Detection and Identification of Rare Audiovisual Cues

Insperata accident magis saepe quam quae speres.
(Things you do not expect happen more often than things you do expect)   Plautus (ca 200(B.C.)

Project no: 027787

DIRAC

Detection and Identification of Rare Audio-visual Cues

Integrated Project
IST - Priority 2

DELIVERABLE NO: D4.11
Information transfer between categories & hierarchy learning

Date of deliverable: 31/12/08
Actual submission date: 20/01/09

Start date of project: 01.01.2006 Duration: 60 months

Organization name of lead contractor for this deliverable: **HUJI**

Project co-funded by the European Commission within the Sixth Framework Program (2002-2006)		
Dissemination Level		
PU	Public	X
PP	Restricted to other program participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Insperata accident magis saepe quam quae speres.
(Things you do not expect happen more often than things you do expect) Plautus (ca 200(B.C.)

D4.11 INFORMATION TRANSFER BETWEEN CATEGORIES & HIERARCHY LEARNING

The Hebrew University of Jerusalem (HUJI)

Abstract:

When an instance of a new object class appears for the first time, how are we to know what to do with it? The first problem is to identify the new example as a new class, which is the problem we focus on in this paper. We propose to identify novel sub-classes based on the discrepancy between two classifiers, a general level classifier which identifies the new example (e.g., it is a dog), and a specific classifier which rejects the new example (e.g., it is not any known dog breed). The classifiers are based on a hierarchical representation of the data, and correspond to adjacent levels of description in the hierarchy. An apparent advantage of this approach over traditional approaches to novelty detection, is that it relies on positive evidence from a general classifier, in conjunction with negative evidence from specific classifiers. In addition, the hierarchical structure enables us to construct a discriminative classifier, which often results in improved performance. We demonstrate the success of our approach on 3 datasets, two of them taken from standard benchmark databases of visual object classes.

1 Introduction

Over the past few years we have witnessed a growing interest in the task of visual object class recognition. Initially most of the work focused on the task of recognizing a single class, for which many methods have been devised. Recently, with the advance in single class recognition, some of the focus shifted to multi-class recognition. In particular, some papers investigated the question of learning from small sample [1, 2, 3], addressing both the computational aspects of shifting from single to multiple classes, and the possible sparseness of real world evidence for a given class. In this paper we address a somewhat different kind of sparseness of evidence. Rather than assume that each objective class is represented by a few samples (at least) during the training phase, we consider the case where some of the classes are entirely missing during the training phase.

Dealing with novel classes requires, before anything else, the ability to identify a test sample as belonging to an unknown class. This task can be regarded as a special case of novelty detection. Many methods have been proposed for novelty detection [4, 5]. Most of these methods detect novel events (also called outliers, unknowns or anomalies) by estimating the boundary of a single class data set. Some examples include the estimation of a spherically shaped boundary around a single class data set [6], the learning of a hyper-plane which separates the class data set from the rest of the feature space (one class SVM) [7], and the non parametric Parzen-window density estimation approach [8]. A few methods use a multi-class discriminative approach, as for example [9] for the detection of novel objects in videos, and [10] for the specific task of face verification. To our knowledge, all novelty detection approaches, which do not rely on samples of outliers or otherwise model the outliers distribution, detect novelty when normality is rejected (i.e., novelty is detected when all classifiers of known objects fail to accept a new sample).

In Section 2 we propose a novel approach for addressing this problem - how to identify a new sample from an unknown class as being such, or in other words, new sub-class identification. We consider a multi-class scenario, where several classes are already known; a sample from an unknown class is identified based on the discrepancy between two classifiers, where one accepts the sample and the second rejects it. The two classifiers are hierarchically related: the accepting classifier fits a general object class description, and the rejecting classifier fits a more specific object class description.[1]

To illustrate, suppose that the training data is composed of four objects of type 'Road Motorbikes', 'Sport Motorbikes', 'Mountain Bicycles' and 'Touring Bicycles'. Clearly one can also learn the notion of 'Two Wheels Vehicles'. Intuitively, when presented with a sample from a new sub-class such as 'Cross Motorbike', the sample's class is similar enough to the known classes in the sense that it is also a member of the general level category - 'Two Wheeled Vehicle'. On the other hand, when compared with the given hierarchy, it is sufficiently unique to be regarded as a class of its own. In other words, we assume that the "Visual Distance" between the existing sub-classes is similar to the distance between the new class and any of the known classes.

