From Meaningful Contours to Discriminative Object Shape

Pradeep Yarlagadda and Björn Ommer

University of Heidelberg,
Speyererstr. 6, 69115 Heidelberg, Germany
{pradeep.yarlagadda,bjoern.ommer}@iwr.uni-heidelberg.de

Abstract. Shape is a natural, highly prominent characteristic of objects that human vision utilizes everyday. But despite its expressiveness, shape poses significant challenges for category-level object detection in cluttered scenes: Object form is an emergent property that cannot be perceived locally but becomes only available once the whole object has been detected and segregated from the background. Thus we address the detection of objects and the assembling of their shape simultaneously. A dictionary of meaningful contours is obtained by clustering based on contour co-activation in all training images. We seek a joint, consistent placement of all contours in an image, since placing them independently from another is not reliable due to the emergence of shape. Therefore, the characteristic object shape is learned by discovering spatially consistent configurations of all dictionary contours using maximum margin multiple instance learning. During recognition, objects are detected and their shape is explained simultaneously by optimizing a single cost function. We demonstrate the benefit of our approach on standard shape benchmarks.

1 Introduction

Category-level object detection in cluttered scenes requires object models that can handle the large intra-class variability and, at the same time, accurately segregate objects from background clutter to avoid distraction by the background and achieve exact localization. Shape-based models provide an effective approach for accurately explaining meaningful object pixels in an image. The fundamental challenge of shape representation is, however, that object form (i.e. the Gestalt) cannot be perceived locally. Unlike color or texture which can be captured by a small image region, the prototypical shape of an object like a giraffe cannot be understood based on local measurements. Shape is an emergent property that becomes only apparent after all the object boundary contours (or, in dual form, its regions) have been grouped. At the same time, invariance w.r.t. missing, occluded parts and intra-class variation require that incomplete Gestalt needs to be dealt with while inter-class similarity renders it futile to detect objects based on single contours, e.g., the leg of a giraffe might resemble the outline of a bottle.

A. Fitzgibbon et al. (Eds.): ECCV 2012, Part I, LNCS 7572, pp. 766–779, 2012.

This leads to a fundamental question: how can we represent shape, if it cannot be measured directly? Although there has been significant progress in edge detection and segmentation (e.g. [1, 2]), segmentation is an ill-posed problem and thus bottom-up contour extraction is intrinsically limited [3]. To avoid the shortcomings of purely image-driven contour extraction, we follow a model-based approach (e.g. [4]) where we search with model contours that have been learned during training. Given a set of training images, contours are extracted and verified against the other training images to make up for the unreliability of the contour extraction process. This overcomplete set of contours needs to be condensed into a feasible sized codebook. However, we do not follow the standard grouping based on visual similarity plus relative part location (e.g. [4]) as this fails when contours are corrupted by the extraction process. Rather we propose a clustering based on the activation pattern of contours where contours are grouped if they are activated similarly in a number of training images.

Although we now have a set of meaningful contours, matching them independently to novel query images (e.g. [4, 5]) still poses robustness issues due to the large intra-class variability. Therefore, we optimize the *joint placement* of all contours which maximally *discriminates* objects from non-objects. But how can we learn meaningful co-placements of contours? During training these optimal *compositions* [6, 7] are not provided and the placement of individual contours is noisy. Therefore, we utilize *multiple instance learning* (MIL) and propose a number of candidate compositions of contours. Given positive and negative bounding boxes, MIL then selects a set of joint placements of codebook contours that are consistent among training images and optimally discriminate objects from non-objects. In addition each codebook contour receives a weight indicating how meaningful it is for discrimination.

Consequently, the difficult questions of selecting meaningful contours and finding consistent co-placements of these contours are shifted to the training phase. Here they can be addressed by optimization over an ensemble of training images rather then just a single query image.

In this work, i) we generate a dictionary of contours based on their co-activation patterns over an ensemble of training images ii) we learn the joint placement of all codebook contours that maximizes the discrimination between class and non-class structure using max margin multiple instance learning and iii) we detect objects and assemble their shape at the same time by optimizing a single cost function that finds consistent joint placements of all dictionary contours. The contributions are summarized in Fig. 1.

2 Related Work

The most prominent approach to category-level object detection in cluttered scenes are currently part-based models using local or semi-local descriptors. Based on appearance patches [8, 9], SIFT [10], geometric blur [11], and other texton-like features [12] local image information is extracted and then combined in a spatial model. These models range from no spatial relationships like bag-of-features [13], conditionally independent parts in voting methods [9, 14, 15]

