Pattern Recognition 45 (2012) 1927–1936

Contents lists available at SciVerse ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Contour-based object detection as dominant set computation

Xingwei Yang [a,*], Hairong Liu [b], Longin Jan Latecki [a]

[a] Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122, United States
[b] Embedded Video Lab Block E4, 08-27 4 Engineering Drive 3, National University of Singapore, Singapore 117576, Singapore

ARTICLE INFO

Article history:
Received 21 December 2010
Received in revised form
29 October 2011
Accepted 1 November 2011
Available online 25 November 2011

Keywords:
Object detection
Shape similarity
Dominant sets

ABSTRACT

Contour-based object detection can be formulated as a matching problem between model contour parts and image edge fragments. We propose a novel solution by treating this problem as the problem of finding dominant sets in weighted graphs. The nodes of the graph are pairs composed of model contour parts and image edge fragments, and the weights between nodes are based on shape similarity. Because of high consistency between correct correspondences, the correct matching corresponds to a dominant set of the graph. Consequently, when a dominant set is determined, it provides a selection of correct correspondences. As the proposed method is able to get all the dominant sets, we can detect multiple objects in an image in one pass. Moreover, since our approach is purely based on shape, we also determine an optimal scale of target object without a common enumeration of all possible scales. Both theoretic analysis and extensive experimental evaluation illustrate the benefits of our approach.

1. Introduction

Object detection in cluttered images, with scale and intra-class variations, is one of the most difficult problems in computer vision. Appearance based methods have had remarkable success in recent years [1–5]. However, in many cases, the appearance between intra-class objects varies a lot [6], which makes the appearance features not reliable. Thus, recently we have observed a significant increase in methods that utilize contour shape [7–11]. However, shape based methods also face many challenges, such as pose variance, missing edges, and viewpoint changes. Among these challenges, a critical one seems to be missing contour fragments in the cluttered edge images. The contour fragments may be missing due to occlusion or due to missing edges, since important contours of target objects are often hard or impossible to detect by state-of-the-art edge detectors [12].

Interestingly, our visual system can perform contour grouping, object detection, and recognition, even if only cluttered edge information is provided, e.g., Fig. 1. We can easily perform all these tasks even if the important contour information is missing, and we may not be able to complete it, e.g., we can recognize the giraffes in Fig. 1, but we may not be able to draw or imagine the missing outline of their heads. Thus, we can perform contour grouping, object detection, and recognition while keeping at least part of missing information ambiguous, and we do not attempt to disambiguate all missing information. In other words, we do not attempt to completely reconstruct all contour parts of the object in the image. This fact is one of the key motivations for the proposed inference method.

We formulate object detection as a labeling or matching problem between image segments and model parts. As we discussed, in order to achieve a human like performance in recognizing objects, it requires computing partial assignment between the image segments and model parts, which has been a critical problem for traditional labeling methods [13–15]. To deal with this problem, we propose to transform the matching problem into finding dominant sets in a correspondence graph, in which each vertex represents a pair of image and model segments and the affinities between vertices are obtained by shape similarity, see Fig. 2. With this modification, missing parts of a true object contour in the image do not negatively influence the selection of dominant sets. The concept of dominant sets has been introduced in [16], where also a method for dominant set computation is also proposed. It maximizes the same quadratic function as the spectral methods in [17,18]. The only difference is that dominant set's constraint is L-1 norm compared to their L-2 norm. This minor difference changes dramatically the properties of obtained solutions. Each dominant set is a local solution of a constrained quadratic function, and it is computed by a recursive procedure that depends on the initialization. Recently [19] proposed a novel initialization strategy that is guaranteed to yield all dominant sets under certain assumptions. Although the assumptions in [19] are derived for the application of common visual pattern discovery, they also apply to our application. Different from [16] and from [19], where dominant sets are treated as final solutions, we view each dominant set as a solution hypothesis that is evaluated with global shape similarity. The main reason is that the value of the target quadratic function is based only on local shape similarity, which may be insufficient for object detection. In other words, the dominant set with the highest value of

[*] Corresponding author. Tel.: +1 215 204 8450.
E-mail addresses: xingwei@temple.edu (X. Yang), lhrbss@gmail.com (H. Liu), latecki@temple.edu (L.J. Latecki).

