in *British Machine Vision Conference (BMVC'03)*, Norwich , UK, Sept. 2003.

Interleaved Object Categorization and Segmentation

Bastian Leibe and Bernt Schiele
Perceptual Computing and Computer Vision Group,
ETH Zurich, Switzerland, {leibe,schiele}@inf.ethz.ch

Abstract

Historically, figure-ground segmentation has been seen as an important and even necessary precursor for object recognition. In that context, segmentation is mostly defined as a data driven, that is bottom-up, process. As for humans object recognition and segmentation are heavily intertwined processes, it has been argued that top-down knowledge from object recognition can and should be used for guiding the segmentation process. In this paper, we present a method for the categorization of unfamiliar objects in difficult real-world scenes. The method generates object hypotheses without prior segmentation that can be used to obtain a category-specific figure-ground segmentation. In particular, the proposed approach uses a probabilistic formulation to incorporate knowledge about the recognized category as well as the supporting information in the image to segment the object from the background. This segmentation can then be used for hypothesis verification, to further improve recognition performance. Experimental results show the capacity of the approach to categorize and segment object categories as diverse as cars and cows.

1 Introduction

The traditional view of object recognition has been that prior to the recognition process, an earlier stage of perceptual organization occurs to determine which features, locations, or surfaces most likely belong together [11]. As a result, the segregation of the image into a figure and a ground part has often been seen as a prerequisite for recognition. In that context, segmentation is mostly defined as a bottom-up process, employing no higher-level knowledge. State-of-the-art segmentation methods combine grouping of similar image regions with splitting processes concerned with finding most likely borders [15, 14, 10]. However, grouping is mostly done based on low-level image features, like color or texture statistics, which require no prior knowledge. While that makes them universally applicable, it often leads to poor segmentations of objects of interest, splitting them into multiple regions or merging them with parts of the background [3].

Results from human vision indicate, however, that object recognition processes can operate before or intertwined with figure-ground organization and can in fact be used to drive the process [13, 16, 12]. This motivates us to explore how high-level knowledge can be used for grouping image regions belonging to the same object. The task we want to solve is object categorization, that is to recognize a-priori unknown objects of a given category in real-world scenes. Figure-ground segmentation in such settings is difficult because of clutter and large within-category variability of object colors, textures, and shapes.

In this paper, we present a local approach that generates object hypotheses without prior segmentation. The hypotheses are then used to obtain a category-specific figure-ground segmentation. We derive a probabilistic formulation of the problem that allows us to incorporate knowledge about the recognized category as well as the supporting information in the image. As a result, we obtain a segmentation mask of the object together with a per-pixel confidence estimate specifying how much this segmentation can be trusted. Thus, figure-ground segmentation is achieved as a result of object recognition. The following section discusses related work. Section 3 describes the learning of a codebook of local appearance for individual object categories, which can be used to generate object hypotheses. Based on these hypotheses, Section 4 then derives the segmentation algorithm. Finally, Section 5 presents experimental results.

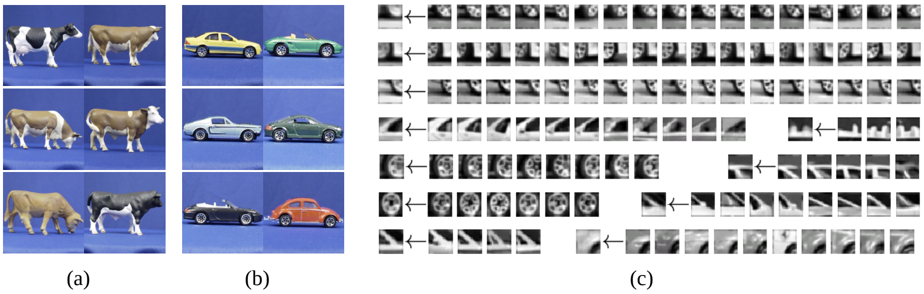
2 Related Work

The idea to use object-specific information for driving figure-ground segmentation has appeared in the literature before. Approaches, such as Deformable Templates [19], or Active Appearance Models [6] are typically used when the object of interest is known to be present in the image and an initial estimate of its size and location can be obtained. Examples of successful applications include tracking and medical image analysis.