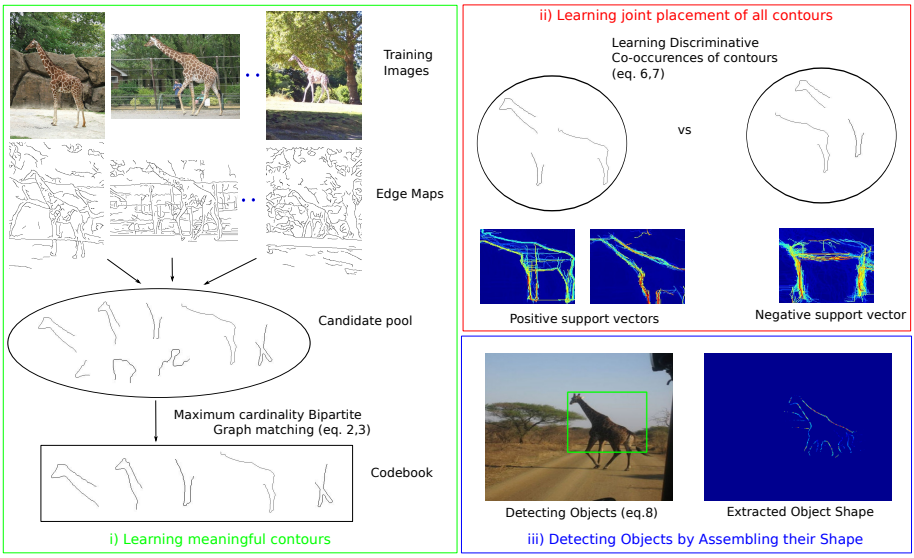

Fig. 1. Overview of the approach. i) Obtaining meaningful contours from a collection of training images, ii) learning discriminative contour co-occurences and iii) using such co-occurences in detecting an object and extracting its shape in a query image.

and pictorial structures [16], over rigid, grid-like structures to joint models of all parts [8] like the constellation model. While the less complex models like Hough voting [15, 17] and bag-of-features can handle large numbers of parts, rich spatial models like constellation models are typically restricted to only few parts (usually less than ten). Our goal is to represent the rich spatial structure of object shape and still utilize a large number of contour parts since this provides for robustness w.r.t. occlusion or noisy contour information.

At the other end of the modeling spectrum are template methods [18] and holistic, texton-based object representations like Histogram of Oriented Gradients combined with complex part-based models that are limited to few parts [19]. Rather than learning object contours and their joint configuration from training data, [20] utilize a hand drawn model for each object class to build a global boundary based shape representation. Detection and segmentation are then tackled by means of semidefinite programming. [21] learn the model parameters of a hierarchical configurable deformable template by extending the Max-Margin learning to AND/OR graphs. Another interesting line of work is based on hierarchical models for object detection [22] and parsing [23–25]. [26] represent object form by applying active shape models.

Obviously holistic models face limitations when objects feature significant articulation and their shape shows high variability. Consequently, the majority of shape-based detection methods are based on spatially flexible matching algorithms and deformable part configurations. More specifically, [27] present a shape based approach based on the partial matching of edge fragments. [28]

utilize a many-to-one matching of contours from query images to a sparse set of model contours. Both approaches require a bottom-up grouping of edge pixels in a query image rather than matching previously learned contours directly onto the edge image. Moreover, the models of [28] consist only of two contours and they are matched independently, whereas our approach optimizes the joint placement of a large number of model contours. [15] learn the discriminative weights for each codebook entity (a semi-local region represented using a texton feature, i.e., geometric blur) based on a weighted sum of all potential matches for each codebook entity. Thus, the codebook co-activation information is lost in the summation process. We consider multiple placements for each model contour in a training image and treat the most relevant co-activation pattern of all contours as hidden variable. We learn this hidden variable and the weights for the codebook co-activations in a max margin MIL framework. In a query image, we optimize the joint placement of all codebook contours to form an object hypothesis rather than letting them vote independently from another. [4] use a dictionary of contour fragments in a boosting framework to perform category level object recognition. Rather than jointly placing all parts, each fragment is again positioned individually. Therefore, the approach is limited when dealing with articulated objects like giraffes, where the relative configuration of parts differs across different instances. In contrast, we directly learn the importance of each contour based on the joint placement of all of our model contours. In contrast to [4, 5], we obtain a dictionary of codebook contours based on their co-activation patterns over all training images rather than using merely the visual similarity between contours.

3 Learning Object Shape

The goal of object shape models is to capture the characteristic form of all instances of an object category. However, representing the form of ensembles of boundary pixels so that the resulting shape discriminates class from non-class structure (i.e. objects from background and other objects) is a challenging problem. Due to large intra-class variability, partial occlusion, and other influences from the environment, modeling and searching directly for the holistic object shape is infeasible. The converse process, a bottom-up edge pixel grouping that is driven predominantly by the query image is also futile, since image-based contour extraction is fundamentally limited and will not provide the complex holistic object shape [3]. Moreover, shape becomes only apparent when all object contours have been grouped. Therefore, placing individual local features or contour fragments independent from another as in [5, 15] is not reliable and tends to produce spurious hypotheses. Rather, contour parts need to influence each other and so object detection and the joint placement of all boundary contours for obtaining the overall shape are two intimately related problems that need to be tackled jointly. The complex, holistic shape of objects is only available once the object is detected and shape-based detection requires a successful grouping of boundary pixels, e.g., by finding a consistent configuration of candidate contours that captures the characteristic shape of objects.

