. To deal with a classification problem for which no training data is available for some classes, all of these approaches need to introduce a coupling between known and unknown classes. According to Lampert et al. [39], given that training data for the unobserved classes is not available, this coupling cannot be learned from samples and often needs to be inserted into the system by human effort.

It is useful to point out the differences between these types of approaches and the one we discuss in this article. In the open set recognition problem, we have training samples for the class of interest and samples for only some of the negative classes. However, in our solution to this problem, there is no need for any coupling between known and unknown classes, neither is any human effort required. Some of these above approaches have formal definitions, but without constraints on smoothness or data accuracy. It should be possible to develop a formalization that combines the related definition for open set recognition with categories, which would formalize the open set variations of these problems.

Finally, the open set recognition problem we consider in this article is also different from general unsupervised and semi-supervised learning techniques (described in [43], [44]). Common unsupervised techniques (such as clustering) do not address the formal definition of the open set problem, which is a more precise labeling than just an identification of groups with similar appearance within a large collection of images [45]. We want to make full use of the available training examples that we have. In addition, semi-supervised learning, which aims at the development of techniques to take advantage of both labeled and unlabeled samples [45], also does not apply to our problem since we are not propagating labels from the known samples to the unknown ones. In fact, our objective is to minimize the total recognition error (Eq. 5) for the class of interest as we discussed in Sec. 2. Any solution to the open set recognition problem could be applied as a tool in semi-supervised learning, but the criterion for evaluation might be significantly different.

4 Introducing the 1-vs-Set Machine

Our initial approach for the open set problem is based on a new variant of SVM that we call the 1-vs-Set Machine. As we described in Sec. 2, the risk minimization inherent in solving the open set problem involves a minimization of the positive labeled region to address open space risk (reflecting *overgeneralization*) combined with margin constraints to minimize empirical risk (reflecting *overspecialization*). In this article, we introduce a formulation with a linear kernel that applies to both 1-class and binary SVMs. Since the open set recognition problem is directly related to human cognition, arguments can be made in favor of linear kernels as an idealized discriminator with biological grounding [46], [47]. Moreover, in our experience, linear kernels produce better results than non-linear kernels for the same open set data (we show this in Sec. 5).

The initial definitions of the 1-class SVM were based on RBF kernels, but multiple works can be found [27], [48] that use a 1-class SVM with linear kernels. Once the equations for the 1-class SVM are defined, as in Sec. 3.1, the minimization problem is still well defined for a linear kernel. Intuitively, 1-class linear SVMs can be viewed as taking all positive data, finding the plane that just touches the support vectors and has the origin on the opposite side of the plane from the training data. For binary SVMs, the linear kernel is a typical choice, and is often used for the open set problem of detection [3], [33]. Here we describe the details of the 1-vs-Set algorithm.

4.1 Formalization of Risk for Linear Kernels

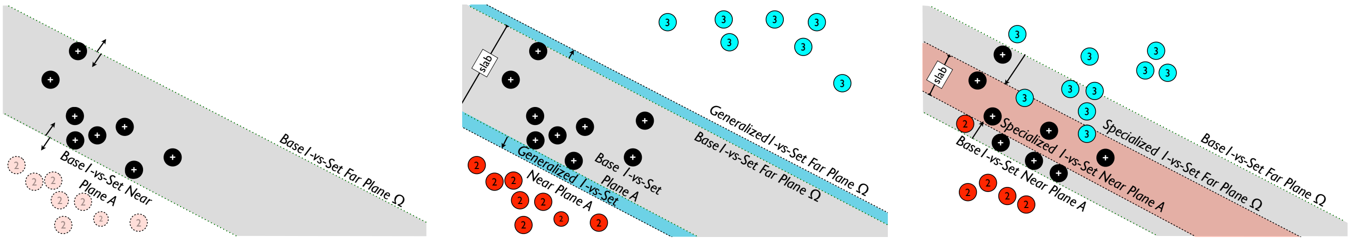
The first step in solving the optimization problem is to define a computationally tractable open space risk term. Our open set concept suggests that there is risk from labeling points far from the positive samples. As mentioned in Sec. 2, one way to look at this is in terms of ratios of Lebesgue measures. But computing $R_{\mathcal{O}}(f)$ for a given f may be intractable. We start with an example to highlight the general issues, but since our goal is just to minimize risk, we are able to find another form such that we minimize $R_{\mathcal{O}}(f)$ without ever explicitly computing it.

As a first approximation for open space risk, which we call shell-modeled risk, we take a large ball around the training samples, and an even larger ball around that one and consider anything between the two balls to be “open space.” More formally, let S_y be a ball of radius r_y containing the training data, and without loss of generality, let it be oriented such that all positive training samples for class y are in the upper half of the ball with h as the associated upper half-space such that the linear SVM defines $f(x) = 1$ when $x \in h$, and $f(x) = 0$ when not. Let S_o be a ball of radius r_o with the same center as S_y and let $r_o \gg r_y$. Shell-modeled open space $\mathcal{S}$ is thus the shell $S_o - S_y$, for an arbitrarily large r_o. Recalling $\mathcal{S} = S_o - S_y$, we can formalize the shell-modeled risk $R_{\mathcal{S}_h}$ for the half-space h intersected with the shell $\mathcal{S}$ as:

$$\begin{aligned} R_{\mathcal{S}_h}(f) &= \frac{\int_{\mathcal{S}} f(x)dx}{\int_{S_o} f(x)dx} = \frac{\int_{S_o \cap h} dx - \int_{S_y \cap h} dx}{\int_{S_o \cap h} dx} \\ &= 1 - \frac{\int_{S_y \cap h} dx}{\int_{S_o \cap h} dx} \geq 1 - \frac{r_y^d}{r_o^d} \approx 1 \qquad (11) \end{aligned}$$

We emphasize that for traditional linear kernels, labeling a half-space positive presents a significant risk of the unknown. We can consider other models to further lower the risk. The next simplest model, adding only one free parameter to the classic linear kernel, is to consider the piecewise constant $f(x)$ to be positive only in the space between two parallel hyperplanes. Consider a slab with fixed thickness δ, *i.e.* the space between two parallel hyperplanes separated by distance δ. Assume that the slab does not contain the center of balls S_o and S_y. It was shown by Lévy and Pellegrino [49] that the relative measure of such a slab compared to the measure of a d-dimensional ball goes to zero as the radius grows. Thus, the slab's d-dimensional shell-modeled open space risk is zero. Therefore, in what follows, we consider this specific slab model, but with additional refinements.