An important novel aspect of our approach is that we identify "interesting" novel events by utilizing accepting classifiers (at a more abstract level), and not based solely on the rejection by existing classifiers as in existing novelty detection methods. Thus, when detecting novelty based on a single class model as in [6, 7, 8], one has to deal with all possible objects. On the other hand, in our proposed method - once the general classifier has accepted the new sample, the problem of novelty detection is reduced to the much smaller subspace containing all specific object sub-classes, and their rejection thereof.

Because of this property, that novelty detection should be based on the rejection by a small number of classifiers (corresponding to all sub-classes of the general accepting classifier), we can adopt a discriminative approach to the problem. And because these classifiers correspond to highly similar objects which are hierarchically related, we hypothesize that a discriminative approach may yield better results than modeling each of the single specific sub-classes separately (possibly with a generative model), as is done in most work on novelty detection task.

We tested this hypothesis experimentally, using three data sets. One is a facial data set where the problem is reduced to face verification [13, 10], which is a special case of our general framework. In these experiments our discriminative method performs significantly better than any other method,

[1]The notion of hierarchal organization of object classes has been acknowledged and used in several recent visual object class recognition papers such as [11, 3, 12].

providing evidence that our hypothesis might be valid. We also show that the hierarchical relations between the sub-classed is crucial for the method to work effectively.

2 Novel Sub-Class Identification: Algorithms

In this section we describe our approach to the detection and identification of novel sub-classes, see Section 2.1. For comparison, we describe in Section 2.2 various alternative methods which we used in our experiments.

To obtain object models, we use the method described in [14]. This algorithm learns a generative relational part-based object model, modeling appearance, location and scale. Location and scale are described relative to some object location and scale, as captured by a star-like Bayesian network. The model's parameters are discriminatively optimized using an extended boosting process. This model has been shown to achieve competitive recognition results on standard benchmark datasets, approaching the state-of-the-art in object class recognition. Based on this model and some simplifying assumptions, the likelihood ratio test function is approximated (using the MAP interpretation of the model) by

$$F(\mathbf{x}) = \max_{C} \sum_{k=1}^{P} \max_{u \in Q(\mathbf{x})} \log p(u|C, \theta^k) - \nu \tag{1}$$

with P parts, threshold ν, C denoting the object's location and scale, and $Q(\mathbf{x})$ the set of extracted image features.

2.1 Our Algorithm: Hierarchical Approach

In order to identify novel sub-classes, we detect discrepancies between two levels of classifiers that are hierarchically related. The first level consists of a single 'general category' classifier, which is trained to recognize objects from any of the known sub-classes, see Section 2.1.1. The second level is based on a set of classifiers, each trained to make more specific distinctions and classify objects from a single known sub-classes. Using these specific classifiers, we build a single classifier which recognizes a new sample as belonging to one of the known sub-classes exactly (and not to any other sub-class). We have experimented with a number of such classifiers, as described in Section 2.1.2.

We look for a discrepancy between the inference made by the two final classifiers, from the two different levels, where the general classifier accepts and the specific classifier rejects a new sample. This indicates that the new sample belongs to the general category but not to any specific sub-class; in other words, it is a novel sub-class.

2.1.1 General Category Level Classifier

In order to learn the general category level classifier, we consider all the known sub-classes as being instances of a single (higher level, or more abstract) class. In accordance, all the examples from the known sub-classes are regarded as the positive set of training examples, and a set of images consisting of either clutter or different unrelated object examples is provided as the negative set. Classification is done based on the LRT value of the object model described above.[2] As we shall see in Section 4, this general classifier demonstrates a high acceptance rate when tested on the novel sub-classes.