Most directly related to our approach, Borenstein & Ullman represent object knowledge using image fragments and their figure-ground labeling from a training set [3]. Class-specific segmentations are obtained by fitting fragments to the image and combining them in jigsaw-puzzle fashion, such that their figure-ground labels form a consistent mapping. While the authors present impressive results for segmenting sideviews of horses, their approach includes no global recognition process. As only the local consistency of adjacent pairs of fragments is checked, there is no guarantee that the resulting cover really corresponds to an object and is not just caused by background clutter resembling random object parts. Our approach enforces global consistency by integrating segmentation with an object recognition process.

Yu & Shi also present a parallel segmentation and recognition system [18]. They formulate the segmentation problem in a graph theoretic framework that combines patch and pixel groupings. A set of 15 known objects is represented by local color, intensity and orientation histograms obtained from a number of different viewpoints. During recognition, these features are matched to patches extracted from the image to obtain object part hypotheses, which are combined with pixel groupings based on orientation energy. A final solution is found using the Normalized Cuts criterion [15]. This method achieves good segmentation results in cluttered real-world settings. However, their system needs to know the exact objects beforehand in order to extract their most discriminant features.

In our application, we do not require the objects to be known beforehand – only familiarity with the object category is needed. That means that the system needs to have seen some examples of e.g. cars and cows before, but those do not have to be the ones that are to be recognized later. Obviously, this makes the task more difficult, since we cannot rely on any object-specific feature, but have to compensate for large in-class variations. The following section describes how our algorithm achieves this by learning a codebook of local appearance.



(a) (b) (c)

Figure 1: *(a,b) Some of the training objects used for cows and cars (from the ETH-80 database [8]). From each object, 16 views were taken from different orientations. (c) Example codebook clusters for cars with their corresponding patches.*

3 A Codebook of Local Appearance for Object Categorization

In order to generate a codebook of local appearances of a particular object category, we use an approach inspired by [17, 1]. From a variety of images (in our case 160 images corresponding to 16 views around the equator of each of the 10 training objects shown in Figure 1(a,b)), image patches of size 25 × 25 pixels are extracted with the Harris interest point detector [7]. Starting with each patch as a separate cluster, agglomerative clustering is performed: the two most similar clusters C_1 and C_2 are merged as long as the average similarity between their constituent patches (and thus the cluster compactness) stays above a certain threshold t:

$$similarity(C_1,C_2) = \frac{\sum_{p\in C_1, q\in C_2} NGC(p,q)}{|C_1| \times |C_2|} > t, \tag{1}$$

where the similarity between two patches is measured by Normalized Greyscale Correlation (NGC):

$$NGC(p,q) = \frac{\sum_i (p_i - \overline{p_i})(q_i - \overline{q_i})}{\sqrt{\sum_i (p_i - \overline{p_i})^2 \sum_i (q_i - \overline{q_i})^2}} \tag{2}$$

This clustering scheme guarantees that only those patches are grouped which are visually similar, and that the resulting clusters stay compact, a property that is essential for later processing stages. From each resulting cluster, we compute the cluster center and store it in the codebook.

Figure 1(c) shows some of the codebook entries, together with the patches they were derived from. With a value of $t = 0.7$, the 8'269 extracted car image patches are reduced to a codebook of size 2'519. While the resulting number of clusters is still high, the most interesting property of the clustering scheme is that all clusters are compact and only contain image patches that are visually similar.

Rather than to use this codebook directly to train a classifier as in [1], we propose to use a probabilistic voting scheme which produces comparable results. For this, the extracted image patches are matched to the codebook using the NGC measure. In contrast to [1], though, we do not activate the best-matching codebook entry only, but all entries whose similarity is above t, the threshold already used during clustering. For every codebook entry, we store all the positions it was activated in, relative to the object center.

