Fast Visual Object Identification and Categorization

Michael Grabner **Helmut Grabner**
Horst Bischof
Institute for Computer Graphics and Vision
Graz University of Technology
{mgrabner, hgrabner, bischof}@icg.tu-graz.ac.at

Abstract

Recently recognition and categorization of objects from images using local features has become very popular. While similar approaches have been used for identification and categorization tasks they have not been treated in a common framework. In this paper we present a method that treats visual object identification, categorization in a common framework by exploiting ideas from image interclass transfer. We propose a hierarchically organized visual memory, where the high levels of the hierarchy represent generic classes and the leaves individual objects. The features used in the nodes of the hierarchy are learned using Adaboost on integral orientation histogram features (using these features makes a real time implementation possible). Learning the discrimination within a layer of the hierarchy is inspired by the work of Ferencz. Therefore, one can view our method as a hierarchical generalization of the interclass image transfer. First experiments demonstrate that the proposed method is able to learn meaningful object categories, as well as identification of individual objects.

1 Introduction

Recently we have witnessed an explosion of computer vision methods dealing on the one hand with visual object identification (e.g. [1, 2, 3]) and on the other hand with visual object categorization (e.g. [4, 5]). In both fields remarkable progress has been achieved. For object recognition local approaches such as [1, 2, 6], based on the detection of local key-points, and construction of a descriptor characterizing the photometric image content around the key-point, have gained much popularity. Even for categorization tasks, as shown by Fergus in [4], these approaches show promising results. Nevertheless, categorization and identification have been treated separately. However, for large scale object recognition systems (dealing with thousands of objects), it is essential to combine both tasks. Especially the identification task becomes easier and faster if the object category is already known due to the limited number of possible objects. On the other hand, also object categorization can benefit from object identification by using the trained images for creating generic models for categorization.

Therefore an important issue for an object recognition system is the ability to learn from

a few examples [5] or even from just a single example like in [3, 7, 8, 9]. Most recent approaches [10] suffered from requiring hundred's of training images for doing categorization. The major question is how knowledge of already trained objects can be used for learning of novel objects. This is also known as the interclass transfer problem. First approaches [11] in this direction find common features and use them for representation of multiple objects and additionally for the description of novel objects.

In this paper we claim that all these problems of identification, categorization and even the interclass transfer are highly related and should be treated in a common framework. We present a method that integrates identification, categorization and detection within one system. Our approach is based on hierarchical grouping of similar objects providing categories and specific object instances in the same framework. By using ideas from interclass transfer, our approach has the ability to incrementally learn novel objects requiring only a few training examples. The hierarchical structure is used for both categorization and identification while learning the system detects object classes on its own. Furthermore, because we use efficient data-structures for feature extraction the system has the ability of be used for real-time applications. Preliminary results are very encouraging and clearly show the power of the approach to categorize and identify objects.

In the remainder of the paper, we first introduce the approach for grouping similar objects and presenting the hierarchical approach for combining identification and categorization, see Section 2. In Section 3 we discuss two experiments, one illustrative and one showing first performance evaluations. Finally, we present some conclusion and work in progress.

2 Hierarchical structuring of objects

Recognizing many objects (> 100) in an adequate time requires a structured representation of the object models avoiding the one to one comparison of current systems. Therefore, we use a hierarchical structure of objects in order to form the so called *object-memory*.

Hierarchical approaches for recognition tasks with large data sets have been proposed in [12] based on the ideas of [13]. The goal of these approaches is primarily speed. However, different to their approaches our basic idea behind the structuring is to group similar objects into *object-layers*. Each object layer consists of a set of models representing a set of learned objects. Inspired by Ferencz [3] these models are classifiers able to make decisions of 'same' or 'different' class which can be used for object identification. For creating these models features are chosen by a feature selection technique to distinguish objects within a single layer.

For illustration, Figure 1 shows the basic idea of the approach using faces. In higher levels generic models are represented while lower levels models allow fine discriminations. As a consequence of using interclass transfer we are able to handle object categorization and identification within the object-memory structure.

In Section 2.1 we introduce the feature extraction technique for building an object model. Section 2.2 presents the algorithm for building the hierarchy.