2.1.2 Specific Category Level Classifiers

At the second more specific level of classifiers, the problem is reduced to the regular novelty detection task of deciding whether a new sample belongs to a known class vs. an unknown class. However, the situation is somewhat unique, with two special characteristics: there are multiple known classes, yet their number is bounded by the degree of the hierarchical tree (they must all be sub-classes of a single accepting object classifier). This suggests that a discriminative approach could be rather effective.

For each new sample, the algorithm proceeds as follows:

[2] A similar approach was discussed in [3] for the transfer of knowledge between closely related classes.

1. Assign a single sub-class. This is essentially a multi-class classification problem, which is solved discriminately as described below.
2. Obtain a normalized confidence measure for the specific assignment.
3. Classify the new sample, by comparing its confidence with a learned threshold.

To solve the multi-class classification problem in step 1 of the algorithms, we explored two alternatives: (i) Learn discriminatively [14] a unique representation for each sub-class, and choose the most likely one. (ii) Learn a shared representation for the general category level (at the parent level), and in this shared feature space apply any standard multi-class SVM approach. The two approaches are described below in more detail. Either way, for a new sample $\mathbf{x}$ which is classified as C_i, we obtain a confidence value $V_{C_i}(\mathbf{x})$.

In step 2 of the algorithm, the confidence $V_{C_i}(\mathbf{x})$ is normalized, relative to the values of correct classification and wrong classification for each sub-class as obtained during training. Specifically, let $V^c_{C_i}$ denote the average value of train examples classified correctly as C_i , and $V^w_{C_i}$ denote the average value of train examples from all other sub-classes classified wrongly as belonging to class C_i. The normalized score $S(\mathbf{x})$ of $\mathbf{x}$ is calculated as follows:

$$S(\mathbf{x}) = (V_{C_i}(\mathbf{x}) - V^w_{C_i})/(V^c_{C_i} - V^w_{C_i}) \tag{2}$$

If the classes can be well separated during training, that is $V^c_{C_i} >> V^w_{C_i}$ and both groups have low variance, the normalized score can be regarded as a confidence measure.

Let us now describe in more detail the two alternative multi-class classification schemes hinted at above:

(i) Unique Object Representation In this approach we learn an object model representation as described above, for each one of the sub-classes. Specifically, instances of the corresponding sub-class are used as positive examples, while all other instances (of the remaining sub-classes) are used as negative examples. Given a new sample, an LRT value is computed for each of the sub-class models, and the sub-class responsible for the highest LRT value is chosen for the assigned class, consequently giving its LRT value as $V_{C_i}(\mathbf{x})$.

(ii) Shared Object Representation As proposed in [11], we can use the representation of the general category as the basis for the distinction between the different sub-classes. Specifically, we used the representation built for the general category level classifier, as described above in Section 2.1.1. All sub-class instances can be represented in this space, and be used for training a multi-class classifier. We used two common approaches for multi-class classification with SVM, 1vs1 and 1vsAll.

- `1vs1`: Multi-class classification is based on pairwise classifiers , via majority voting. Once a sample is classified as one of the known classes, we compute $V_{C_i}(\mathbf{x})$ as the average margin over all classifiers contributing to the winning vote.
- `1vsAll`: Multi-class classification is based on individual classifiers, which are each trained to separate a single sub-class from all the other sub-classes; afterwards, the classifier with the highest signed margin is chosen, giving its value as $V_{C_i}(\mathbf{x})$.

2.2 Common Novelty Detection Approaches

For comparison, we implemented two single-class (standard) approaches to novelty detection, and a non-hierarchical discriminative approach.

2.2.1 Single Class Based Decision

We implemented two different single class approaches used in the standard novelty detection scenario, using both proposed representations: unique and shared.

- `Unique Representation`: A classifier is computed for each sub-class as described above, without using the instances of the remaining sub-classes as negative examples. The set of negative examples used for training is the same as the one used to train the general level classifier.

- `Shared Representation`: In the feature space defined by the shared representation, we train a one class SVM as our single class classifier (using [15]), following the method described in [7] for novelty detection. We extend this method to our multi-class scenario by considering the highest margin of all one class SVMs as $V_{C_i}(\mathbf{x})$.