Since for all slabs with small δ, the shell-modeled risk will approach zero for large shells, a more refined model is desired to differentiate between slabs. We can consider the risk for a fixed but large shell size – in which case the thickness of the slab is directly proportional to risk. However, we will also want to include terms for open space closer to the training data. The


+
2
Base I-vs-Set Near Plane A
Base I-vs-Set Far Plane Ω
slab
3
Generalized I-vs-Set Far Plane Ω
Base I-vs-Set Far Plane Ω
Base I-vs-Set Plane A
Generalized I-vs-Set Near Plane A
Specialized I-vs-Set Far Plane Ω
Specialized I-vs-Set Near Plane A
Base I-vs-Set Near Plane A

(a) Base Linear 1-v-Set Machine (b) Generalization (c) Specialization

Fig. 4. Example of linear 1-vs-Set Machine showing the (a) base slab for both the 1-class and binary formulations, where the second class is only considered in the latter case (b) the generalization, and (c) the specialization operators. Blue refers to generalization, red for specialization and gray for the base linear 1-vs-Set Machine.

refined model will use marginal-style penalties where possible, and penalties related to ratios of Lebesgue measure within the large shell when not.

We define the class of functions $\mathcal{H}$ for the 1-vs-Set linear kernels in d-dimensions to be the slab between two parallel d-dimensional hyperplanes (A and Ω introduced in Sec. 1). We initialize the planes to just contain all positive training data. We can generalize beyond the initial training data by further separating the planes, or, we can specialize by moving either of the planes, bringing them closer together. For a given plane orientation, the open space risk is proportional to the separation distance between the planes. Thus our initial optimization starts by adjusting parameters based on plane separation. In particular, we define the overgeneralization risk as the expansion of plane distance: $\frac{\delta_\Omega - \delta_A}{\delta^+}$, where δ_A is the marginal distance of the near plane, δ_Ω is the marginal distance of the far plane, and δ^+ is the separation needed to account for all positive data. In a similar manner, we define risk for overspecialization as $\frac{\delta^+}{\delta_\Omega - \delta_A}$. During the optimization, these two terms are balanced with the empirical risk determined by classifying the available training samples with respect to the original margin. The decision to limit the growth based on the spacing of the intra-class data is our initial solution to balance overgeneralization risk from false positives if we added large balls, with the need to generalize to avoid future false negatives.

In the margin spaces ω_A around the near plane and ω_Ω around the far plane, we allow user specified control with parameters p_A and p_Ω to weight the importance of those nearby open spaces. We provide for these additional refinements (described in Sec. 4.2) because only the user can predict the openness of the problem and the importance of local open spaces. Combining the overgeneralization and overspecialization risk, along with any specified refinement, our open space risk R_ς for a linear kernel slab model is:

$$R_\varsigma = \frac{\delta_\Omega - \delta_A}{\delta^+} + \frac{\delta^+}{\delta_\Omega - \delta_A} + p_A\omega_A + p_\Omega\omega_\Omega \qquad (12)$$

4.2 Solving the Optimization Problem and Refining with Near and Far Plane Pressures

Given these definitions, we can numerically optimize the risk within the space of a slab. The optimization process for the 1-class and binary machines is detailed in Algs. 1 and 2.

Algorithm 1 Linear 1-vs-Set Machine Risk Optimization

```
Require: Parameter $\lambda_r$; Optional parameters $\alpha, \beta$
Require: Positive features $\hat{V} = \{v_1, v_2, \dots, v_m \mid v_i \in \mathcal{P}\}$
Require: Negative features $\hat{K} = \{k_1, k_2, \dots, k_n \mid k_i \in \mathcal{K}_j, 1 < j \leq c\}$, for other known classes $K_1, \dots, K_c$
procedure TRAIN($\lambda_r, \alpha, \beta, \hat{V}, \hat{K}$)
    if 1-class then Train a linear SVM $f$ using $\hat{V}$
    else Train a linear SVM $f$ using $\hat{V}$, $\hat{K}$
    end if
    for $\forall u_i \in \hat{V}, \hat{K}$ do
        Classify $\eta_i = f(u_i)$          ▷ Generate decision scores
    end for
    $\hat{s} = \mathrm{sort}(\hat{\eta})$          ▷ Sort decision scores
    $s_k = \min(\forall s_i \in f(\hat{V}))$
    $s_j = \max(\forall s_i \in f(\hat{V}))$
    $A$ = margin plane of $f$
    $\Omega$ = plane parallel to $A$ at $s_j$
    Greedy Optimization iteratively move $A$ to $s_{k+1}$ or $s_{k-1}$, $\Omega$ to $s_{j-1}$ or $s_{j+1}$ to minimize $R_\varsigma(f) + \lambda_r R_\mathcal{E}$, while satisfying any constraints provided by $\alpha, \beta$ in Eq. 6.
end procedure
```

Fig. 4 illustrates this process. The base linear 1-vs-Set machine, shown in Fig. 4(a), will just touch the extremes of the positive examples. We then turn to greedy optimization to move the planes simultaneously. If all negative training classes are outside that slab, the overspecialization risk terms will counteract the open space risk term and move the planes to generalize, as in Fig. 4(b). If the negative examples overlap the base slab, the overspecialization risk will be 1, and the overgeneralization risk term and probably the empirical risk term $R_\mathcal{E}$ will require the planes to move inward, as in Fig. 4(c).

Alg. 1 will result in an optimization where each plane is on a decision score from f. This is followed by a fine tuning to place each plane in between the point isolated during optimization and the next closest positive or negative point, with a special case when the plane is at an extreme of the data. We refine the plane positions, generalizing or specializing from the margin between the closest data and the plane based on parameterized "pressures" p_A and p_Ω that control how far to move the plane between the decision scores. If a decision score is the extreme, then we cannot really define a margin-based

Algorithm 2 Linear 1-vs-Set Machine Plane Refinement

```
Require: Linear SVM $f$ trained in Alg. 1
Require: Planes $A$ and $\Omega$ from Alg. 1
Require: Near and far plane pressures $p_A$ and $p_\Omega$
Require: Counts of positive and negative features $m$, $n$
Require: Sorted decision scores $\hat{s}$
procedure REFINE($f, A, \Omega, p_A, p_\Omega, m, n, \hat{s}$)
    Let $i$ be the index of decision score $s_i$ touching $A$
    if $i > 0$ then Shift $A$ to $s_i(\frac{1}{2} - p_A) + s_{i-1}(p_A - \frac{1}{2})$
    else Shift $A$ to $s_0 - p_A\delta^+$        ▷ No Margin, just generalize
    end if
    Let $j$ be the index of decision score $s_j$ touching $\Omega$
    if $j < (m+n)$ then Shift $\Omega$ to $s_j(p_\Omega - \frac{1}{2}) + s_{j+1}(\frac{1}{2} - p_\Omega)$
    else Shift $\Omega$ to $s_{m+n} + p_\Omega\delta^+$        ▷ No Margin, just generalize
    end if
end procedure
```

Algorithm 3 1-vs-Set Machine Prediction

```
Require: Test feature vector $t_x$
Require: Linear SVM $f$ trained in Alg. 1
Require: Planes $A$ and $\Omega$ from Alg. 2
function PREDICT($t_x, f, A, \Omega$)
    if ($A \leq f(t_x)$ and $f(t_x) \leq \Omega$) then Return +1
    else Return -1
    end if
end function
```

refinement. This is a relatively common case for Ω. When this occurs, we limit the generalization to be the user-specified pressure times the positive data width δ^+. The procedure for using pressures to refine positions is detailed in Alg. 2.