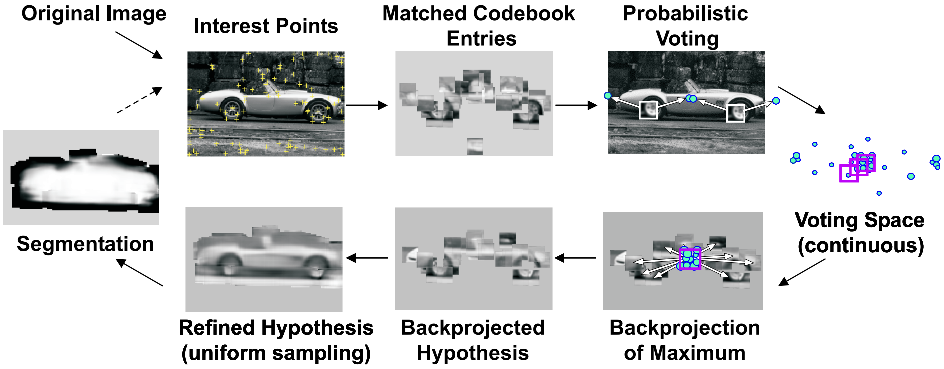


Figure 2: *The recognition procedure. Image patches are extracted around interest points and compared to the codebook. Matching patches then cast probabilistic votes, which lead to object hypotheses that can later be refined. Based on the refined hypotheses, we compute a category-specific segmentation.*

During recognition, we use this information to perform a Generalized Hough Transform [2, 9]. Given a test image, we extract image patches and match them to the codebook to activate codebook entries. Each activated entry then casts votes for possible positions of the object center. Figure 2 illustrates this procedure. We search for hypotheses as maxima in the continous vote space using Mean-Shift Mode Estimation [4, 5]. For promising hypotheses, all patches that contributed to it can be collected (Fig. 2(bottom)), therefore visualizing what the system reacts to. Moreover, we can refine the hypothesis by sampling all the image patches in its surroundings, not just those locations returned by the interest point detector. As a result, we get a representation of the object including a certain border area.

In the following, we cast this recognition procedure into a probabilistic framework. Let **e** be our evidence, an extracted image patch. Each image patch may have several valid interpretations I_i, namely the matching codebook clusters. Each interpretation is weighted with the probability $p(I_i|\mathbf{e})$. If a codebook cluster matches, it can cast its votes for different object positions. That is, for every I_i, we can obtain votes for several object identities o_n and positions x_j, which we weight with $p(o_n,x_j|I_i)$. Thus, any single vote has the weight $p(o_n,x_j|I_i)p(I_i|\mathbf{e})$, and the patch's contribution to the hypothesis is

$$p(o_n,x_j|\mathbf{e}) = \sum_i p(o_n,x_j|I_i)p(I_i|\mathbf{e}). \tag{3}$$

By basing the decision on single-patch votes and assuming a uniform prior for the patches, we obtain

$$p(o_n,x_j) \sim \sum_k p(o_n,x_j|\mathbf{e}_k). \tag{4}$$

From this probabilistic framework, it immediately follows that the $p(I_i|\mathbf{e})$ and $p(o_n,x_j|I_i)$ should both sum to one. In our experiments, we assume a uniform distribution for both (meaning that we set $p(I_i|\mathbf{e}) = \frac{1}{|I|}$, with $|I|$ the number of matching codebook entries), but it would also be possible, for example, to let the $p(I_i|\mathbf{e})$ distribution reflect the relative matching scores.