2.2.2 No Hierarchical Relation

In order to explore the significance of hierarchy in our proposed scheme, we followed the training procedure as described in Section 2.1.2. We only changed one thing - instead of using a group of hierarchically related sub-classes, we collected a random group of sub-classes; every other step remained unchanged. For instance, instead of using 'Sport Motorbikes', 'Road Motorbikes', 'Mountain Bikes' and 'Touring Bikes' as the known group and 'Cross Motorbike' as the unknown group, we chose 'bat', 'soda can', 'sock' and 'Touring Bike' as the known group and 'Cross Motorbike' as the unknown. We only examined the unique representation, since the shared representation is by definition hierarchically based.

3 Datasets

We used three different hierarchies in our experiments. In the first hierarchy, the general parent category level is the 'Two-Wheeled Vehicle', see Fig. 1. 22 object classes, taken from [16], were added, in order to serve together with the original data set as the pool of object classes used for the random grouping described in Section 2.2.2. In the second hierarchy, the general parent category level is the 'Face' level, while the more specific offspring levels are faces of six different individuals, see Fig. 2. In the third hierarchy the general parent category level is 'Closed Frame Vehicle', see Fig. 3.

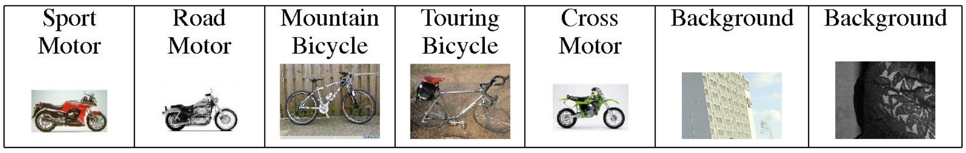

Figure 1: Examples from the object classes and background images, used to train and test the different Category level models of the 'Two-Wheeled Vehicle' hierarchy. The more specific offspring levels are: 'Sport-Motorbikes', 'Road-Motorbikes', 'Mountain-Bicycles' , 'Touring-Bicycles' and 'Cross-Motorbikes'. These images were taken from the Caltech-256 [16] dataset, pruned to reduce the overlap between sub-classes. Clutter images are used as background (or negative examples).

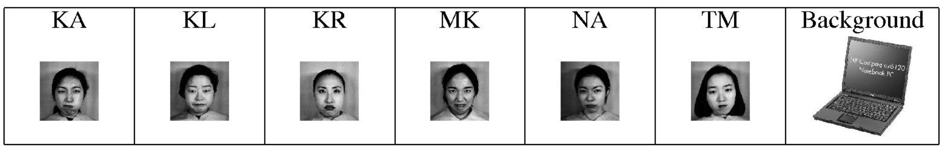

Figure 2: Examples from the object classes in the 'Faces' hierarchy, taken from [17]. General object images were used as negative examples.

4 Experimental Results

The following describes the results obtained by using the different methods proposed in Section 2.

Methods: All experiments were repeated 50 times with different random sampling of test and train examples. We used 39 images for the training of each specific level class in the 'Two Wheeled Vehicle' hierarchy, 15 images in the 'Faces' hierarchy and 20 images in the 'Closed Frame Vehicle' hierarchy. For each dataset with n sub-classes, n conditions are simulated, leaving each of the sub-classes out as the unknonwn class.

Bus	Truck	Sedan	Sport	Small Private	SUV	Background

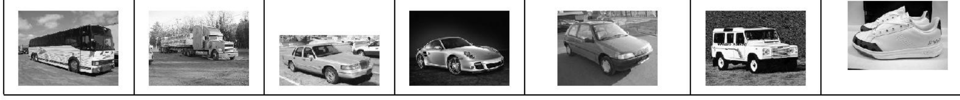

Figure 3: Examples from the object classes in the 'Closed Frame Vehicle' hierarchy, where the more specific offspring levels are: 'Bus', 'Small Private', 'SUV', 'Sedan', 'Sport Car' and 'Truck'. These images were chosen manually from Google and PicSearch, showing vehicles at similar canonical orientation. General object images were used as negative examples.