Extracting meaningful contours from a query image and grouping them consistently to obtain the overall object shape is a notoriously difficult problem. Therefore, we are shifting this challenge to the training stage where a set of images are available so that groupings from one image can be verified on the others. In Sect 3.1, we obtain a codebook of meaningful contours by clustering contour fragments based on their co-activation on all the training images rather than merely using their visual similarity. Training images are only annotated with bounding boxes. Consequently, there is an uncertainty when placing codebook contours individually. Rather than integrating over spurious matches (e.g. [15] Eq.11), we present an approach based on multiple instance learning (Sect 3.2) to group contours and obtain the exact placement for each codebook contour. Based on discriminative learning, characteristic groupings of contours are then obtained that separate objects from clutter.

3.1 Learning Meaningful Contours

To obtain a set of meaningful contours from the training images, we first compute the probabilistic edge maps for each image using [1]. We follow the standard procedure of normalizing the provided object bounding boxes so that they have the same scale and aspect ratio. Thereafter, we peform edge linking to obtain lists of connected edge pixels. In a next step, we extract a set of non-disjoint contours from each linked edge segment by first computing points of high curvature and considering the midpoints between them. Randomly selecting pairs of these points from an edge and taking the contour segments in between yields a set of candidate contours. Each contour has a shift vector $\mathbf{s}^{c_i}$ from its centroid to the center of the bounding box. Combining all the segments from all training images yields on the order of 10^4 contours. Many of these are redundant and the size of this set needs to be reduced to a compact, feasible sized subset of meaningful contours.

A common approach is to cluster contours based on their visual similarity, potentially also adding the relative location in the image [4]. However, such a clustering founded on visual similarity, e.g. based on the chamfer distance between contours c_i and c_j, has deficits. For instance, contours that are fractured or corrupted by noise can fall in different clusters although they are matched to similar locations in the training images.

Therefore, we compute the pairwise dissimilarity matrix Δ_{ij} for all pairs c_i, c_j not by means of their visual similarity but based on where they match in an ensemble of training images. We use fast directional chamfer matching [29] for obtaining matching locations for each candidate contour in each training image. Let $A^i_{m,h}$ denote the m-th match of contour c_i in training bounding box h. $\mathcal{E}^h$ denotes the edge map of h. For the m-th match, we record the chamfer score $\gamma_m(c_i, \mathcal{E}^h)$ and the location of the match in the image $l_m(c_i, \mathcal{E}^h)$ (see Sect. 4.1),

$$A^i_{m,h} := \left(\gamma_m(c_i, \mathcal{E}^{t_h}), l_m(c_i, \mathcal{E}^{t_h}), \mathbf{s}^{c_i}\right)^\top \tag{1}$$

We cluster the contours based on their activation patterns A^i over all the training images.

$$A^i := \begin{bmatrix} A^i_{1,1}\ A^i_{1,2}\ \dots \\ A^i_{2,1}\ \dots \\ A^i_{3,1}\ \dots \\ \vdots \end{bmatrix} \tag{2}$$

We compute the dissimilarity matrix Δ_{ij} as $\Delta_{ij} := \sum_h \Theta\big(A^i_{\bullet,h}, A^j_{\bullet,h}\big)$. The dissimilarity Θ of both contours on training image h is obtained using *maximum cardinality bipartite matching*. For the bipartite matching, the elementary distance between the m-th match of c_i and the m'-th match of c_j is defined as

$$\begin{aligned} \theta_{i,j}(h,m,m') :=& |\gamma_m(c_i, \mathcal{E}^{t_h}) - \gamma_{m'}(c_j, \mathcal{E}^{t_h})| + \frac{\|\mathbf{s}^{c_i} - \mathbf{s}^{c_j}\|}{\max(|c_i|, |c_j|)} \\ &+ \frac{\|l_m(c_i, \mathcal{E}^{t_h}) + \mathbf{s}^{c_i} - l_{m'}(c_j, \mathcal{E}^{t_h}) - \mathbf{s}^{c_j}\|}{\nu} \end{aligned} \tag{3}$$

where ν is the average length of all object bounding box diagonals in the training data. Given the pairwise dissimilarity matrix Δ_{ij} we perform pairwise clustering using *Ward's method* and obtain a codebook $\mathcal{C}$ that contains on the order of 10^2 contours. The representative for each cluster is the element that has maximal average affinity to all elements in this cluster.

3.2 Learning a Discriminative Model for Object Shape

Given the codebook $\mathcal{C}$, we need to learn how to jointly place all the contours so that the overall configuration optimally discriminates the shape of objects from non-objects. During the training stage, we are only provided groundtruth for the bounding box of objects, but obviously not for the placement of contours therein. As discussed before, relying on chamfer matching to yield an optimal match for each contour will result in spurious matches due to large intra-class variability and noise. Therefore, we consider multiple placements for each contour within the bounding box and learn to jointly place all contours. Therefore, candidate matches of contours are grouped and a MIL-based procedure [30] is used to find the group with best joint placement. Failing to learn the best joint placement and just selecting appropriate matches for all contours independently significantly degrades the performance—on average we observed a 10% drop on ETHZ shape dataset compared to the MIL-based procedure we propose in this paper.