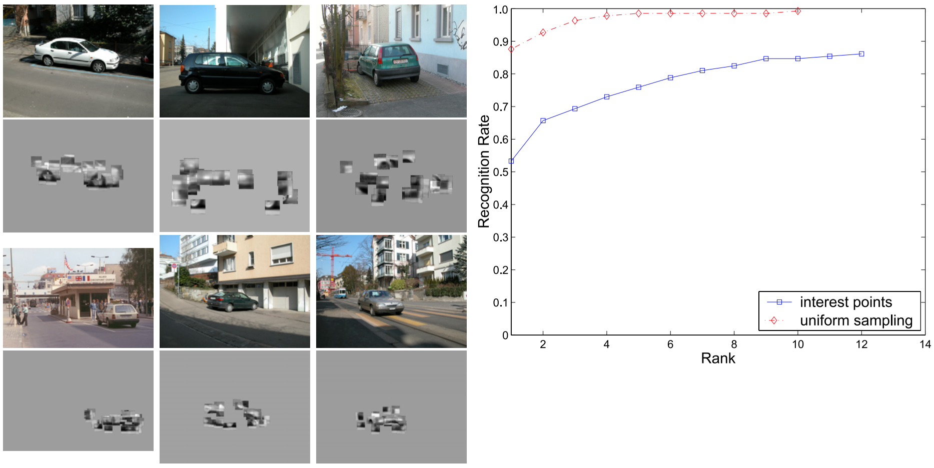


Figure 3: *(left) Example car images and recognition results from the test set (images 1-5: 1st hypothesis; image 6: 7th hypothesis). (right) Quantitative recognition results if all hypotheses up to a certain rank are considered.*

In order to evaluate the system's recognition capability, we have applied it to a database of 137 images of real-world scenes containing one car each in varying poses. Based on interest points, the system is able to correctly recognize and localize 53.3% of the cases with its first hypothesis and up to 86.1% with the first 12 hypotheses. Taking all available patches by uniform sampling, performance improves to 87.6% with the first hypothesis and 98.5% with the first 5 hypotheses[1]. Figure 3 shows the quantitative recognition results and some example images from the test set. These results clearly show the system's ability to categorize objects in a variety of different poses. In the following, we want to extend the approach to obtain pose-specific segmentations of objects. In the context of this paper, we explore in particular the possibility to segment side views of cars and cows.

4 Object Segmentation

In this section, we derive a probabilistic formulation for the segmentation problem. As a starting point, we take a refined object hypothesis (o_n, x) obtained by the algorithm from the previous section. Based on this hypothesis, we want to segment the object from the background.

Up to now, we have only dealt with image patches. For segmentation, we now want to know whether a certain image pixel $\mathbf{p}$ is *figure* or *ground*, given the object hypothesis. More precisely, we are interested in the probability $p(\mathbf{p} = \mathit{figure}|o_n, x)$. The influence of a given patch $\mathbf{e}$ on the object hypothesis can be expressed as

$$p(\mathbf{e}|o_n,x) = \frac{p(o_n,x|\mathbf{e})p(\mathbf{e})}{p(o_n,x)} = \frac{\sum_I p(o_n,x|I)p(I|\mathbf{e})p(\mathbf{e})}{p(o_n,x)} \tag{5}$$

where the patch votes $p(o_n, x|\mathbf{e})$ are obtained from the codebook, as described in the previous section. Given these probabilities, we can obtain information about a specific

[1] Since the object size in our images is roughly twice that of Agarwal & Roth's [1], we double the tolerances used in their evaluation and accept a hypothesis if $\delta_x \leq 56$, $\delta_y \leq 28$, and bounding box overlap is above 50%.

pixel by summing over all patches that contain this pixel:

$$p(\mathbf{p} = \mathit{figure}|o_n,x) \quad = \quad \sum_{\mathbf{p}\in\mathbf{e}} p(\mathbf{p} = \mathit{figure}|\mathbf{e},o_n,x)p(\mathbf{e}|o_n,x) \tag{6}$$

with $p(\mathbf{p} = \mathit{figure}|\mathbf{e},o_n,x)$ denoting patch-specific segmentation information, which is weighted by the influence $p(\mathbf{e}|o_n,x)$ the patch has on the object hypothesis. Again, we can resolve patches by resorting to learned patch interpretations I stored in the codebook:

$$\begin{aligned} p(\mathbf{p} = \mathit{figure}|o_n,x) &= \sum_{\mathbf{p}\in\mathbf{e}} \sum_{I} p(\mathbf{p} = \mathit{figure}|\mathbf{e},I,o_n,x)p(\mathbf{e},I|o_n,x) & (7) \\ &= \sum_{\mathbf{p}\in\mathbf{e}} \sum_{I} p(\mathbf{p} = \mathit{figure}|I,o_n,x)\frac{p(o_n,x|I)p(I|\mathbf{e})p(\mathbf{e})}{p(o_n,x)}. & (8) \end{aligned}$$

This means that for every pixel, we build a weighted average over all segmentations stemming from patches containing that pixel. The weights correspond to the patches' respective contributions to the object hypothesis. For the *ground* probability, the result can be obtained in an analogue fashion.