4.1 General Category Level Classifier

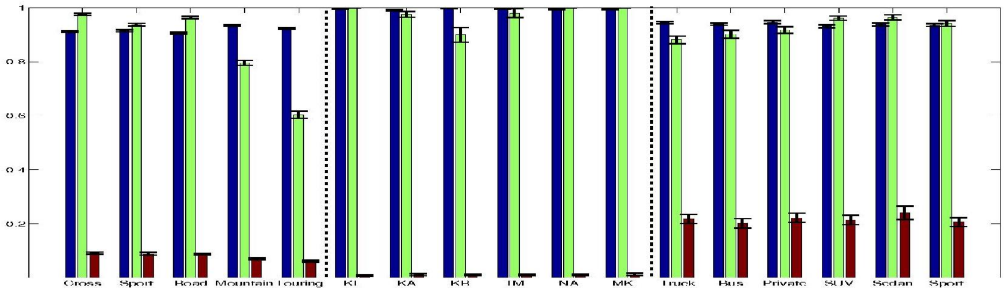

Figure 4: General category classifier, showing rate of classification as the general object. The three colored bars denote the following: blue (left) - known sub-classes used for the training of the general classifier; green (middle) - unknown sub-class not used for the training of the general classifier; and red (right) - clutter (two-wheels category) or other objects (faces and cars categories), classes used as background in the training of the general classifier. Results are shown for all three datasets (separate by a dashed line), showing from left to right: Two Wheels dataset, Faces dataset, and Closed Frame Vehicles dataset. For each dataset, each group of 3 bars corresponds to a different sub-class being left out as the unknown class.

Classification rates for the general level classifier are shown in Fig. 4. The bars indicate the number of samples classified as belonging to the general level classifier, divided by the total number of samples for either the known sub-classes, unknown sub-classes or background objects. We can see that for all classes but the 'Touring-Bicycle' and 'Mountain-Bicycle', the rate of acceptance is similar for known and unknown classes. In particular, note the marked difference as compared to the low (false) recognition rate of background instances, which are mostly rejected by the proposed general level classifier. Thus, in most cases, the general category level classifier appears to capture a more abstract object notion, which includes similar objects that have never seen before.

4.2 Specific Category Level Classifier

Results of classifying test samples by the different classifiers described in Sections 2.1.2,2.2 are shown in Figs. 567. The ROC curves are obtained by varying the threshold of the final classifier (step 3 of the algorithm described in Section 2.1.2), which controls how often a sample is declared unknown vs. known (where higher value corresponds to known). We note that the number of known and unknown examples in the test set was not balanced.

Our discriminative method, described in Section 2.1.2, always perform best for all 3 datasets. For clarity of representation, and since the unique representation always performed better, we omitted the results of the shared representation method in Figs. 567. We note that in the 'Faces' data set, the results obtained by the shared representation with 1vs1 classification scheme were similar to the unique representation scheme. The results of the shared representation with 1vsAll classification scheme always showed inferior performance.

For comparison, we observe that the performance of the single class control methods described in Section 2.2.1 is always very much inferior to the performance of our methods, often at the chance