Let $\Gamma^h_i = (\gamma_1(c_i, \mathcal{E}^h), \gamma_2(c_i, \mathcal{E}^h), \dots)$ denote the matches for c_i in bounding box h. For the m-th match of c_i, we concatenate the chamfer score with the spatial consistency to form a 2-d feature vector ${f_h}^{i,m}$ that will be discussed in Sect. 4. The spatial consistency of a match measures how well the object hypothesis generated from m-th match of c_i agrees with the object bounding box h. Now

we concatenate the 2-d feature representations of all contours to represent the joint placement of all parts. Let ${m_i}^a$ be some match for a contour $c_i \in \mathcal{C}$. Then we obtain a candidate configuration a for the placement of all parts represented by ${f_h}^a = ({f_h}^{1,m_1^a}, {f_h}^{2,m_2^a}, \ldots, {f_h}^{|C|,m_{|C|}^a})$. We start with a contour $c_i \in \mathcal{C}$ and let each of its matches $\gamma_m(c_i, \mathcal{E}^h)$ predict an object hypothesis. Conditioned on this hypothesis, we obtain an object representation ${f_h}^a$ by choosing the spatial maximally consistent match for each of the other contours. By repeating this process for all codebook contours, we obtain a bag of candidate configurations $F_h = \{{f_h}^a\}$.

However, not all the configurations in the bag F_h are meaningful. If for instance some contour c_i is providing a spurious match against background clutter within the bounding box then the resulting feature vectors are also affected. Therefore, we introduce an indicator variable $s(h) \in \{1, \ldots, |F_h|\}$ which selects the most useful candidate configuration for describing the object bounding box. For negative bounding boxes which are obtained by randomly sampling boxes from regions not containing a positive box, all the configurations inside a bag are used as negative examples. Let $Y_h \in \{-1, 1\}$ denote the bag label and let μ be some non-linear function on the co-activation feature vectors ${f_h}^a$. Then we seek the weights $\mathbf{w}$ for each dimension of this transformed feature vector so that the most representative example (identified by $s(h)$) of a positive bag h with $Y_h = 1$ and all the examples of a negative bag h with $Y_h = -1$ have maximum margin separation. Therefore, for a positive bag, the following constraint has to be satisfied for the configuration identified by $s(h)$

$$< \mathbf{w}, \mu({f_h}^{s(h)}) > +b \geq 1 - \xi_h \tag{4}$$

And the following constraint has to be satisfied for all the configurations a in a negative bag.

$$- < \mathbf{w}, \mu({f_h}^a) > -b \geq 1 - \xi_h \tag{5}$$

Thus, we have the following max margin multiple instance learning problem.

$$\begin{aligned} &\min_s \min_{\mathbf{w},b,\xi} \frac{1}{2}\|\mathbf{w}\|^2 + \rho \sum_{\forall h} \xi_h \\ &\text{s.t } \forall h \;\; Y_h = -1 \;\; \wedge - < \mathbf{w}, \mu({f_h}^a) > -b \geq 1 - \xi_h, \forall a \in h \\ &\text{or } \; Y_h = 1 \; \wedge < \mathbf{w}, \mu({f_h}^{s(h)}) > +b \geq 1 - \xi_h \\ &\text{and } \; \xi_h \geq 0 \end{aligned} \tag{6}$$

Equation (6) is expressed in a compact form as

$$\begin{aligned} &\min_s \min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 + \rho\Big(\sum_{Y_h=1} \max(0, 1 - < \mathbf{w}, \mu({f_h}^{s(h)}) > -b) \\ &+ \sum_{Y_h=-1} \max(0, 1 + b + \max_a(< \mathbf{w}, \mu({f_h}^a) >))\Big) \end{aligned} \tag{7}$$

Converting (7) into dual form and utilizing a kernel function $\mathcal{K}$(in our implementation, we use a second degree polynomial kernel) to compute the pairwise distances between original feature vectors $({f_h}^{a_1}, {f_h}^{a_2})$ eliminates the need to explicitly know the function μ. Therefore, equation (7) is optimized in its dual form by iteratively optimizing the indicator variables $s(h)$ and the usual SVM parameters i.e., the support vectors S_{h^a}, their co-efficients α_{h^a} and the offset b. For a positive bag, the dual variable $\alpha_{h^{s(h)}}$ has to satisfy $0 \leq \alpha_{h^{s(h)}} \leq \rho$. For a negative bag, the dual variable has to satisfy $0 \leq \sum_a \alpha_{h^a} \leq \rho$. Thus the effect of each configuration in a negative bag is limited to the box constraint ρ. The minimization starts by choosing $s(h)$ for each bag corresponding to the co-activation feature vector constructed from best match for each of the contours. After the optimization, we obtain the parameters α, S, b and use them in cost function ψ,

$$\psi_{\alpha,S,b}(f) = \sum_{h:Y_h=1} \alpha_{h^{s(h)}} \mathcal{K}(f, S_{h^{s(h)}}) - \sum_{a,h:Y_h=-1} \alpha_{h^a} \mathcal{K}(f, S_{h^a}) + b. \qquad (8)$$

In query images this cost function is applied to find a consistent joint placement f of all codebook contours and the score of ψ is used to rank and classify the resulting hypotheses.