The most important part in this formulation is the per-pixel segmentation information $p(\mathbf{p} = \mathit{figure}|I,o_n,x)$, which is only dependent on the matched codebook entry, no longer on the image patch. If we store a fixed segmentation mask for every codebook entry (similar to Borenstein & Ullman's approach [3]), we obtain a reduced probability $p(\mathbf{p} = \mathit{figure}|I,o_n)$. In our approach, we remain more general by keeping a separate segmentation mask for every stored *occurrence position* of each codebook entry. We thus take advantage of the full probability $p(\mathbf{p} = \mathit{figure}|I,o_n,x)$. The following section describes in more detail how this is implemented in practice.

4.1 Implementation

For learning segmentation information, we make use of a high-quality figure-ground segmentation mask that is available for each of our training images. We can thus obtain a figure-ground mask for any image patch from the training data. In this paper, we have experimented with two different ways of integrating segmentation information into the system, corresponding to the different interpretations of the probability $p(\mathbf{p} = \mathit{figure}|I,o_n,x)$ described above.

In the first approach, as inspired by Borenstein & Ullman [3], we store a segmentation mask with every image patch obtained from the training images. When the patches are clustered to form codebook entries, the mask coherence is integrated into the similarity measure used for clustering. Thus, it is ensured that only patches with similar segmentation masks, in addition to similar appearance, are grouped together. Whenever a codebook entry is matched to the image during recognition, its stored segmentation mask is applied to the image. The entry may cast votes for different object identities and positions, but whatever it votes for, the implied segmentation mask stays the same. When an object hypothesis is formed as a maximum in vote space, all patch interpretations contributing to that hypothesis are collected, and their associated segmentation masks are combined to obtain the per-pixel probabilities $p(\mathbf{p} = figure|o_n,x)$.

In the second approach, pioneered in this paper, we do not keep a fixed segmentation mask for every codebook entry, but we store a separate mask for every location it occurs

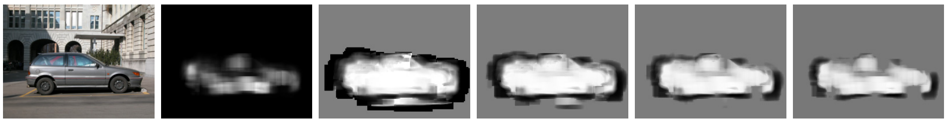

(a) image (b) confidence (c) $\theta = 0.0$ (d) $\theta = 0.1$ (e) $\theta = 0.4$ (f) $\theta = 1.0$

Figure 4: *Segmentation results with different confidence levels θ.*

in on the training images. With the 2’519 codebook entries used for the car category, we thus obtain 20’359 occurrences, with one segmentation mask stored for each. For the cow category, the codebook contains only 2’244 clusters, but these occur in a total of 50’792 locations on the training images, owing to the larger texture variability on the cow bodies. Whenever a codebook entry is matched to the image using this approach, a separate segmentation mask is associated with every object position it votes for. As such, the same vertical structure can indicate a solid area if it is in the middle of a cow’s body, and a strong border if it is part of a leg. Which option is finally selected depends on the winning hypothesis and its accumulated support from other patches. In any case, the feedback loop of only taking the votes that support the winning hypothesis ensures that only consistent interpretations are used for the later segmentation.