The parameterized pressures impact how much specialization versus generalization to apply. When considering the risk from a large slab we note that the near plane is likely to have any unknown negative data impinge on or near the positive boundary. For the far plane, it is more likely that added positive data will be slightly beyond the existing data, while negatives may not be so close. Thus we provide separate pressures so users may specialize on the near plane while generalizing on the far plane. In our experiments, we typically had better results after applying Algs. 1 and 2 when the near plane specialized with respect to the normal SVM margin, while the far plane generalized from the initial optimization result. We note, however, that this is partially just semantics as any position of the far plane is really a specialization with respect to a standard SVM, which could be viewed as a "far plane" at infinity. When addressing open set problems, the risk of the unknown is reduced by specializing the slab to be closer to the positive examples.

Finally, from the learned model f, and the refined planes A and Ω, any test vector t_x can be classified using Alg 3. In the software implementation, we sort the distances and search from the base position to optimize $R_\varsigma(f) + \lambda_r R_\mathcal{E}$. The code also supports setting fixed recall or precision, which is easy to implement given the explicit optimization process that satisfies both Def. 1 and Eq. 6. Since we are using an extension of the LIBSVM [50] library and sorting, our implementation is non-optimized, but the overall complexity of the linear 1-vs-Set machine can be made $O(n)$ for n data items by using the ideas of Joachims in [51], and simple selection to find the points close to the near and far planes.

5 Experimental Analysis

An important goal of our experiments is to highlight the radically different nature of data sets once they are recontextualized to reflect an open set problem. Torralba and Efros have recently noted that "Indeed, some datasets, that started out as data capture efforts aimed at representing the visual world, have become closed worlds unto themselves" [3]. They go on to analyze the various biases that exist in popular data sets, which are easily learned and leveraged to inflate recognition accuracies in a closed set scenario. By considering these same sets as open set problems in a cross-data context, we can directly address the problem of negative set bias (what the data set considers to be "the rest of the world" [3]). Here we propose testing scenarios that are more aligned with real world scenarios where we do not have knowledge of all classes.

For the object recognition experiments presented in Sec. 5.1, we make use of two different feature approaches. The first approach is the popular Histogram of Oriented Gradients (HOG) [4] descriptor, which is commonly used for detection problems. Applying the standard procedure described by Dalal and Triggs, we produced a 3,780-dimension feature vector for each image considered in our experiments below. In the second approach, the underlying features used for classification are generated by extracting points of interest (PoIs) from the images using Difference of Gaussians as proposed in [52], and then computing an LBP-like [53] feature descriptor in a window around each detected PoI. Feature vectors are composed of 59-dimension histogram bins that summarize the feature descriptor information for each image.

For the face verification experiments presented in Sec. 5.2, we also make use of two different feature approaches. The first approach is the same LBP-like descriptor used for the object recognition experiments, but applied exactly as described for faces in Sapkota et al. [53]. This results in 3,776-dimension histogram bins that are used as feature vectors for learning. The second approach is the common Gabor feature, which has been shown to produce very good results for face verification [54]. Applying the feature process described by Pinto et al. [54], we generate 86,400-dimension feature vectors.

Open set recognition presents a couple of new challenges with respect to evaluation. Specifically, we need to address the choice of which statistic to evaluate classification performance, as well as the organization of the data sets. This leads us to a few procedures that are not commonly used in object recognition and face verification. Our experiments below consider several aspects of classification, including the statistical significance of the 1-vs-Set machine's results, an assessment of the parameter space defined by p_A and p_Ω, and the impact

TABLE 2

Top half: detailed comparison between all of the different classifiers for the HOG features over the open universe of 88 classes. **Bottom half**: summary comparisons between the 1-vs-Set Machines and all other classifiers for the HOG features over the open universe of 212 classes, and the LBP-like features for both open universes. A ** or ++ means the results from testing H_0 (the F-measure of the Alg. on the row is lower than that of the Alg. on the column) were statistically significant at p = 0.01, * or + means p = 0.05. The * symbol indicates the 1-vs-Set Machine is significantly better in F-measure, while + indicates a baseline machine is significantly better. Dashes indicate no statistical significance, and a gray cell means no test was performed (a machine versus itself).

2-tailed paired t-test	binary 1-vs-Set	binary linear	binary RBF	1-class 1-vs-Set	1-class linear	1-class RBF
binary 1-vs-Set (HOG 88)		**	**	**	**	**
binary linear (HOG 88)	—		—	++	++	++
binary RBF (HOG 88)	—	++		++	++	++
1-class 1-vs-Set (HOG 88)	—	—	—		**	—
1-class linear (HOG 88)	—	—	—	—		—
1-class RBF (HOG 88)	—	—	—	—	++	
binary 1-vs-Set (HOG 212)		**	*	**	**	**
1-class 1-vs-Set (HOG 212)	—	—	—		—	*
binary 1-vs-Set (LBP-like 88)		**	**	**	**	**
1-class 1-vs-Set (LBP-like 88)	—	—	—		**	—
binary 1-vs-Set (LBP-like 212)		*	—	**	**	**
1-class 1-vs-Set (LBP-like 212)	—	—	—		**	—

of problem openness on classification performance. All 1-vs-Set machines in these experiments follow the most general optimization of Eq. 5 where $\lambda_r = 1$, and were not trained with explicit constraints (In Eq. 6, $\alpha = 0$ and $\beta = 1$). The near and far plane pressures are set at default values of $p_A = 1.6$ and $p_\Omega = 4$ for all experiments (except the one where we assess the impact of changing these parameters) to provide an extra measure of generalization.

