Available online at www.sciencedirect.com
SCIENCE DIRECT
Pattern Recognition 39 (2006) 1290–1299
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/patcog

Using natural class hierarchies in multi-class visual classification

Ilkka Autio*

Department of Computer Science, P.O. Box 68, 00014, University of Helsinki, Finland

Received 24 January 2006; accepted 24 January 2006

Abstract

We address the problem of computationally efficient visual classification of objects, and propose a system for solving multi-class problems in domains that have inherent hierarchic structure, such as subclass-superclass-relationships based on visual similarity. Class relationships are used at runtime to select the computationally simplest feature space that allows classification at high level of confidence for each example view. Classification accuracies can then be further improved using rank-order voting over multiple views. Our experimental results show that our system compares favorably to previously published results using a demanding benchmark. The results support the hypothesis that class hierarchies based on visual similarities are feasible and useful in controlling the accuracy vs. speed tradeoffs in classification.

Keywords: Object recognition; Object classification; Appearance-based object recognition; Sequence-based object recognition; Multi-object recognition; Hierarchic object recognition; Efficient object recognition

1. Introduction

Often the most difficult problem in a visual classification task is the choice of a suitable feature space. The selection of the feature space is largely application specific, although certain rules of thumb can be inferred from published works in visual classification [1–13].

The feature space selection problem becomes increasingly hard in multi-class problems where there is no single "trick", feature type, or kernel that is suitable for all classes. With problems of this type, it is advantageous to use diverse types of features [6]. Basic two-class problems may also benefit from diversity, if it results in robustness through increased resistance to noise.

Assuming that different types of features require extraction algorithms of varying computational complexity, it may be best to avoid classifiers that require the evaluation of the full set of features for every example. Some features may be *specialized* for discriminating between certain classes while being useless in the general case. If such features have costly extraction functions, unnecessary usage should be avoided when possible. The minimization of classification time through the selective use of features has aroused much interest recently [2,4,5].

We propose a system that solves visual multi-class problems by extracting features in a dynamic manner designed to minimize unnecessary extraction of complex and costly features. The computational complexity of classifying an example depends on both the classes the example belongs to, as well as the individual difficulty of the particular examples.

At the first stage, we construct a tree-like hierarchy that groups together visually similar classes as *superclasses*. First, the hierarchy makes explicit the mutually exclusive subclasses that are easily confused with each other, e.g., due to the subtlety of their differences in appearance. Second, the hierarchy may allow the grading of the seriousness of the classification errors. Erring on the broad category level may be considered more serious than erring on the subcategory level. For example, suppose we have two superclasses—one for visually similar animals and the other for spherical fruits. Using a crude classifier, the animals might be hard to tell apart although the superclasses would be easily distinguishable.

* Tel.: +358 9 19151168; fax: +358 9 19151120.
E-mail address: iautio@cs.helsinki.fi.

0031-3203/$30.00 © 2006 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2006.01.015

At the second stage, we train computational nodes that are connected according to the hierarchy. The nodes have the capacity to assign easy labels of an example directly, but they can also query more specialized nodes down the tree, if the current confidence is insufficient for direct assignment. The lower nodes discriminate between subclasses using more specialized and possibly costly features.

Finally, we use voting methods in classifying short video sequences, and demonstrate the advantages of using motion cues.

2. Related previous work

Hierarchic classification is a well established concept. The basic examples include decision trees [14,15], mixtures of experts [16], and hierarchical mixtures of experts [17]. Decision trees have the attractive property of using features selectively at runtime unlike the mixtures of experts-models. Hierarchical classification systems do not necessarily follow the *class hierarchy*, e.g., the subtrees of a decision tree do not necessarily correspond to semantically meaningful superclasses.

Some visual classification systems have successfully used a hierarchy of classes in driving an input example towards the most specific classifier applicable [18,19]. Some others [6,7] recognize the existence of perceptual similarity-based class hierarchies in multi-class problems, but are not focused on cost-efficient classification. Still others [12] use lower level hierarchies for two-class problems, e.g., training separate classifiers for different object poses in face detection. In Refs. [2,4,5,12] the proposed systems take advantage of early-stopping-mechanisms of various forms in two-class problems, such as face detection. For example, there are multi-stage-cascades or pipelines of simple classifiers which allow the quick rejection of non-face examples. Different types of features are not used, and the solutions do not have straightforward extensions to multi-class problems.

In Ref. [20], there is an interesting multi-class boosting procedure that reduces computational effort through feature-sharing between the classes. The procedure uses *fragments* similar to [1,3,21,22], which may have high computational complexity, are somewhat class-specific, and are suitable only if the objects are quite rigid. Less rigid parts-based representations exist, but we do not consider them universally useful, although such representations provide an ideal example of features having varying specificity and complexity. For example, examining the results in Refs. [3,20,21], one can observe gracefully increasing class-specificity as the size of the fragments grows.

Of the systems we are aware of, the system in Ref. [6] resembles ours the most. The key similarity is the use of heterogeneous, diverse types of features. The main differences are as follows: First, we do not assume an external segmentation oracle. Second, we pay attention to computational complexity and provide a parameter for tuning the speed versus accuracy tradeoff, whereas in Ref. [6] no such possibility is apparent, as the computational effort of a correct recognition is completely determined by the class of an example. Third, we take motion cues into account, and classify sequences of object views in addition to simple static views.

3. Hierarchic class relationships and similarity

We assume a multi-class setting having enough object classes to make the concept of a class hierarchy meaningful, e.g., more than three. Some of the mutually exclusive classes may be visually similar and hard to distinguish from each other. Although it seems unlikely that there exists a generic metric for similarity as perceived by humans [23], it seems that there are certain conventions or constraints that apply much of the time.

Perceived similarity is a blend of physically based visual similarity of the objects and other factors, such as associating the objects by semantics [23,24]. The semantics, in turn, may depend on the context and the problem at hand, making similarity modeling difficult.

3.1. The usefulness of class relationships in structuring the classification task

Designing a multi-class classifier for a wide variety of object classes may be difficult. There may be few constraints regarding the appearance, pose, and the allowed deformations of the objects. Thus, only very basic features requiring the least assumptions may be feasible, if the features must be meaningful for objects of all classes. For example, shape features requiring segmentation may be ruled out due to the apparent lack of general purpose segmentation algorithms.

The basic features may capture crude overall statistics or other characteristics related to the global appearance of the images. Visually similar classes, however may appear identical at the crude level, as their differences may be subtle. If, however, classification proceeds hierarchically, gradually ruling out most of the alternative classes, we may eventually have enough constraints to make sophisticated and specialized features available that can model the relevant subtle differences while discarding the rest.

Perceived similarity, including the possible semantic “taint”, may be used in grading the severity of the classification errors. Broadly speaking, a *visual* classification system tends to be judged according to how well it imitates the visual abilities of the end user. If a system fails at the very specific class level, but gets the broad category correct, the results seem more acceptable compared to the case of getting even the broad category wrong. If the system does not in any way see the broad categories perceived by the end user, it may be hard to produce acceptable results. In other words, perceived similarity may influence the loss-function the classifier should optimize.