In our experiments, we obtained much better results with the occurrence masks, even when edge information was used to augment matches. In the following, we therefore only report results for occurrence masks. In addition, we assume uniform priors for $p(\mathbf{e})$ and $p(o_n,x)$, so that these elements can be factored out of the equations. In order to obtain a segmentation of the whole image from the figure and ground probabilities, we build the likelihood ratio for every pixel:

$$L = \frac{p(\mathbf{p} = \mathit{figure}|o_n,x)}{p(\mathbf{p} = \mathit{ground}|o_n,x)}. \tag{9}$$

Figure 4 shows an example segmentation of a car, together with $p(\mathbf{p} = \mathit{figure}|o_n,x)$, the system’s confidence in the segmentation result. The lighter a pixel, the higher its probability of being *figure*. The darker it is, the higher its probability of being *ground*. The uniform gray region in the background does not contribute to the object hypothesis and is therefore considered neutral. By only considering pixels where $\max(p(\mathit{figure}), p(\mathit{ground})) > \theta$, the computed probability can be used to set a certain ”confidence level” for the segmentation and thus limit the amount of missegmentation. Figures 4(c)-(f) show segmentation results with different confidence levels (The confidences are not in the range $[0,1]$ because we omitted a normalization factor in the implementation). As can be observed, the segmentation with the lowest confidence level still contains some missegmented areas, while higher confidence levels ensure that only trusted segmentations are made, although at the price of leaving open some uncertain areas. This estimate of how much the obtained segmentation can be trusted is especially important when the results shall later be combined with a bottom-up segmentation method, e.g. based on contour grouping.

5 Results

The enlargement shown in Figures 5(a)-(e) demonstrates the advantage of the proposed approach compared to gradient-based methods. At the bottom of the car, there is no visible border between the black car body and the dark shadow underneath. Instead, a

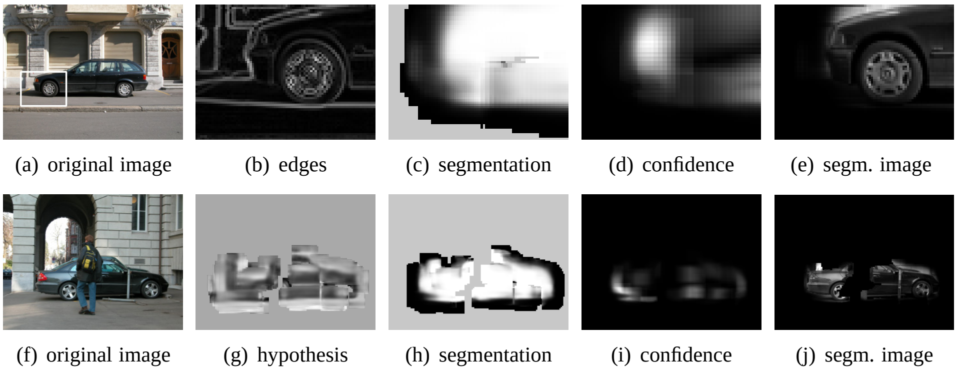


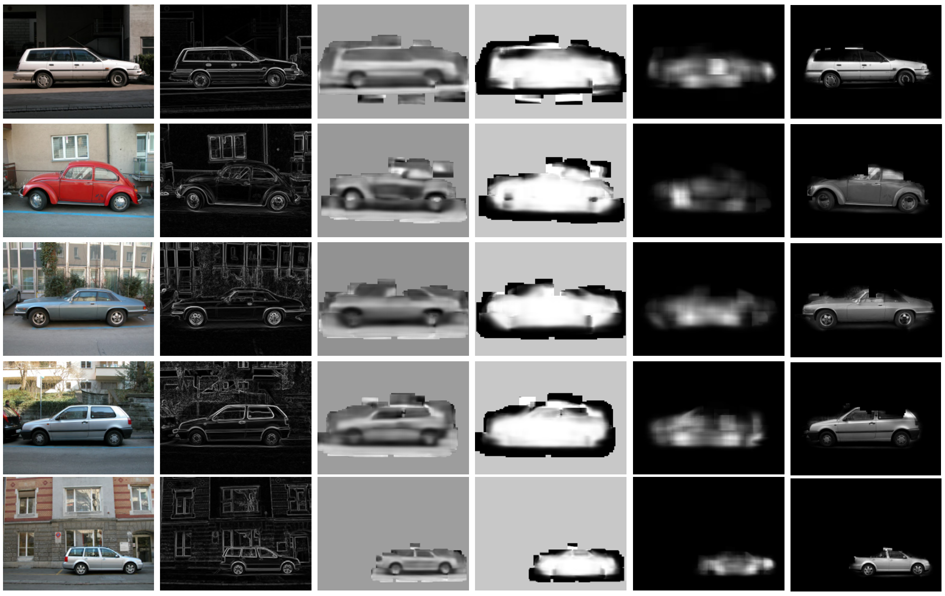
Figure 5: *(top) Example where object knowledge compensates for missing edge information. (bottom) Segmentation result of a partially occluded car. The system is able to segment out the pedestrian, because it contributes nothing to the car hypothesis.*