4 Detecting Objects by Describing Their Shape

To detect all instances of an object class in novel query images, their characteristic shape is to be extracted. To capture object shape, codebook contours need to be pieced together properly. Therefore, all these contours need to be jointly matched to a query image so that the grouping of all contours discriminates between objects from the class and all other structure. As a result, objects are segregated from background clutter which in turn improves classification and localization since distracting clutter is suppressed.

4.1 Detecting Meaningful Contours

Let $\mathcal{E}^q$ be the edge map of the query image obtained by using [1]. $\mathcal{E}^{c_i}$ denotes the template edge map created from the codebook contour c_i. $\phi\left(\mathcal{E}_j^q\right)$ denotes the edge gradient orientation at the pixel $\mathcal{E}_j^q \in \mathbb{R}^2$ in the query image.

Given the dictionary $\mathcal{C} = \{c_1, \ldots, c_n\}$ of codebook contours for both objects and non-objects, each contour can be matched to a query image using fast directional chamfer matching [29].

As opposed to the training stage, object scale and aspect ratio are obviously unknown in a query image. Hence, each codebook contour has to be matched at different scales and aspect ratios to a query image. Applying directional chamfer matching [29] yields matches with scores $\gamma_m^{\sigma_m, r_m}(c_i, \mathcal{E}^q)$. The best match has for instance the matching score

$$\gamma_1^{\sigma_1,r_1}(c_i,\mathcal{E}^q) = |\mathcal{E}^{c_i}|^{-1} \sum_{\mathcal{E}_j^{c_i}\in\mathcal{E}^{c_i}} \min_{\mathcal{E}_k^q\in\mathcal{E}^q} \left\{ \left\| \begin{bmatrix} \sigma_1 r_1 & 0 \\ 0 & \sigma_1 \end{bmatrix} \mathcal{E}_j^{c_i} - \mathcal{E}_k^q \right\| + \lambda \left| \phi\left(\begin{bmatrix} \sigma_1 r_1 & 0 \\ 0 & \sigma_1 \end{bmatrix} \mathcal{E}_j^{c_i} \right) - \phi\left(\mathcal{E}_k^q\right) \right| \right\} \tag{9}$$

4.2 Representing Ensembles of Contours

Matching individual codebook contours to query images as done in [4, 5] is prone to yield spurious matches due to intra-class variations of contours. We cannot correctly detect objects by placing each contour individually. Rather, we need to represent an object hypothesis by jointly matching all contours from $\mathcal{C}$ and letting the model learned in Sect. 3.2 propose the right joint placement of contours. For each contour, we obtain multiple matches per scale and aspect ratio, yielding a set of scores $\Gamma = \{\gamma_1^{\sigma_1,r_1}(c_i,\mathcal{E}^q),\ldots,\gamma_k^{\sigma_k,r_k}(c_i,\mathcal{E}^q)\}$ and the corresponding coordinates of the matches $\mathcal{L} = \{l_1^{\sigma_1,r_1}(c_i,\mathcal{E}^q),\ldots,l_k^{\sigma_k,r_k}(c_i,\mathcal{E}^q)\}$. We chose k empirically based on the distribution of chamfer matching scores for the contours over the training images. From this shortlist of matches, we need to find the optimal match for each contour so that the overall configuration is maximally consistent with the joint placement of all contours from the training. As in Sect. 3.1 $\mathbf{s}^{c_i}$ denotes the shift vector of c_i. Then a candidate match $l_m^{\sigma_m,r_m}(c_i,\mathcal{E}^q)$ votes for an object bounding box

$$\mathbf{b}_m^i = \left(\left(l_m^{\sigma_m,r_m}(c_i,\mathcal{E}^q)\right)^\top - \left(\mathbf{s}^{c_i}\right)^\top \begin{bmatrix} \sigma_m r_m & 0 \\ 0 & \sigma_m \end{bmatrix}, \sigma_m, r_m \right). \tag{10}$$

A shortlist $\mathcal{B} = \{\mathbf{b}_1, \mathbf{b}_2, \dots\}$ of potential object hypotheses is created by collecting the hypotheses $\mathbf{b}_m^i$ of all contours c_i in a Hough accumulator [9] and performing the usual non-max suppression. Subsequently, we represent each candidate bounding box $\mathbf{b}_k \in \mathcal{B}$ using the co-activation pattern of all codebook contours. Therefore, we need to measure for each $\mathbf{b}_m^i$ its spatial consistency with an overall object hypothesis $\mathbf{b}_k$ using the standard Pascal VOC criterion.

$$\delta\left(\mathbf{b}_m^i, \mathbf{b}_k\right) := \frac{area(\mathbf{b}_m^i \cap \mathbf{b}_k)}{area(\mathbf{b}_m^i \cup \mathbf{b}_k)}. \tag{11}$$

Let $\hat{m}_{(i,k)}$ denote the m-th match of model contour c_i to query image. For $\mathbf{b}_k$, m-th match has the following directional chamfer and spatial consistency score

$$\mathbf{f}_{m(i,k)}^i := \left(\gamma_m^{\sigma_m,r_m}(c_i,\mathcal{E}^q), \delta\left(\mathbf{b}_m^i, \mathbf{b}_k\right)\right). \tag{12}$$