Concerning statistics, accuracy is a natural choice for evaluating binary decision classifiers. Simply defined, accuracy refers to the correctly classified samples (true positives TP and true negatives TN) out of all of the classification decisions (TP, TN, false positives FP, and false negatives FN).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

Similarly, class averaged accuracy summarizes accuracies across all c classes for a given problem:

$$\text{Class Avg. Acc.} = \frac{1}{c}\sum_{i=1}^{c} \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i} \tag{14}$$

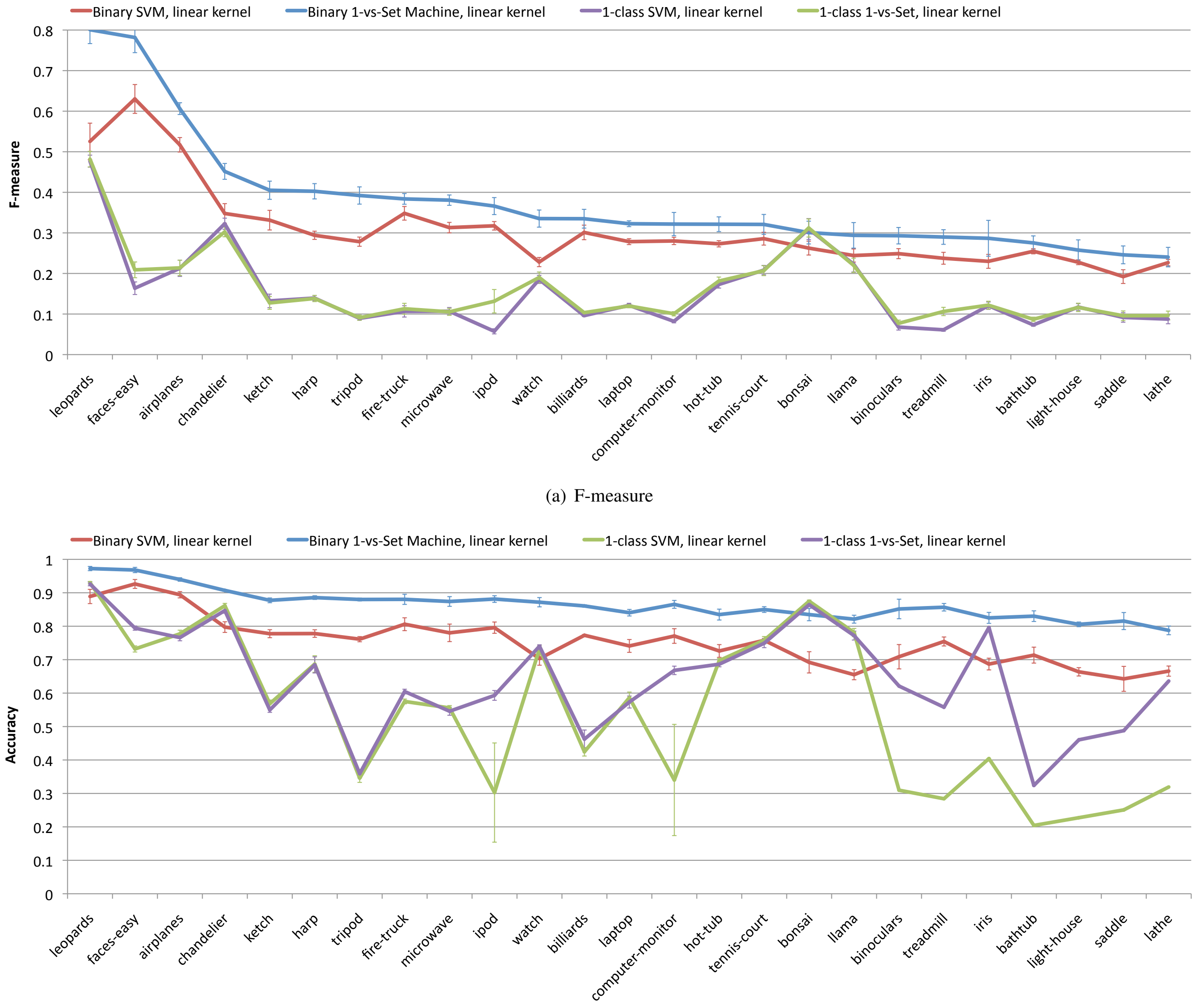
Class averaged accuracy cannot be used for open set recognition because the total number of classes c is always undefined. However, the typical accuracy measure of Eq. 13 can, but it tends to underemphasize the distinction between correct positive and negative classifications. Remember – we are primarily interested in identifying a small number of positive samples out of a much larger pool of negatives. To highlight this point, consider a case where a classifier returns one true positive out of 100 positive test samples, and zero false positives out of 100,000 negative test samples. This classifier is 99.9% accurate on this test – even though it is essentially a "no" classifier. For this reason, recall and precision are a common alternative.

Recall refers to the amount of correctly classified positive examples with respect to all the available positive examples: $\frac{TP}{TP+FN}$. Precision refers to the amount of correctly classified positive examples with respect to all of the false and true positives: $\frac{TP}{TP+FP}$. If we consider precision and recall for the task of comparing different classifiers, we encounter the problem of an "apples to oranges" comparison, where a collection of statistics not fixed to a specific precision or recall are present. For example, for the same training and testing data, the 1-vs-Set machine might produce a recall of 75% at a precision of 32%, while a binary SVM produces a recall of 62% at a precision of 25%. While *ad hoc* thresholding could be applied to the decision scores to produce a precision-recall curve, a better way to resolve this issue is to use F-measure, which provides us with a consistent point of comparison across inconsistent precision and recall numbers. In information retrieval and machine learning, F-measure is applied as a combination of precision and recall given by their harmonic mean:

$$\text{F-measure} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{15}$$

5.1 An Evaluation of Object Recognition

The data we consider for open set object recognition follows a cross-data set methodology adapted from [3]. For training, we choose all classes from the Caltech 256 set. For testing, we choose images for the positive class from Caltech 256, but for the negative classes, choose images from ImageNet [55]. Despite the bias within Caltech 256, we wanted to ensure some consistency between training and testing samples for the positive class, while attempting to generalize or specialize to the negatives from ImageNet based on a limited sampling of negatives from Caltech 256. While we have a sense of what our positives are during training, there is no way to know *a priori* if positive classes across data sets are consistent. However, negatives are definitely negative, thus we should be able to handle any examples from any data set based on our optimization. From this data, we construct two "open



(a) F-measure

(b) Accuracy

Fig. 5. A comparison between the 1-vs-Set Machine and typical SVMs with a linear kernel using two different statistics: F-measure (a) and Accuracy (b). These plots represent detail from the open universe of 88 classes with HOG features test found in Table 2. The classes shown here correspond to the top 25 for the binary 1-vs-Set Machine ranked by F-measure. Error bars reflect standard error. In every case shown, the binary 1-vs-Set Machine produces a higher F-measure and accuracy score compared to a binary SVM. The 1-class 1-vs-Set Machine shows more modest gains. In this very difficult open set scenario, accuracy places more emphasis on correct negative classification instances, where F-measure provides a more meaningful balance between correct positive and negative classification instances.

universes" of different sizes, allowing us to vary the training and testing data, which is somewhat constrained by the number of images provided by both data sets for the same classes. The first open universe consists of 88 classes selected at random, where we choose one class as positive, n classes as open set training data or binary negatives (where n varies depending on the experiment), and 87 classes as negatives for testing. The second, more open universe, consists of 212 classes selected at random, where we choose one class as positive, n classes as open set training or binary negatives, and 211 classes as negatives for testing.