strong shadow line extends much further to the left of the car. The proposed algorithm can compensate for that since it ”knows” that if a codebook entry matches in this position relative to the object center, it must contain the car’s border. Since at this point only those patch interpretations are considered that are consistent with the object hypothesis, the system can infer the missing contour.

Figures 5(f)-(j) show another interesting case. Even though the car in the image is partially occluded by a pedestrian, the algorithm finds it with its second hypothesis. Refining the hypothesis yields a good segmentation of the car, without the occluded area. The system is able to segment out the pedestrian, because it contributes nothing to the car hypothesis. This is something that would be very hard to achieve for a system purely based on pixel-level discontinuities.

More segmentation results for cars and cows can be seen in Figures 6 and 7. All the cars and the first three cows have been correctly found with the recognition system’s first hypothesis (The last cow was found with the second hypothesis). Next to each test image, the gradient magnitude is shown to illustrate the difficulty of the segmentation task. Even though the images contain low contrast and significant clutter, the algorithm succeeds in providing a good segmentation of the object. Confidence and segmentation quality are especially high for the bottom parts of the cars, including the cars’ shadows (which were labeled *figure* in the training examples). Most difficulties arise with the car roofs and cow heads. These regions contain a lot of variation (e.g. caused by (semi-) transparent windows or different head orientations), which is not sufficiently represented in the training data. What is remarkable, though, is that the cows’ legs are captured well, even though no single training object contained exactly the same leg configuration. The local approach can compensate for that by combining elements from different training objects.

Another interesting effect can be observed in the cow images 1 and 4. Even though there are strong edge structures on the cows’ bodies, no borders are introduced there, since the system has learned that those edges belong to the body. On the other hand, relatively weak edges around the legs lead to strong segmentation results. The system has learned that if a certain structure occurs in this region, it must be a leg. No heuristics are needed for this behavior – it is entirely learned from training data.



(a) original (b) edges (c) hypothesis (d) segmentation (e) confidence (f) segm. image

Figure 6: *Example results for car images.*

6 Conclusion

In this paper, we have proposed an algorithm that achieves figure-ground segmentation as a result and extension of object recognition. The method uses a probabilistic formulation to integrate learned knowledge about the recognized category with the supporting information in the image. As a result, it returns a figure-ground segmentation for the object, together with a per-pixel confidence estimate specifying how much this segmentation can be trusted. We have applied the method to the task of categorizing and segmenting unfamiliar objects in difficult real-world scenes. Experiments show that it works for categories as diverse as cars and cows and that it can cope with cluttered backgrounds and partial occlusions.

For more accurate segmentation results, obviously, the combination with traditional contour or region based segmentation algorithms is required. The result images show that edges are quite prominent in those regions where our proposed algorithms has problems, such as on the car roofs or cow heads. On the other hand, category-specific knowledge can serve to resolve ambiguities between low-level image structures in those regions where our algorithm is confident. In short, both kinds of methods are mutually beneficial and should be combined, ideally in an iterative process. The probabilistic formulation of our algorithm lends itself to an easy integration with other segmentation methods.

Acknowledgments: This work is part of the CogVis project, funded in part by the Comission of the European Union (IST-2000-29375), and the Swiss Federal Office for Education and Science (BBW 00.0617). The authors would like to thank Andreas Remund for his contribution to the code.