2.1 Feature selection

The goal of our proposed feature selection is to distinguish objects by searching for discriminative features between the objects. We use SIFT like features [1] together with boosting [10] for feature selection. The basic idea of SIFT is to use gradient orientation histograms as a descriptor. In many applications these features have demonstrated impressive matching performance. There are also relations to the human visual system as shown by [14]. In addition it has been shown [15] that by using integral orientation histograms the descriptor

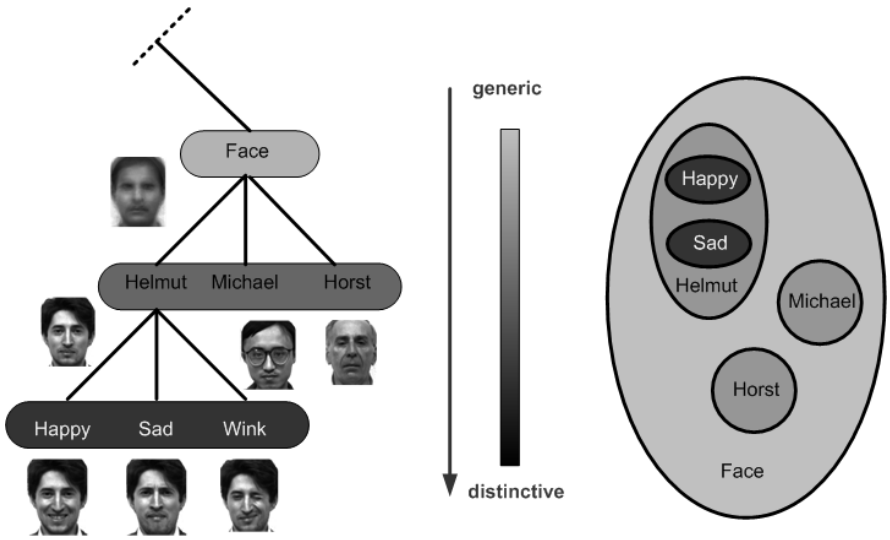


Figure 1: Object Memory (left) and the interclass relations (right). On the top the model describes a generic face while going into more deeper layers models become more discriminative.

can be computed very fast, which is one important issue in practice. To achieve robustness a large number of intermediate features [16] is used. Therefore, we first determine a feature pool of randomly selected rectangular regions (varying in position and scale) which represent possible features.

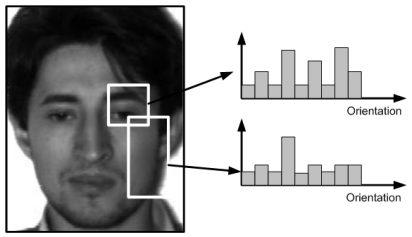


Figure 2: Integral orientation histograms [15] are used for a fast extraction of orientation histograms from rectangular patches.

Second, we select for each object the most discriminate features by boosting [17]. For doing this we have to build a weak classifier (hypothesis) for each feature.

Since usually we are dealing with very few examples we use the idea of learning a distance function $d : I \times I \rightarrow [0, 1]$ from [18], where I corresponds to a orientation histogram of a feature patch. This is evaluated for all possible combinations of samples of class 'same' vs. 'same' and class 'same' vs. 'different', respectively. As proposed by Ferencz [3] we fit two Gamma-distributions to the computed distances for the two classes, derived from the distance function. Thus we are able to obtain a hypothesis, simply by choosing the more probable class. This principle is visualized in Figure 3.

To summarize, a single hypothesis, extracted from a single feature, corresponds to a weak classifier in the boosting method. Performing N boosting iterations we select N features with corresponding weights. Thus we obtain for the local patches features and their

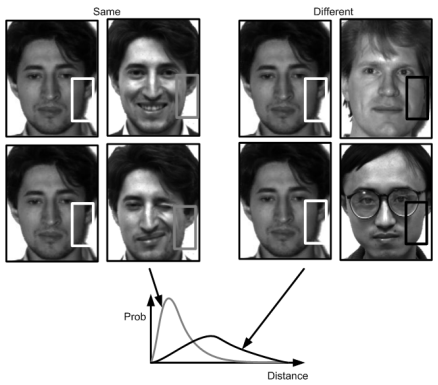


Figure 3: Gamma-distributions for the classes 'same' and 'different' are derived from a few training samples for a single feature similar to Ferencz [3].

weights. The final decision (strong classifier), which represents the output of an object model, is given by evaluating the term

$$hStrong(\mathbf{x}) = \mathrm{sign}\Big(\sum_{n=1}^{N} \alpha_n \cdot hWeak(\mathbf{x})\Big). \tag{1}$$

