Report on Research during Student Exchange

A. Opelt[1], A. Zisserman[2]

[1] Graz University of Technology, Institute of Electrical Measurement and Measurement Signal Processing,
Schiestattgasse 14b, 8010 Graz, Austria
{opelt}@emt.tugraz.at
[2] Oxford University, Dept. of Engineering Science,
Oxford, OX1 3PJ, United Kingdom
az@robots.ox.ac.uk

Abstract. The aim of this research period is the investigation of different levels of geometry for generic object recognition. We started on a base level system without geometry using a codebook of appearances ([5]). Geometry was then investigated in two stages: First an obejct category is learned by using an objects class specific geometric constraint. Second we used loose geometric relations of object boundaries to learn a generic object representation.

1 Overview

The goal of the research is to learn to recognize object categories from examples. In a similar manner to which a child is shown examples of animals in picture books, the goal here is to learn a representation of a category from a set of images, e.g. of real and cartoon animals. This is a challenging task because of the usual Computer Vision problems of varying size and pose, varying lighting, partial occlusion and background clutter. However, the important challenge here is within-class variability - the many visual instances that are associated into a visual category, e.g. a motorbike, horse or human.

Object category recognition normally involve an appearance model and a geometric model. These two components can be used at different levels. In the previous work of e.g. Opelt *et al.* [9] object categories were just learned based on appearance, not using geometry at all. In contrast, the work of e. g. Fergus *et al.* [6] uses a very strong geometric model in combination with the object appearances. Our aim was to investigate the spectrum in between these approaches. Also we were interested in how geometry works with special object categories. So a class of e.g. spotted cats is less constrained in their geometry than e.g. bikes or cars.

Starting from a point of using no geometry and a codebook of the appearance of object parts (see section 2.1), we developed two novel geometric frameworks using *only* the object boundary (not its intensity appearance). First in section 2.2 we were interested in learning a specific object categtory by its constraints. We developed a bike-detector which uses the geometric model of a bike to gain better recognition results. Second (see section 2.3) we built a system which uses the shape of boundary-fragments of the object with their geomertic relation to the centre. Additionally we use a boosting framework such that a classifier learns the geometric configuration of the object boundary.

2 Geometry in three stages

2.1 Level-0, no geometry

As a first task in this exchange programm, we implemented a state-of-the-art recognition system [5]. This system was used to first explore the idea of using a codebook of visual parts for object recognition. Based on this baseline system we explored the usage of geometry combined with simple visual codebook entries. Results of this baseline system are not directly compareable with our work because it solves the easier task of image categorization instead of the detection of objects of a certrain category i.e. it can say if an object is in an image but not *where* it is in the image.

2.2 Level-1, category specific geometry

To use geometry at a first level we built a framework which uses boundary-fragments as codebook entries and was strongly inspired by the work of Leibe and Schiele [8]. The codebook is learned from a couple of positive training images by extracting the object boundary in a semi-supervised manner. This means we use the ground truth of the object location. We reduce clutter by pre-segmentation with mean-shift [4]. Each entry of a boundary-fragment consists of its shape and the position of the object centre. These boundary-fragments are further clustered by agglomerative clustering. As a distance we used the chamfer distance ([1]). We used bikes as a category with special geometric constraints. We extracted scale and orientation from the size and position of the wheels of a bike. Therfore we built a robust wheel detector (see figure 1 for examples). For comparison, Jurie and Schmid ([7]) recently presented an approach also tested on the category bikes.

Recognition is done by comparing the boundary-fragments of a new test image with all codebook entries. If a codebook entry matches it votes for the object centre in a hough space. With mean shift mode estimation ([2], [3]) we find the maxima in this hough voting space. This results in hypotheses of the object appearances in the image. We assume that images of a visual object category have special features in common.

2.3 Level-2, geometry of object parts

Here we extended our first idea to make it more general. Figure 2 shows examples of the category cows, that we used for this task. The training part of the framework is shown in figure 7. We use again the boundary-fragments and the ground truth for training. But now we optimize the selection of our codebook entries with respect to a validation set. This is done by throwing random seeds on the

Fig. 1. Examples of our wheel detection algorithm. The original images are shown in rows 1 and 3. The corresponding detected wheels are shown in rows 2 and 4.

contour of each training image. Based on these seed positions we grow boundary segments from which we calculate the costs of matching and localization stability on the validation set. This means we are interested in boundary-fragments that match often on the positive images in the validation set, and not often in the negative validation images, and have reliable voting on the object centres of the positive images. Figure 3 shows examples of this growing boundary segments on which we based our optimization. Examples of the results of this optimized codebook are shown in figure 4

We then use these optimized codebook entries and feed geometric combinations of k entries into a Boosting learning module. The weak classifiers are obtained by calculating a distance $d = \frac{1}{\sum sim_i * ms}$ to the validation set. sim_i stands for the similarity of the codebook entry $i = 1 \ldots k$ to the training image and ms is a factor that grows if the centres of the matched parts of the hypothesis are close together (which is a strong vote for the object centre). Figure 5 shows examples of the combination of codebook entries matched to the validation set.