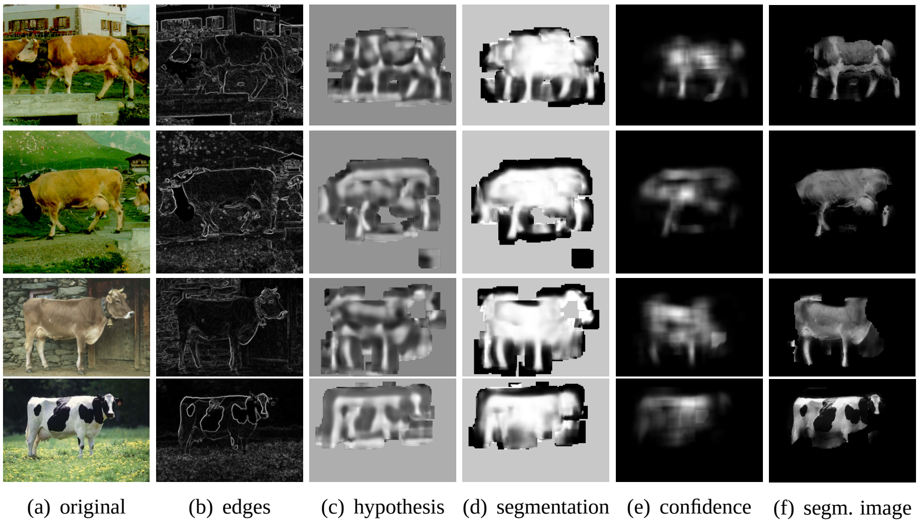

(a) original (b) edges (c) hypothesis (d) segmentation (e) confidence (f) segm. image

Figure 7: *Example results for cow images.*

References

[1] S. Agarwal and D. Roth. Learning a sparse representation for object detection. In *ECCV'02*, 2002.

[2] D.H. Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981.

[3] E. Borenstein and S. Ullman. Class-specific, top-down segmentation. In *ECCV'02*, LNCS 2353, pages 109–122, 2002.

[4] Y. Cheng. Mean shift mode seeking and clustering. *Trans. PAMI*, 17(8):790–799, Aug. 1995.

[5] D. Comaniciu and P. Meer. Distribution free decomposition of multivariate data. *Pattern Analysis and Applications*, 2(1):22–30, 1999.

[6] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active appearance models. In *ECCV'98*, 1998.

[7] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, pages 147–151, 1988.

[8] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *CVPR'03*, Madison, WI, June 2003.

[9] D. Lowe. Object recognition from local scale invariant features. In *ICCV'99*, 1999.

[10] J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *IJCV*, 43(1):7–27, 2001.

[11] D. Marr. *Vision*. W.H. Freeman, San Francisco, 1982.

[12] A. Needham. Object recognition and object segregation in 4.5-month-old infants. *J. Exp. Child Psych.*, 78(3):3–24, 2001.

[13] M.A. Peterson. Object recognition processes can and do operate before figure-ground organization. *Current Directions in Psychological Science*, 3:105–111, 1994.

[14] E. Sharon, A. Brandt, and R. Basri. Fast multiscale image segmentation. In *CVPR'00*, pages 70–77, 2000.

[15] J. Shi and J. Malik. Normalized cuts and image segmentation. In *CVPR'97*, pages 731–737, 1997.

[16] S.P. Vecera and R.C. O'Reilly. Figure-ground organization and object recognition processes: An interactive account. *J. Exp. Psych.: Human Perception and Performance*, 24(2):441–462, 1998.

[17] M. Weber, M. Welling, and P. Perona. Unsupervised learning of object models for recognition. In *ECCV'00*, 2000.

[18] S.X. Yu and J. Shi. Object-specific figure-ground segregation. In *CVPR'03*, 2003.

[19] A.L. Yuille, D.S. Cohen, and P.W. Hallinan. Feature extraction from faces using deformable templates. In *CVPR'89*, 1989.