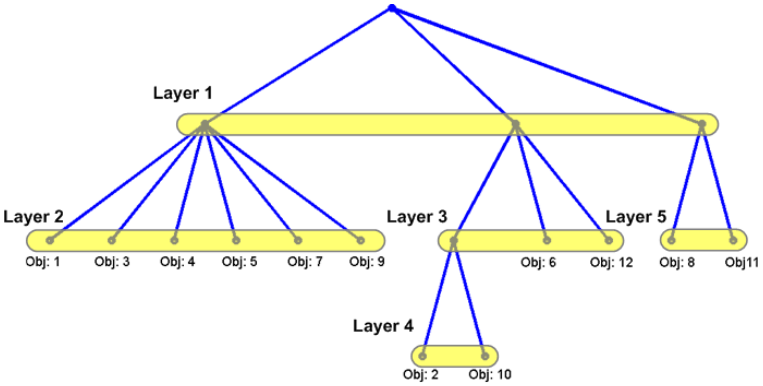
This can be expressed by a linear combination over the chosen features. A confidence can be obtained by just taking the sum in Eq. (1) (without the $\mathrm{sign}(\cdot)$ function).

2.2 Building the hierarchical structure

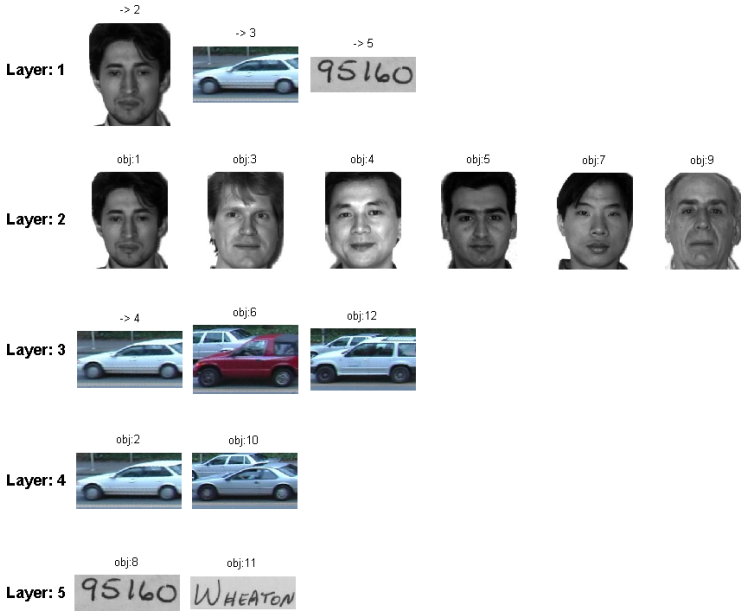
For organizing our object-memory in a hierarchical manner we need at least two objects to start with (an object consists of a set of representative images). In a first step, we put these two objects into a single layer and use the above described feature selection method to learn models for the two objects (i.e. discriminative features). When adding a novel object to the memory we first search for the most suitable layer. This is done by simply evaluating the novel training images, starting with the first layer. If each model in the current layer classifies the object as 'different', then the object is added to the current layer as a new object. A model is trained to distinguish the novel object from the objects in the layer. Furthermore, we retrain the parent model to achieve a more generic representation in the upper object layer. Otherwise, if we find a model in the current layer which can describe the novel object, we proceed recursively at the next layer. If there is no next layer, we create a new layer, which is initialized by the novel object and its parent, by training features distinguishing between these two.

3 Experiments

This section presents the results of two experiments. The first one is illustrative, the second one illustrates the recognition performance. For both experiments public available databases have been used.