We follow a multiple trial randomized testing procedure that selects different training, open set training, and testing sets for each experimental trial. This is done to verify consistency in the reported results across numerous trials, thus limiting any misleading impressions outliers might give for a single test. We cycle through all of the classes five times, treating each class positively once per iteration. The individual training and testing sample breakdowns vary as a function of experiment, and are noted below as we describe the individual tests. To ensure a fair comparison, the 1-class 1-vs-Set machine and all of the binary classifiers are trained with the exact same positive and negative examples. The 1-class machines use a default ν parameter of 0.5, the binary machines use a default C parameter of 1, and machines with an RBF kernel use a default γ parameter of one divided by the number of features (all LIBSVM default settings). These tests represent a total of 532,400 images for the open universe of 88 classes, and 13,610,400 images for the open universe of 212 classes, in different combinations across all of the randomized tests.

Our primary goal is to establish, in a rigorous statistical manner, the advantage of the 1-class and binary 1-vs-Set machines over typical SVM classifiers for open set recognition problems. We do this by applying a 2-tailed paired t-test [56] over all of the results for the classes from each of the open worlds and for each of our two feature sets to generate summary statistics. The t-test allows us to determine if two sets of classification results differ from each other in a significant way. The resulting p-values are assessed at the 0.05 confidence level (95% confidence). Our null hypothesis H_0 states the F-measures from the first set of classification results are lower than that of the second set. We reject H_0 when the p-value is less than 0.05. We also note cases where p is less than 0.01.

For the open universe test of 88 classes with both the HOG and LBP-like features, we train on 70 positive images and 14 negative images each from five other classes (approximately 5% of the available classes), and test on 30 positive images and 435 negative images across all the negative classes. For the more difficult open world of 212 classes, we train on 30 positive images and 5 negative images each from six other classes (approximately 3% of the available classes), and test on 30 positive images and 6,330 negative images across all the negative classes.

The results of these statistical tests are presented in Table 2. In a direct comparison with other machines utilizing a linear kernel, the results for the 1-class and binary 1-vs-Set machines are statistically significant: the null hypothesis is rejected in all but a single case. Even when we move to a cross-kernel comparison with typical machines utilizing RBF kernels, the binary 1-vs-Set machine produces better results that are statistically significant in all but a single case. Although this is an "apples to oranges" comparison, RBF kernels are an obvious alternative to any machine making use of a linear kernel on the exact same data, thus we include these results. We can conclude that for open set recognition problems, the 1-vs-Set machine is a suitable alternative to typical SVM classifiers.

In the next series of experiments, we break-out detail from the 88-class open universe set with HOG features to look at specific aspects affecting classification. First, the reader might be interested in the actual F-measures outside of the statistical summary presented above. Fig. 5(a) shows the F-measures for the top 25 classes ranked by binary 1-vs-Set machine performance. The binary SVM with a linear kernel is also plotted as a baseline comparison for the same classes. In every case shown, the binary 1-vs-Set machine is able to achieve a higher F-measure than the typical binary SVM by solving the constrained minimization problem over the exact same training data. The 1-class 1-vs-Set machine also shows a gain in F-measure, albeit at far more modest intervals for these particular classes. However, the 1-class 1-vs-Set machine should not be neglected: for 27 of the 88 classes, it produces better F-measures than the binary 1-vs-Set machine. Compared to published results on the typical closed set testing scenarios for the underlying data sets, these F-measures might seem low, but one must remember that this experiment is far more difficult: both machines saw only 5% of the total classes during training, and all of the classes during testing. By comparison, the accuracy numbers shown in Fig. 5(b) for the same classes are much higher. Accuracy places more emphasis on correct negative classification instances in large open set scenarios.

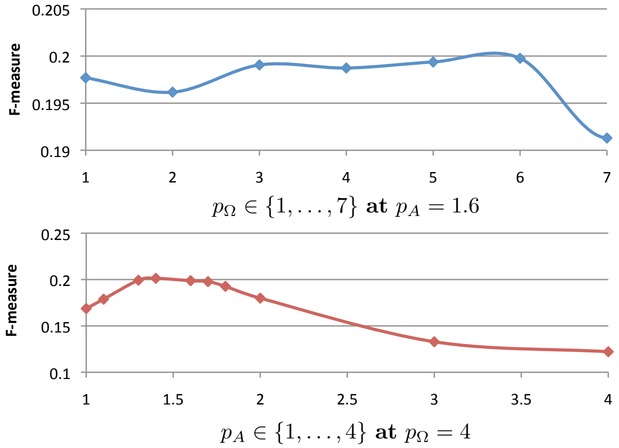


Fig. 6. Examples of the near and far plane pressure parameter space and corresponding F-measures when one of the two parameters is fixed at our selected default. F-measure in this plot is calculated over all of the classes in the open world of 88 classes with HOG features. Notice how movement on the near and far planes during Algs. 1 and 2 makes a difference in the resulting F-measures over the test data. Importantly, we see that the addition of a second plane Ω has an impact on recognition performance.

Next we turn to an assessment of the parameter space defined by the near pressure p_A and far pressure p_Ω described in Sec. 4.2. To analyze our results in a broader context, F-measure for this experiment considers all of the true positives, false negatives and false positives across all classes from a series of randomized tests, as opposed to just those from a single class as we did for the experiments above. Using the 88-class open universe set with HOG features, we searched the parameter space using binary 1-vs-Set machines to gain a better understanding of the impact of plane movement. Plotting portions of this data around our default parameters of $p_A = 1.6$ and $p_\Omega = 4$ (Fig. 6), we can see that movement on the near and far planes during training is indeed affecting the results achieved during testing. Of particular interest is the impact of moving the second plane Ω added by our algorithm (blue curve in Fig. 6), which limits false positives in what was a positive half-space. A bit too much generalization from the plane established by Alg. 1 (point 7 on the x-axis of the blue