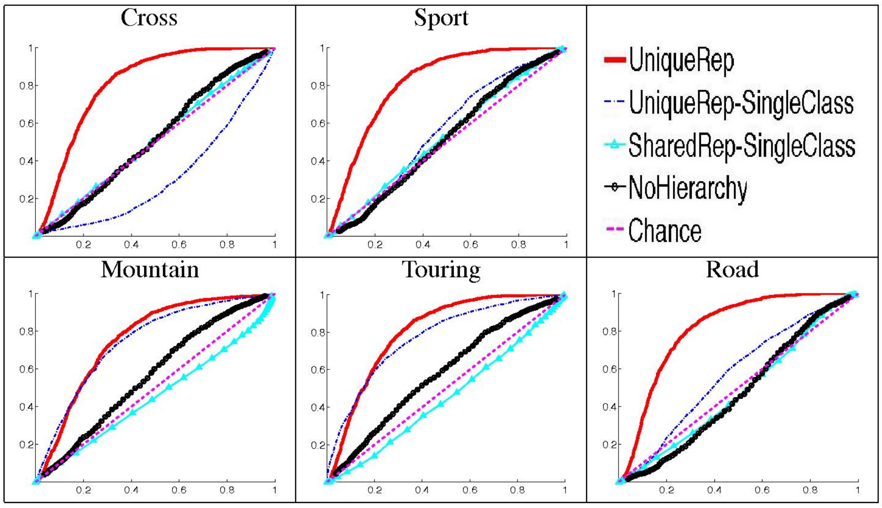

Figure 5: ’Two Wheeled Vehicle’ Specific level Classiers - ROCs. Results are shown separately for each sub- class left out as the unknown class (e.g., Cross denotes results when using the images of ’Sport Motor’,’Road Motor’,’Mountain Bike’ and ’Touring Bike’ as known sub-classes, while regarding the images of ’Cross Motor’ as the unknown sub-class). Recall results correspond to the Y-axis, and 1-Specity to the X-axis. UniqueRep refers to our proposed method of using a unique representation and a classier learnt discriminatively. UniqueRep-SingleClass refers to one of the control methods using the unique representation (see text), while SharedRep-SingleClass refers to the the second control method using the shared representation. Finally, NoHierarchy refers to the third control method using a random set of unrelated sub-classes, as described in Section 2.2.2.

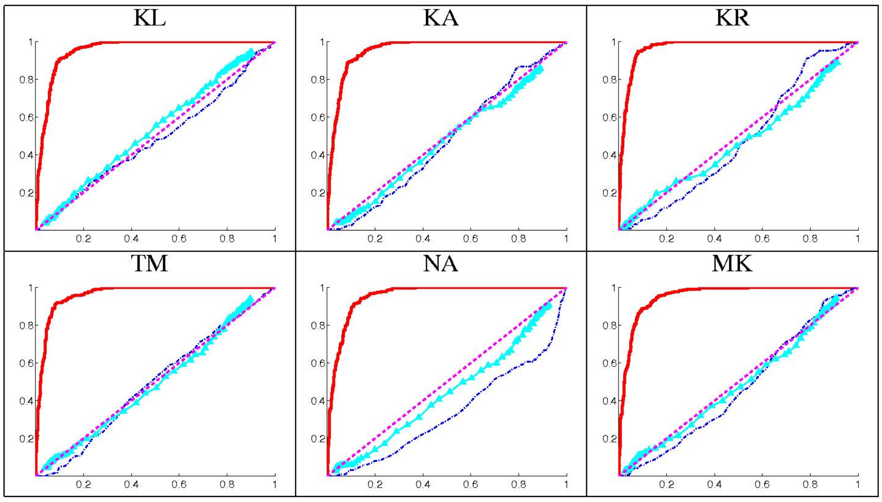

Figure 6: ’Faces’ Specific level Classifiers - ROCs (See caption of fig. 5).

level. Moreover, classification by the discriminative classifier which uses sub-classes that are unrelated hierarchically as described in Section 2.2.2, is also very poor as can be seen for the ’Two Wheeled Vehicle’ data set.

4.3 Discussion

The results above seem to confirm the viability of our approach. With 3 different datasets, and 3 different sub-class hierarchies, we see that the first premise of our approach seems to hold - it