This results in a final classifier consisting of a linear combination of these weak classifers for an overall categorization plus geometric relations for object

Fig. 2. Example images of the category cows.

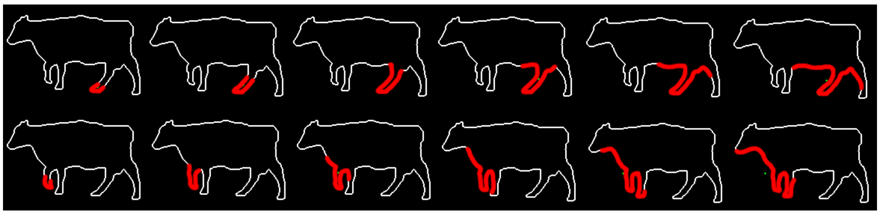

Fig. 3. Each row shows the growing of an boundary segment located at a random seed on the contour of a training image.

detection. Figure 6 shows examples of such learned weak classifiers for $k = 2$ and $k = 3$.

Figure 8 shows the procedure of classifiing a new test image and detecting the object location. We compute the output of the final Boosting classifier on this new edge image to get an idea of the image classification. To solve the shortcomings of approaches doing just this classification without a proper object detection are the second part of our testing procedure. There each hypothesis consisting of k codebook entries votes with its geometric information for an object centre in a hough voting space. With mean shift mode estimation ([2], [3]) we find the object location and give the approximated outline of the object. Figure 9 shows quanitiative results of our object recognition.

3 Conclusion and Outlook

In conclusion, we have presented research at three levels of geometry.

- We explored the idea of a codebook of image patches,
- and extended it to the use of boundary-fragments. Also the level of geometry was increased and
- Boosting was incorparated for a better recognition performance.

We are currently experimenting with these frameworks and are planing to submit to ICCV 2005. ECVision will be acknowledged in this publication.

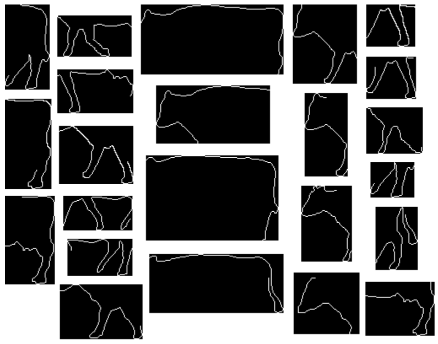

Fig. 4. Examples of the optimized codebook entries.

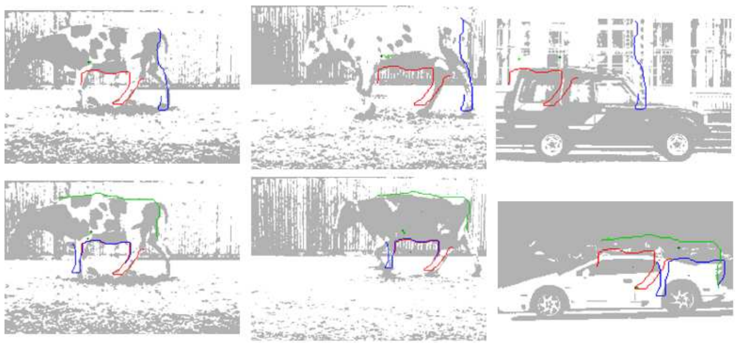

Fig. 5. Combinations of codebook entries matched on the validation images in order to find the best weak classifier in a boosting iteration. The first row shows examples with $k = 2$ and the second row uses $k = 3$.

References

1. H. Breu, . Gil, D. Kirkpatrick, and M. Werman. Linear time euclidean distance transform algorithms. *IEEE PAMI*, 17(5):529–533, May. 1995.
2. Y. Cheng. Mean shift mode seeking and clustering. *IEEE PAMI*, 17(8):790–799, Aug. 1995.
3. D. Comaniciu and P. Meer. Distribution free decomposition of multivariate data. *Pattern Analysis and Applications*, 2(1):22–30, 1999.
4. D. Comaniciu and P. Meer. Mean shift: A robust approach towards feature space analysis. In *IEEE PAMI*, volume 24(5), pages 603–619, 2002.
5. G. Csurka, C. Bray, C. Dance, and L. Fan. Visual categorization with bags of keypoints. In *Proc. European Conference on Computer Vision*, 2004.
6. R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc. Conference on Computer Vision and Pattern Recognition*, pages 264–272, 2003.
7. F. Jurie and C. Schmid. Scale-invariant shape features for recognition of object categories. In *Proc. of Conference on Vision and Pattern Recognition*, 2004.