Thus the overall object hypothesis $\mathbf{b}_k$ can be represented by concatenating all the matching scores to obtain their co-activation pattern.

$$\mathbf{f}_k := \left(\mathbf{f}^1_{m(1,k)}, \ldots, \mathbf{f}^n_{m(n,k)}\right) \in \mathbb{R}^{2n}. \quad (13)$$

We cannot find the correct match $\hat{m}_{(i,k)}$ for each c_i independently. We thus need a joint optimization procedure to find a consistent match from the possible options for each contour. The hypothesis corresponding to the optimal placement of all the contours is then denoted by $\left(\mathbf{f}^1_{\hat{m}_{(1,k)}}, \ldots, \mathbf{f}^n_{\hat{m}_{(n,k)}}\right)$.

4.3 Modeling Shape by Jointly Placing All Object Contours

To jointly find the optimal matches for all the codebook contours, we use the cost function ψ from equation (8). We utilize the second order polynomial kernel function $\mathcal{K}(f_{k_1}, f_{k_1}) = (1 + < f_{k_1}, f_{k_1} >)^2$. The optimal placement $m^*_{i,c}$ for each c_i can be computed using (8) conditioned on the placement of the other codebook contours. Thus, we employ a greedy algorithm to find the optimal placement of each c_i. We initialize the co-activation feature vector by best matches for each contour and then update the placement of contours one at a time. We visit the contours in a random schedule and update the contour placements. We reach rapid convergence for the cost function within 5 sweeps over all contours. Although techniques such as [31] could be potentially used for solving the joint placement problem, speed is an issue with such techniques. We found the sequential greedy approach to converge quickly and to produce competitive results which are described in the experimental section.

5 Experimental Evaluations

We report our experimental evaluations on the standard benchmark datasets for shape-based detection which have been widely used [15, 20, 27, 28, 32, 33], the ETHZ shape dataset and INRIA Horses dataset. These datasets feature significant intra-class variations, scale variations, different lighting conditions and articulations. To evaluate detection performance, we use the PASCAL criterion. Thus the detections are considered correct if the intersection of object hypothesis and the groundtruth over their corresponding union is greater than 50 %. Note that this is a stricter criterion than the 20 % overlap criterion used by [29] to report their performance on ETHZ shape classes. For performing our evaluations, we use the standard protocol described in [33], i.e., use the first half of images in each class for training, and test on the second half of this class as positive images plus all images in other classes as negative images. During the training stage, we only utilize the groundtruth bounding box annotations for the objects and build our shape model from this input.

We use the fast directional chamfer matching code provided by [24] (evaluates 1.05 million hypotheses per image in 0.42 seconds) to obtain matches for each contour. Our codebook contains on the order of 100 contours. We found the performance to be robust with respect to small changes in the number of codebook

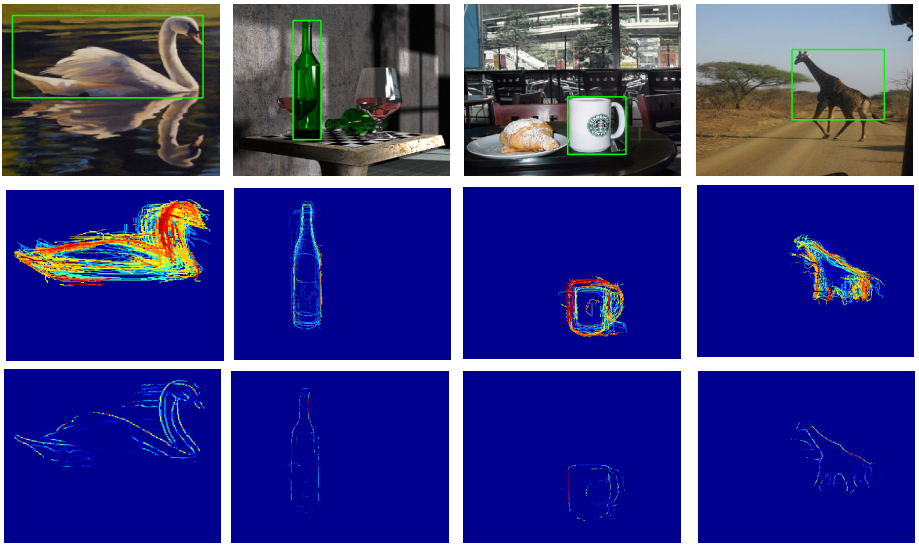

Fig. 2. The first row shows the query images in which the object has been detected (green box). The second row presents the output obtained from our joint placement of all model contours for the infered best bounding box hypothesis shown in column one. The last row shows the backprojection into the query image which explains only relevant object boundaries.