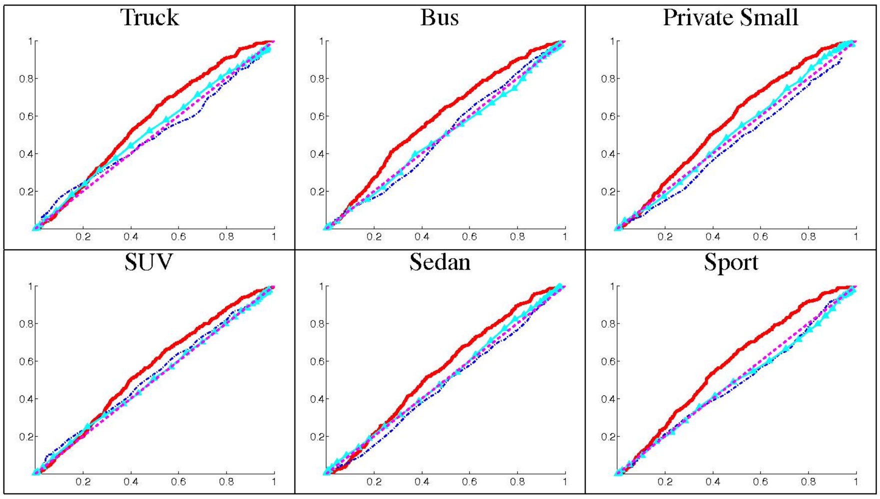

Figure 7: 'Closed Frame Vehicle' Specific level Classifiers - ROCs (See caption of fig. 5).

is possible to build a general level classifier, which will generalize to new unseen subclasses of related objects. Our discriminative novelty detection classifier, which is trained using the sub-class descendants of the general level object, seems to perform much better than more traditional novelty detection approaches. In two of the datasets it obtained excellent (for the Faces dataset) or very-good (for the Two Wheeled Vehicle dataset) classification results in identifying unknown classes, while still performing poorly on the third dataset. Clearly this last result should be improved by employing better single sub-class classifiers.

5 Summary

We have described a novel approach to new sub-class object detection. Its main distinguishing features are two: First, unlike more traditional novelty detection approaches, we use the hierarchical structure inherent in the data in order to build a hierarchy of representations and corresponding classifiers. The hope is that new sub-classes may be positively recognized by one of these classifiers, possibly at some high abstract level. Second, our novelty detection is based on a discriminative algorithm, which uses the hierarchical structure to constrain the set of possible objects. We tested our algorithm with three visual object class datasets, showing very promising results, and significant improvement over some more traditional approaches.

References

[1] E. Bart and S. Ullman. Cross-generalization: learning novel classes from a single example by feature replacement. *Proc. CVPR*, pages 672–679, 2005.

[2] F.F. Li, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE PAMI*, 28(4):594–611, 2006.

[3] A. Zweig and D. Weinshall. Exploiting Object Hierarchy: Combining Models from Different Category Levels. *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, 2007.

[4] M. Markou and S. Singh. Novelty detection: a review-part 1: statistical approaches. *Signal Processing*, 83(12):2481–2497, 2003.

[5] M. Markou and S. Singh. Novelty detection: a review-part 2: neural network based approaches. *Signal Processing*, 83(12):2499–2521, 2003.

[6] D.M.J. Tax and R.P.W. Duin. Support Vector Data Description. *Machine Learning*, 54(1):45–66, 2004.

[7] B. Scholkopf, R.C. Williamson, A.J. Smola, J. Shawe-Taylor, and J. Platt. Support vector method for novelty detection. *Advances in Neural Information Processing Systems*, 12:582–588, 2000.

[8] D.Y. Yeung and C. Chow. Parzen-window network intrusion detectors. *Proceedings of the 16th International Conference on Pattern Recognition*, 4:385–388, 2002.

[9] CP Diehl and II JB. Real-time object classification and novelty detection forcollaborative video surveillance. *Proc. of the International Joint Conference on Neural Networks*, 3, 2002.

[10] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. *Proc. of Computer Vision and Pattern Recognition Conference*, 2005.

[11] A. Bar-Hillel and D. Weinshall. Subordinate class recognition using relational object models. *NIPS*, 2006.

[12] Marcin Marszałek and Cordelia Schmid. Semantic hierarchies for visual object recognition. *CVPR*, 2007.

[13] J. Matas, M. Hamouz, K. Jonsson, J. Kittler, Y. Li, C. Kotropoulos, A. Tefas, I. Pitas, T. Tan, H. Yan, et al. Comparison of face verification results on the XM2VTS database. *Proc. 15th International Conference on Pattern Recognition*, 4:858–863, 2000.

[14] A. Bar-Hillel, T. Hertz, and D. Weinshall. Efficient learning of relational object class models. *ICCV*, 2005.

[15] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[16] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report UCB/CSD-04-1366, California Institute of Technology, 2007.

[17] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba. Coding facial expressions with gabor wavelets. *Proc. ICAFGR*, pages 200–205, 1998.