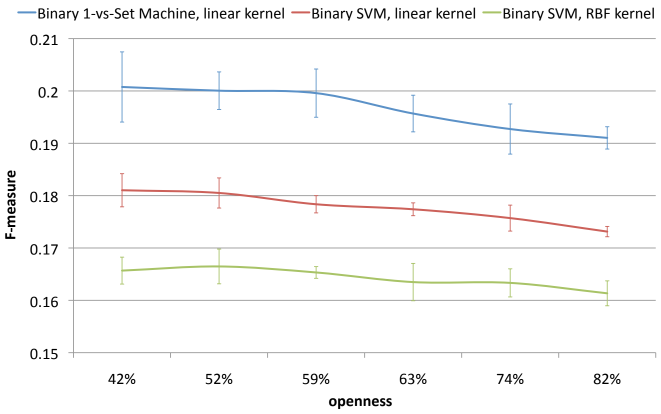


Fig. 7. An assessment of F-measure as a function of openness (growing from left to right) for a collection of binary classifiers. F-measure in this plot is calculated over all of the classes in the open universe of 88 classes with HOG features. As expected, all three machines decrease in accuracy as the universe grows to be more open. Even in the most open setting (82%), the 1-vs-Set Machine yields 8,129 fewer false positives compared to the binary SVM with a linear kernel, and 10,377 fewer false positives compared to the binary SVM with an RBF kernel. All 1-vs-Set Machine results are significantly better at a 95% confidence interval.

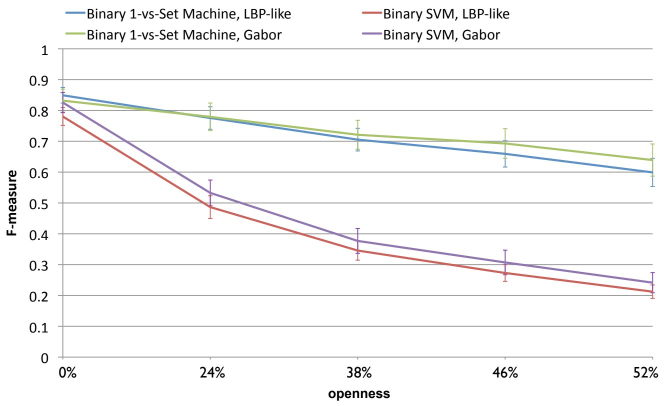


Fig. 8. Face verification results as a function of openness (growing from left to right) for a collection of binary classifiers and LBP-like and Gabor features. F-measure in this plot is calculated over all of the classes in the subset of LFW we consider. Notice that in closed set testing (0%), there is not much difference between the 1-vs-Set Machines and the typical binary classifiers. In all open set cases, the 1-vs-Set machine results are significantly better at a 95% confidence interval.

curve in Fig. 6) can cause a dip in F-measure.

Finally, we consider the impact of openness on F-measure. Intuitively, when more classes are available during training, we expect that the resulting classifiers should be more accurate. And this is exactly what we observe in practice. For the curves shown in Fig. 7, we chose 60 images as positive testing data for each class, and varied the openness of the test from 42% (30 negative classes seen during training) to 82% (3 negative classes seen during training). Testing remained constant at 30 positive images and 435 negative images. Again, F-measure for this experiment considers all of the true positives, false negatives and false positives across all classes from a series of tests. All three binary classifiers decrease in accuracy as the world grows to be more open. However, even when just a small number of classes are available during training, the 1-vs-Set machine is able to drastically reduce the number of false positives compared to the other machines.

5.2 An Evaluation of Face Verification

We also analyze the task of face verification, where people are the classes, with less obvious inter-class variations. We chose to evaluate classes from another well-known and challenging data set: Labeled Faces in the Wild (LFW) [43]. LFW (like many verification sets) is traditionally used for image-pair matching, which is really neither an open nor closed problem based on the *learning* criteria of this article. However, we can still define training and testing sets from it. The twelve people with at least 50 images (providing sufficient training data) were selected as gallery classes. For open set testing, we randomly selected 82 "impostors" from other people in the LFW set, yielding a total of 1,316 test images across all classes.

The impact of problem "openness" is also a very important factor for face verification, where a benchmark test might not reflect the performance of an algorithm over time as more people attempt to verify. Our experiments evaluated this scenario, starting with a completely closed world of twelve people, and adding more impostors in each individual experiment. Once again, we follow a five trial randomized testing procedure that selects different training (35 positive and negative samples per person) and testing sets (14 different test samples per person) for each experiment. Galleries are represented by binary 1-vs-Set machines and binary SVMs with linear kernels, which are trained with the exact same positive and negative examples. The closed set scenario for verification considers only the twelve known people for both the gallery and the probes (the test samples). The four subsequent experiments consider different numbers of *probe* classes to vary openness from 24% (30 probe classes) to 52% (94 probe classes). The gallery remains fixed at twelve people. This is slightly different from the experiments described for object recognition, where we varied the number of training classes, but more consistent with a critical analysis of typical face verification testing.

Fig. 8 shows a comparison between the binary 1-vs-Set machine and binary SVM with a linear kernel across the LBP-like and Gabor features. When the experiment is completely closed, the problem appears easy, with all machines producing high F-measures that are similar. However, as the problem grows to be more open, a large gap in F-measure appears between the 1-vs-Set machine and the binary SVM. Once experiments move beyond closed set testing protocols, it quickly becomes clear that typical machines with strong features are not always sufficient to address the open set aspect of the problem. This is particularly important for face verification, which is used in real world authentication applications. We also evaluated the statistical significance of the 1-vs-Set machine's results using the 2-tailed paired t-test. In all open set cases, the results are significantly better at a 95% confidence interval.

6 Discussion and Future Work

By revisiting the ideas of the general recognition problem and SVM-based recognition systems, we have gained a better understanding of the challenges of the problem and the shortcomings of the most frequently used solutions. In an open world, we cannot have knowledge of all classes, and it is impossible for us to sample and train on every possible image configuration for a class. Even if we could, with negatives greatly outnumbering positives, choosing representative negative training examples for a binary or multi-class classifier is problematic. The open set assumption changes how we must evaluate what is a "solution" – the risk of unknown classes must be accounted for without causing unforeseen errors.

With this in mind, we formalized the open set recognition problem as a risk-minimizing constrained functional optimization problem. As a first step towards a solution, we introduced a novel "1-vs-Set Machine" as an extension of the 1-class and binary Support Vector Machines to better support generalization and specialization in a manner that is consistent with the open set problem definition. The experiments for object recognition and face verification show that the 1-vs-Set machine is highly effective at improving accuracy when compared to 1-class and binary SVMs for the same problems. Interested readers can download our source code for the 1-vs-Set machine, as well the computed features for all experiments from: http://www.metarecognition.com/openset/.