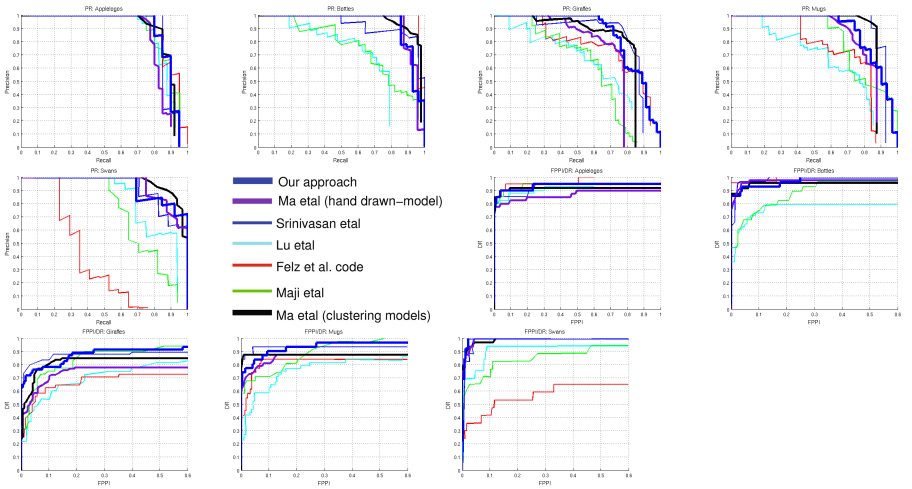

Fig. 3. Performance Comparison in terms of Detection Rate/FPPI and PR Curves for ETHZ Shape Classes

Table 1. Comparison of detection rates for 0.3/0.4 fppi on ETHZ Shape Classes

	Applelogos	Bottles	Giraffes	Mugs	Swans	**Mean**
Ours	95/95	**100/100**	**91.3/91.3**	**96.7/96.7**	**100/100**	**96.5/96.5**
[27]	92/92	97.9/97.9	85.4/85.4	87.5/87.5	**100/100**	92.6/92.6
[28]	95/95	**100/100**	87.2/89.6	93.6/93.6	**100/100**	95.2/95.6
[20]	**100/100**	96/97	86/91	90/91	98/100	94/96
[15]	95/95	92.9/96.4	89.6/89.6	93.6/96.7	88.2/88.2	91.9/93.2
[19]	95/95	**100/100**	72.9/72.9	83.9/83.9	58.8/64.7	82.1/83.3
[16]	95/95	96.3/100	84.7/84.7	**96.7/96.7**	94.1/94.1	93.3/94.1
[32]	93.3/93.3	97/97	79.2/81.9	84.6/86.3	92.6/92.6	89.3/90.5
[34]	95/95	89.3/89.3	70.5/75.4	87.3/90.3	94.1/94.1	87.2/88.8
[33]	77.7/83.2	79.8/81.6	39.9/44.5	75.1/80	63.2/70.5	67.1/72

Table 2. Comparison of Mean Average Precision (AP) on ETHZ Shape dataset

Method	Ours	[28]	[27]	[15]	[19]	[16]
Mean Average Precision	**0.882**	0.872	0.877	0.771	0.712	0.874

Table 3. Comparison of detection rates for 1 fppi on INRIA Horses dataset

Method	Ours	[20]	[17]	[15]
Detection rate	**93.68**	92.4	87.3	85.3

contours. Each test image needs a total processing time (matching all codebook contours and evaluating the model for all candidate hypotheses) on the order of seconds. During the training stage, the multiple instance learning converges within 10 iterations of alternating between indicator variables and dual variables (Sec.3.2). The whole training process is on the order of few hours.

During the testing stage, we search over 7 different scales and 3 different aspect ratios. We evaluate our approach in terms of detection rate over fppi(false positives per image) curves. The detection rates are reported in Tab. 1 at the usual threshold of 0.3/0.4 % fppi and we observe competitive performance compared to the state-of-the-art. The average detection rate is 96.5 % at 0.3 fppi thereby achieving a gain of 1.3 % over the best performing method so far. Our detection rates reach peak value before 0.3 fppi and hence the performance stays same at 0.3/0.4 fppi when comparing with other approaches. We achieve a mean average precision of 0.882 which is improving the performance of state-of-the-art methods summarized in Tab. 2). All in all, we observe a comprehensive gain over the current approaches in terms of various performance measures. Since we jointly explain each object hypothesis, we do not need a separate verification stage and we even outperform a two-stage detection system [15].

In Tab. 1 and Tab. 2, we have included the performance achieved by the latest code release of the popular sliding window based approach [16]. Thus, we are comparing ourselves not only with the state-of-the-art in shape-based methods

but also against the currently best performing recognition system which utilizes many other cues besides shape. Compared to [16], we achieve a gain of 0.8 % in terms of mean average precision. Category-wise, we outperform on 4 categories. In terms of fppi/detection rate, there is an average gain of 3.2 % at 0.3 fppi.

For INRIA Horses dataset, we compare our approach with results reported by other current methods at 1 fppi in Tab. 3. We achieve a detection rate of 93.68 % compared to the current state-of-the-art performance of 92.4 % reported in [20]. [17] and [15] achieve the detection rates of 87.3 and 85.3 % respectively.