(a) Hierarchical structure of the object-memory



(b) Layer content of the object-memory

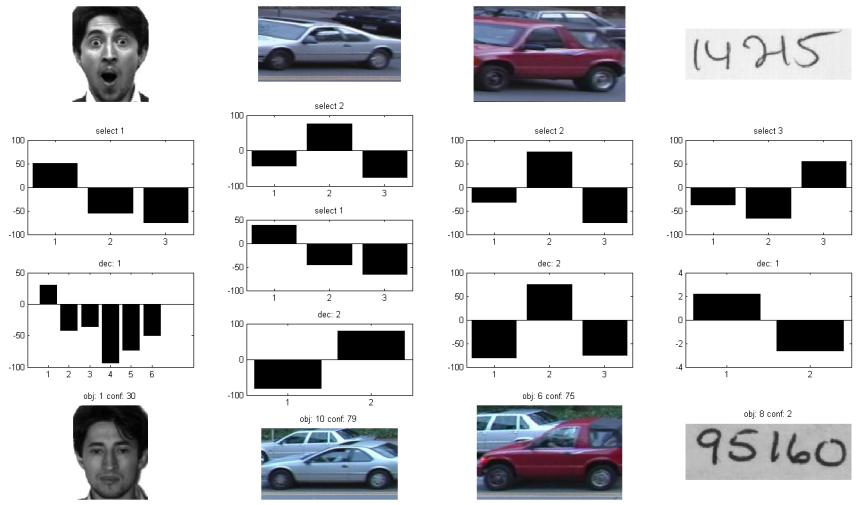
Figure 4: The object-memory is able of grouping similar objects into the same layer. All three object categories have been clustered. (a) depicts the hierarchical ordering of the object models and (b) the content of each object layer represented by one corresponding training image (i.e. nodes higher in the hierarchy hold more generic representations).

3.1 Experiment 1: Faces, cars and handwritten text

For the first experiment we selected objects from the Yale-Face database[1], the car database which has been provided by Ferencz[2] and third images from a handwritings-database[3] containing zip-codes and other handwritten letters. The purpose of this experiment is to show how the hierarchical structure automatically groups similar objects together. Therefore six faces, four cars and 2 handwritings have been chosen.

Figure 4 shows the learned object-memory. This nicely illustrates how the system is able of grouping similar objects into the same layer. In Figure 4(a) the hierarchical structure of the object models is shown while Figure 4(b) depicts the object layers with the object models (i.e. object models are represented by a representative training image). Furthermore, considering the category cars, we see that the hierarchical approach automatically determines the granularity of the grouping.

For evaluation we used 84 different test images. Evaluation results for some test images are shown in Figure 5. We achieved 94% detection rate (recall) and 86% precision on all 84 test images.



(a) (b) (c) (d)

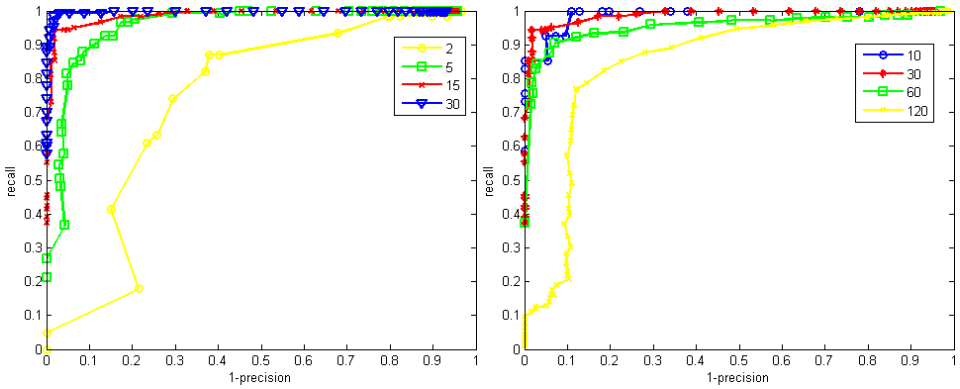
Figure 5: Illustrates the evaluation of four different test images (top row). Below each of them the evaluation path is represented by bar plots. They show the responses of each classifier at each layer in the evaluation path, see Figure 4. In the last row we finally see the identified object from the object-memory which is represented by a corresponding training image. In the last column we can see that the ZIP code has not been identified correctly (because this specific code has not been trained), nevertheless the chosen category is correct.

[1] `http://cvc.yale.edu/projects/yalefaces/yalefaces.html` (2005 Oct, 20)
[2] `http://www.cs.berkeley.edu/ ferencz/vid/dataset.html` (2005 Oct, 20)
[3] `http://www.cedar.buffalo.edu/Databases/CDROM1/` (2005 Oct, 20)

3.2 Experiment 2: Cars

The intention of the second experiment is to show the performance of the approach depending on the size of the object-memory. The car objects from the database provided by Ferencz has been used for evaluation. The object-memory was trained using 6 sample images for each object. To obtain the recall-precision curves we varied the threshold for the confidence of identification. If we have in a single layer multiple positive responses we evaluate multiple hypotheses.