Torralba and Efros [3] point out the effect of the closed-world assumption: a focus on beating the latest benchmark reports on the newest data set. Many researchers in the vision community have lost sight of the original purpose of these data sets: *recognizing the visual world.* By reformulating the recognition problem as open set recognition, we naturally avoid over-training biases. An open set testing methodology reduces the chance of data set bias because one cannot train on most of the data. While leave-one-out cross-validation is popular, the open set formulation suggests a leave-most-out validation. By restructuring tests over existing data sets we hope to encourage researchers to begin to address the more natural open set form of the recognition problem. For instance, our results on an open set reformulation of LFW may not be as impressive as the closed set testing, but they highlight the actual difficulty of unconstrained face verification.

The next step for this work is to extend our 1-vs-Set machine model to RBF kernels, which have a bounded volume that can also be adapted via generalization or specialization. This includes exploring alternative kernel density estimators outside of an SVM framework. Another future direction is to optimize other parameters beyond open space risk and empirical risk. The binary SVM bias term ρ and cost C are natural choices. We emphasize that the 1-vs-Set machine is only a first step towards an algorithm that is a truly suitable solution for open set recognition. Specific learning approaches that incorporate open set recognition into their fundamental design (especially at the initial density estimation stage) are of great interest.

Acknowledgments

The authors thank the financial support of ONR MURI N00014-08-1-0638, FAPESP 2010/05647-4, Army SBIR W15P7T-12-C-A210, and Microsoft. We also thank Jacques Wainer, who provided valuable feedback on early drafts of this work.

References

[1] R. P. Duin and E. Pekalska, "Open Issues in Pattern Recognition," in *Computer Recognition Systems*, M. Kurzynski, E. Puchala, M. Wozniak, and A. Zolnierek, Eds. Springer, 2005, pp. 27–42.

[2] X. Zhou and T. Huang, "Small Sample Learning during Multimedia Retrieval using BiasMap," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2001.

[3] A. Torralba and A. A. Efros, "Unbiased Look at Dataset Bias," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2011, pp. 1521–1528.

[4] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2005.

[5] A. Smola, "Learning with Kernels," Ph.D. dissertation, Technische Universität Berlin, Berlin, Germany, November 1998.

[6] V. Vapnik, *Estimation of Dependences Based on Empirical Data*. Springer-Verlag, 1982.

[7] B. Schölkopf and A. J. Smola, *Learning with Kernels*. The MIT Press, 2002.

[8] J. Keynes, *A Treatise on Probability*. Macmillan & Company, Limited, 1921.

[9] N. Shackel, "Bertrand's Paradox and the Principle of Indifference," *Philosophy of Science*, vol. 74, no. 2, pp. 150–175, 2007.

[10] T. Joachims, "A Support Vector Method for Multivariate Performance Measures," in *Intl. Conf. on Machine Learning*, 2005, pp. 377–384.

[11] P. Phillips, P. Grother, and R. Micheals, "Evaluation Methods on Face Recognition," in *Handbook of Face Recognition*, A. Jain and S. Li, Eds. Springer, 2005, pp. 329–348.

[12] L. Fayin and H. Wechsler, "Open Set Face Recognition Using Transduction," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1686–1697, Nov. 2005.

[13] A. Rocha and S. Goldenstein, "From Binary to Multi-class: Divide to Conquer," in *Intl. Conference on Computer Vision Theory and Applications*, Lisbon, Portugal, February 2009, pp. 323–330.

[14] B. Raskutti and A. Kowalczyk, "Extreme Re-balancing for SVMs: a Case Study," *ACM SIGKDD Explorations*, vol. 6, no. 1, pp. 60–69, 2004.

[15] C. Park, J. Z. Huang, and Y. Ding, "A Computable Plug-In Estimator of Minimum Volume Sets for Novelty Detection," *Operations Research*, vol. 58, pp. 1469–1480, September 2010.

[16] D. M. Tax and R. P. Duin, "Support Vector Domain Description," *Pattern Recognition Letters*, vol. 20, pp. 1191–1199, November 1999.

[17] X. Wang, P. Tino, M. A. Fardal, S. Raychaudhury, and A. Babul, "Fast Parzen Window Density Estimator," in *IEEE Intl. Joint Conference on Neural Networks*, 2009, pp. 3267–3274.

[18] E. Parzen, "On Estimation of a Probability Density Function and Mode," *The Annals of Mathematical Statistics*, vol. 33, pp. 1065–1076, September 1962.

[19] B. Schölkopf, J. Platt, J. Shawe-Taylor, A. Smola, and R. Williamson, "Estimating the Support of a High-dimensional Distribution," Microsoft Research, Tech. Rep. MSR-TR-99-87, 1999, available at: ftp://ftp.research.microsoft.com/pub/tr/tr-99-87.pdf.

[20] X. Zhou and T. Huang, "Relevance Feedback in Image Retrieval: A Comprehensive Review," *Multimedia Systems*, vol. 8, no. 6, pp. 536–544, 2003.

[21] Y. Chen, X. Zhou, and T. Huang, "One-class SVM For Learning in Image Retrieval," in *IEEE Conf. on Image Processing*, 2001, pp. 34–37.

[22] C. Zhang, X. Chen, M. Chen, and S. Ching, "A Multiple Instance Learning Approach for Content Based Image Retrieval Using One-Class Support Vector Machine," in *IEEE Intl. Conf. on Multimedia and Expo.*, 2005, pp. 1142–1145.

[23] K. Goh, E. Change, and B. Li, "Using One-Class and Two-Class SVMs for Multiclass Image Annotation," *IEEE Trans. on Knowledge and Data Engineering*, vol. 17, no. 10, pp. 1333–1346, Oct. 2010.

[24] J. Hongliang, L. Qingshan, and L. Hanqing, "Face Detection Using One-Class-Based Support Vectors," in *IEEE Automated Face and Gesture Recognition*, 2004, pp. 457–462.

[25] H. Cevikalp and B. Triggs, "Efficient Object Detection Using Cascades of Nearest Convex Model Classifiers," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2012.

[26] M. Wu and J. Ye, "A Small Sphere and Large Margin Approach for Novelty Detection Using Training Data with Outliers," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 11, pp. 2088–2092, Nov. 2009.

[27] M. Manevitz and M. Yousef, "One-class SVMs for Document Classification," *Journal of Machine Learning Research*, vol. 2, pp. 139–154, March 2002.

[28] C. Forstall, S. Jacobson, and W. Scheirer, "Evidence of Intertextuality: Investigating Paul the Deacon's *Angustae Vitae*," *Literary & Linguistic Computing (LLC)*, vol. 26, no. 3, pp. 285–296, Sept. 2011.