6 Conclusion

We have presented an approach that detects objects while, simultaneously, assembling their shape. Meaningful contours are obtained by clustering based on contour co-activation over the training images. The characteristic object shape is represented by learning consistent configurations of all model contours in a maximum margin MIL framework. Rather than placing each contour independently, this approach seeks a joint placement of all contours that discriminates class from non-class structure. In a query image, detection and shape extraction are tackled jointly by optimizing a single cost function that yields optimal configurations of model contours and a classification. In the experimental validation the approach has shown competitive performance on widely used benchmark datasets for shape-based detection. [1]

References

1. Maire, M., Arbelaez, P., Fowlkes, C., Malik, J.: Using contours to detect and localize junctions in natural images. In: CVPR (2008)
2. Carreira, J., Sminchisescu, C.: Constrained Parametric Min-Cuts for Automatic Object Segmentation. In: CVPR (2010)
3. Borenstein, E., Ullman, S.: Combined top-down/bottom-up segmentation. PAMI 30, 2109–2125 (2008)
4. Shotton, J., Blake, A., Cipolla, R.: Multi-scale categorical object rrecognition using contour fragments. PAMI 30, 1270–1281 (2007)
5. Opelt, A., Pinz, A., Zisserman, A.: Incremental learning of object detectors using a visual shape alphabet. In: CVPR (2006)
6. Biederman, I.: Recognition-by-components: A theory of human image understanding. Psychological Review 4, 115–147 (1987)
7. Ommer, B., Buhmann, J.: Learning the compositional nature of visual object categories for recognition. PAMI 32 (2010)
8. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: CVPR, pp. 264–271 (2003)
9. Leibe, B., Leonardis, A., Schiele, B.: Robust object detection with interleaved categorization and segmentation. IJCV 77, 259–289 (2008)
10. Lowe, D.: Object recognition from local scale-invariant features. In: ICCV (1999)

[1] This work has been supported by the Excellence Initiative of the German Federal Government, DFG project number ZUK 49/1.

11. Berg, A.C., Berg, T.L., Malik, J.: Shape matching and object recognition using low distortion correspondence. In: CVPR, pp. 26–33 (2005)
12. Julesz, B.: Textons, the elements of texture perception and their interactions. Nature 29(290), 91–97 (1981)
13. Csurka, G., Dance, C.R., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: ECCV, Workshop Stat. Learn. in Comp. Vis. (2004)
14. Gall, J., Lempitsky, V.: Class-specific hough forests for object detection. In: CVPR (2009)
15. Maji, S., Malik, J.: Object detection using a max-margin hough transform. In: CVPR (2009)
16. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. PAMI 32, 1627–1645 (2010)
17. Yarlagadda, P., Monroy, A., Ommer, B.: Voting by Grouping Dependent Parts. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 197–210. Springer, Heidelberg (2010)
18. Gavrila, D.: A bayesian, exemplar-based approach to hierarchical shape matching. PAMI 29 (2007)
19. Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: CVPR (2008)
20. Toshev, A., Taskar, B., Daniilidis, K.: Object detection via boundary structure segmentation. In: CVPR, pp. 950–957 (2010)
21. Zhu, L., Chen, Y., Lin, C., Yuille, A.: Max-margin learning of hierarchical configural deformable templates (hcdt) for efficient object parsing and pose estimation. IJCV 93, 1–21 (2011)
22. Fidler, S., Leonardis, A.: Towards scalable representations of object categories: Learning a hierarchy of parts. In: CVPR (2007)
23. Ahuja, N., Todorovic, S.: Connected segmentation tree: A joint representation of region layout and hierarchy. In: CVPR (2008)
24. Kokkinos, I., Yuille, A.L.: Hop: Hierarchical object parsing. In: CVPR (2009)
25. Tu, Z., Chen, X., Yuille, A., Zhu, S.: Image parsing: Unifying segmentation, detection, and recognition, vol. 2 (2005)
26. Sala, P., Dickinson, S.: Contour Grouping and Abstraction Using Simple Part Models. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 603–616. Springer, Heidelberg (2010)
27. Ma, T., Latecki, L.: From partial shape matching through local deformation to robust global shape similarity for object detection. In: CVPR (2011)
28. Srinivasan, P., Zhu, Q., Shi, J.: Many-to-one contour matching for describing and discriminating object shape. In: CVPR (2010)
29. Liu, M., Tuzel, O.: A.Veeraraghavan, Chellappa, R.: Fast directional chamfer matching. In: CVPR (2010)
30. Andrews, S., Tsochantaridis, I., Hofmann, T.: Support vector machines for multiple-instance learning. In: NIPS (2003)
31. Narasimhan, M., Bilmes, J.: A submodular-supermodular procedure with applications to discriminative structure learning. In: UAI, pp. 401–412 (2005)
32. Riemenschneider, H., Donoser, M., Bischof, H.: Using Partial Edge Contour Matches for Efficient Object Category Localization. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 29–42. Springer, Heidelberg (2010)
33. Ferrari, V., Jurie, F., Schmid, C.: From images to shape models for object detection. IJCV 87, 284–303 (2010)
34. Ommer, B., Malik, J.: Mulit-scale object detection by clustering lines. In: ICCV (2009)