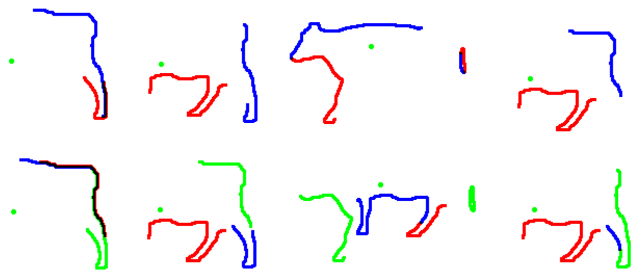

Fig. 6. Examples of learned weak classifiers. The first row shows examples with $k = 2$ and the second row uses $k = 3$.

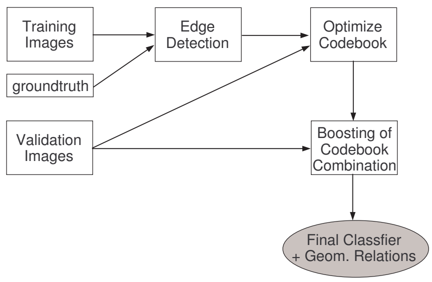

Fig. 7. Overview of the training procedure of our framework.

8. B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, number Vol. 2, pages 409 – 415, 2003.
9. A. Opelt, M. Fussenegger, A. Pinz, and P. Auer. Weak hypotheses and boosting for generic object detection and recognition. In *Proc. European Conference on Computer Vision*, volume 2, pages 71–84, 2004.

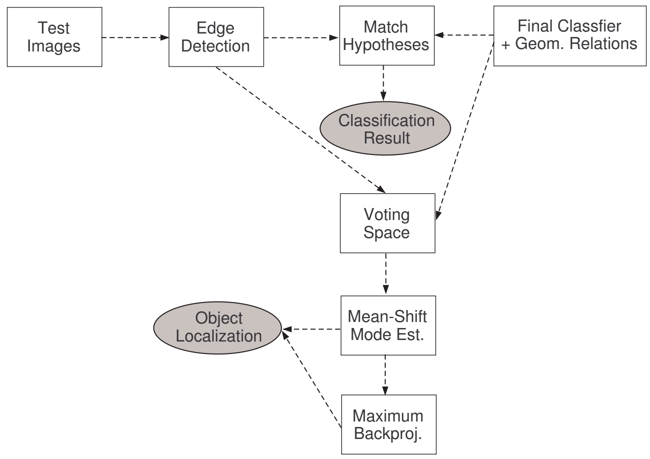

Fig. 8. Overview of the testing procedure of our framework.

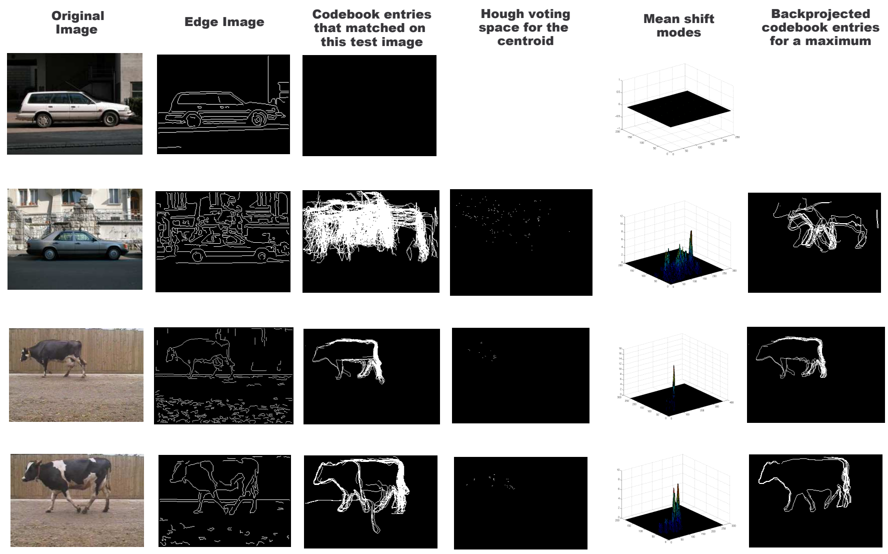

Fig. 9. Overview of the recognition procedure. The first row shows a correct test image where we have not recognized a cow. The second row shows the recognition of a false positive. The last two rows show correct detections of cows.