[29] Y. Shen and Y. Yang, "A Novel Data Description Kernel Based on One-Class SVM for Speaker Verification," in *Intl. Conf. on Acoustics, Speech and Signal Processing*, 2007, pp. 489–492.

[30] H. Kadri, M. Davy, A. Rabaoui, Z. Lachiri, and N. Ellouze, "Robust Audio Speaker Segmentation Using One Class SVMs," in *EURASIP European Signal Processing Conf*, 2008.

[31] S. Rossignol and O. Pietquin, "Single-Speaker/Multi-Speaker Co-Channel Speech Classification," in *Conf. of the International Speech Communication Association*, 2010.

[32] A. Rabaoui, H. Kadri, Z. Lachiri, and N. Ellouze, "One-Class SVMs Challenges in Audio Detection and Classification Applications," *EURASIP Journal on Advances in Signal Processing*, 2008.

[33] T. Malisiewicz, A. Gupta, and A. A. Efros, "Ensemble of Exemplar-SVMs for Object Detection and Beyond," in *Intl. Conf. on Computer Vision*, 2011.

[34] Jayadeva, R. Khemchandani, and S. Chandra, "Twin Support Vector Machines for Pattern Classifications," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 905–910, May. 2007.

[35] S. Chew, S. Lucey, P. Lucey, S. Sridharan, and J. Cohn, "Improved Facial Expression Recognition via Unit-Hyperplane Classification," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2012.

[36] P. Bartlett and M. Wegkamp, "Classification with a Reject Option using a Hinge Loss," *Journal of Machine Learning Research*, vol. 9, pp. 1823–1840, June 2008.

[37] M. Rohrbach, M. Stark, and B. Schiele, "Evaluating Knowledge Transfer and Zero-Shot Learning in a Large-Scale Setting," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2011, pp. 1641–1648.

[38] A. Zweig and D. Weinshall, "Exploiting Object Hierarchy: Combining Models from Different Category Levels," in *Intl. Conf. on Computer Vision*, 2007, pp. 1–8.

[39] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning To Detect Unseen Object Classes by Between-Class Attribute Transfer," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2009, pp. 951–958.

[40] E. Bart and S. Ullman, "Single-example Learning of Novel Classes Using Representation by Similarity," in *British Machine Vision Conference*, 2005, pp. 951–958.

[41] M. Palatucci, D. Pomerleau, G. Hinton, and T. M. Mitchell, "Zero-Shot Learning with Semantic Output Codes," in *Neural Information Processing Systems*, 2009, pp. 1–9.

[42] L. Wolf, T. Hassner, and Y. Taigman, "The One-shot Similarity Kernel," in *Intl. Conf. on Computer Vision*, 2009.

[43] M. G. Quiles, Z. Liang, F. Breve, and A. Rocha, "Label Propagation Through Neuronal Synchrony," in *IEEE Intl. Joint Conference on Neural Networks*, Barcelona, Spain, July 2010, pp. 2517–2524.

[44] G. Heidemann, "Unsupervised Image Categorization," *Image and Vision Computing*, vol. 23, no. 10, pp. 861–876, October 2004.

[45] C. M. Bishop, *Pattern Recognition and Machine Learning*, 1st ed. Springer, 2006.

[46] T. Serre and T. Poggio, "A Neuromorphic Approach to Computer Vision," *Communications of the ACM*, vol. 53, no. 10, pp. 54–61, 2010.

[47] J. DiCarlo and D. Cox, "Untangling Invariant Object Recognition," *Trends in Cognitive Science*, vol. 11, no. 8, pp. 333–341, Aug. 2007.

[48] T. Onoda, H. Murata, and S. Yamada, "Non-relevance Feedback Document Retrieval Based on One Class SVM and SVDD," in *IEEE Intl. Joint Conference on Neural Networks*, 2006, pp. 1212–1219.

[49] P. Lévy and F. Pellegrino, *Problemes Concrets D'analyse Fonctionnelle*. Gauthier-Villars Paris, 1951, vol. 8.

[50] C. Chang and C. J. Lin, "LIBSVM: a Library for Support Vector Machines," 2001, available at: http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

[51] T. Joachims, "Training Linear SVMs in Linear Time," in *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2006, pp. 217–226.

[52] D. Lowe, "Distinctive Image Features From Scale-Invariant Keypoints," *Intl. Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[53] A. Sapkota, B. Parks, W. Scheirer, and T. Boult, "FACE-GRAB: Face Recognition with General Region Assigned to Binary Operator," in *IEEE Intl. Workshop on Biometrics*, 2010.

[54] N. Pinto, J. J. DiCarlo, and D. D. Cox, "How Far Can You Get with a Modern Face Recognition Test Set Using Only Simple Features?" in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2009.

[55] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2009.

[56] NIST, *NIST/SEMATECH e-Handbook of Statistical Methods*, ser. 33. U.S. GPO, 2008.

Walter J. Scheirer received his M.S. degree in computer science from Lehigh University (2006), and his Ph.D. in engineering from the University of Colorado, Colorado Springs (2009). He is currently a postdoctoral fellow at Harvard University, and a research assistant professor at the University of Colorado, Colorado Springs. His primary research interests include computer vision, machine learning, biometrics, and digital humanities.

Anderson de Rezende Rocha received his B.Sc (Computer Science) degree from Federal Univ. of Lavras (UFLA), Brazil in 2003. He received his M.S. and Ph.D. (Computer Science) from the Univ. of Campinas (Unicamp), Brazil, in 2006 and 2009, respectively. Currently, he is an assistant professor in the Institute of Computing, Unicamp, Brazil. He is a Microsoft Research Faculty Fellow. His interests include digital image and video forensics, pattern analysis, machine learning, and general computer vision.

Archana Sapkota is a doctoral candidate in the Department of Computer Science at the University of Colorado, Colorado Springs. She works as a Research Assistant at the Vision and Security Technology Lab under the supervision of Dr. T. E. Boult. Her research interests include: face recognition, computer vision, pattern analysis, and biometrics. Prior to joining UCCS, she received a bachelors degree in computer science and engineering from the Indian Institute of Technology, Roorkee, India and worked in various industry roles for 3 years.

Terrance E. Boult , El Pomar Professor of Innovation and Security at the University of Colorado, Colorado Springs had published over 180 Papers and holds 9 patents (8 pending). Prior to joining UCCS, Dr. Boult held professorships at Lehigh and Columbia universities. He is also CEO/CTO of Securics Inc., a company in the biometrics and security space. Dr. Boult has served as an Assoc. Editor for TPAMI, has been the PAMI-TC chair and is a member of the IEEE Golden Core Society.