Figure 6(a) analyzes the model complexity i.e. how many features are used for description at each node. We can see that if we use more than 15 features, performance increases only slightly. In addition, we varied the number of learned objects by using a fixed model complexity of 15 features. Up to 60 objects the system performs very well while for 120 objects the model complexity has to be increased. These results are comparable to [3].



(a) RPC varying the model complexity (30 objects have been learned).

(b) RPC varying the number of trained objects (using 15 features for an object model).

Figure 6: Experimental evaluations of the car experiment.

4 Conclusion

A hierarchical structuring technique has been introduced solving both - object identification and object categorization. In addition, the proposed algorithm can be used within real-time applications. The key-idea of the approach is the grouping of similar objects into layers and to organize them in a hierarchical manner. Boosting in combination with orientation histograms is used as feature selection method for distinguishing objects within a single layer. The property of just being discriminative within a single layer remarkably simplifies the identification task. The ability of categorizing is reached by higher layers after retraining with child-layers has been done. However, the effect of retraining of layers using information of their children belongs to current research. In addition we are investigating the effects of dynamically determining complexity for object models.

Acknowledgments

The project results have been developed in the MISTRAL Project (Measurable intelligent and secure semantic extraction and retrieval of multimedia data). MISTRAL is financed by the Austrian Research Promotion Agency (www.ffg.at). Furthermore this work has been sponsored in part by the Austrian Federal Ministry of Transport, Innovation and Technology under P-Nr. I2-2-26p VITUS2 (Video Image Analysis for Tunnel Safety).

References

[1] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[2] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions.," in *Proceedings of the British Machine Vision Conference*, 2002.

[3] A. Ferencz, E. Learned-Miller, and J. Malik, "Building a classification cascade for visual identification from one example," in *Proceedings of the International Conference on Computer Vision*, Washington, DC, USA, 2005.

[4] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2003, vol. 2, p. 264.

[5] Li Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshop*, 2004, pp. 178–187.

[6] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, 2004.

[7] M. Fink, "Object classification from a single example utilizing class relevance metrics," in *Advances in Neural Information Processing Systems 17*, Lawrence K. Saul, Yair Weiss, and Léon Bottou, Eds., pp. 449–456. MIT Press, Cambridge, MA, 2005.

[8] E. Bart and S. Ullman, "Cross-generalization: Learning novel classes from a single example by feature replacement," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2005, vol. 1, pp. 672–679.

[9] M.G. Miller, N.E. Matsakis, and P.A. Viola, "Learning from one example through shared densities on transforms," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2000, vol. 1, pp. 464–471.

[10] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2001, vol. 1, pp. 511–518.

[11] A. Torralba, K.P. Murphy, and W.T. Freeman, "Sharing features: efficient boosting procedures for multiclass object detection," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2005, vol. 2, pp. 762–769.

[12] S. Obdrzálek and J. Matas, "Sub-linear indexing for large scale object recognition," in *Proceedings of the British Machine Vision Conference*, 2005, vol. 1, pp. 1–10.

[13] V. Lepetit, P. Lagger, and P. Fua, "Randomized trees for real-time keypoint recognition," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2005, vol. 2, pp. 775 – 781.

[14] S. Edelman, N. Intrator, and T. Poggio, "Complex cells and object recognition," *Unpublished manuscript*.

[15] Fatih Porikli, "Integral histogram: A fast way to extract histograms in cartesian spaces," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, San Diego, CA, USA, June 2005, IEEE Computer Society.

[16] S. Ullman, M. Vidal-Naquet, and E. Sali, "Visual features of intermediate complexity and their use in classification," *Nature Neuroscience*, vol. 7, no. 5, pp. 1–6, 2002.

[17] P. Yang, Shiguang Shan, Wen Gao, Li S.Z., and Dong Zhang;, "Face recognition using ada-boosted gabor features," in *Proceedings of the Conference on Automatic Face and Gesture Recognition*, 2004, pp. 356–361.

[18] Sebastian Thrun, "Is learning the n-th thing any easier than learning the first?," in *Advances in Neural Information Processing Systems*, David S. Touretzky, Michael C. Mozer, and Michael E. Hasselmo, Eds. 1996, vol. 8, pp. 640–646, The MIT Press.