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.11.010

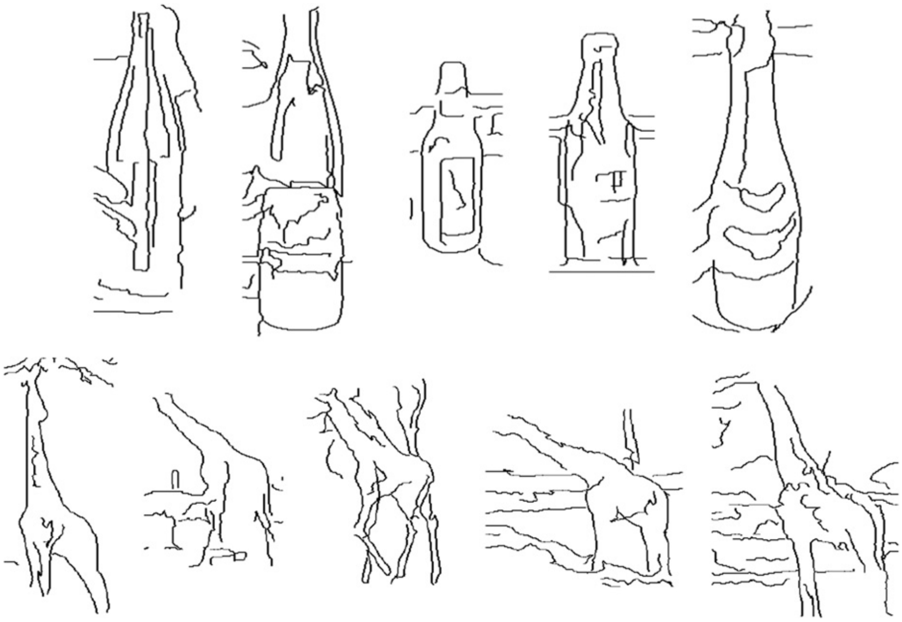

Fig. 1. Parts of the object contours are missing due to missing edges.

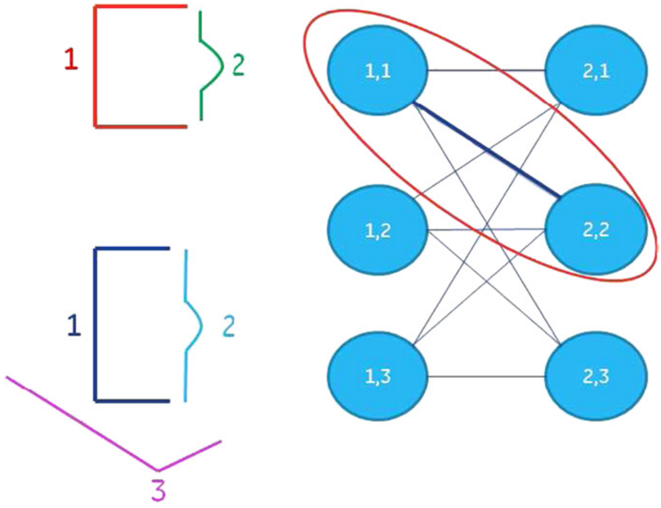

Fig. 2. This is a toy example illustrating the proposed object detection framework as dominant set computation. The object contour model is shown in top left. It is composed of two segments. Edge fragments extracted from a query image are shown below. The graph shows all considered matching pairs of segments. The thickness of edges represent shape similarity of segment configurations. The object detection solution, obtained as the dominant set of the graph, is shown as the red ellipse. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the target quadratic function does not necessarily imply a correct object detection. This is also the reason why we need to consider all dominant sets.

There are at least three key advantages of the proposed method. It is insensitive to noise and outliers, thus, it can detect objects in cluttered images. The fact that we consider all dominant sets provides another main advantage. It allows us to detect multiple objects in one pass, which is also difficult for traditional labeling methods. We do not need traverse all possible solutions, like sliding windows. The multiple detections are obtained from local optimal solutions of a target function, which reduces the complexity a lot. Each object instance is represented by a different dominant set. Moreover, as the proposed method is purely based on shape similarity, we can automatically detect objects in different scales without enumerating the scales, which deals with the problem of resolution. It is an important benefit compared to other methods, such as sliding window [20] and Hough voting [21]. Both methods have to explicitly enumerate various scales in a certain scale range to obtain the best solution. They require a predetermined scale range, which is a hidden parameter not mentioned in most papers.

Example images in Fig. 3 demonstrate the benefit of the proposed method. The white lines are edge segments after edge linking (see Section 3) and the red lines are the detected segments. In these images, parts of the objects are missing due to occlusion, and some of them contain multiple objects like the four apple logos in the top left. Our approach can detect multiple objects at the same time, i.e., in one pass, and allow partial assignment between image segments and model parts, which makes the system robust to missing edges and occlusion. For example, the two mugs are partially occluded, and many bottles are missing some edges. The main steps of the algorithm are introduced in Algorithm 1.

Algorithm 1. The main steps for the algorithm.

```
1: Input: Object Model and Image
2: Use Preprocessing to obtain model parts $S=\{s_1,\ldots,s_m\}$
   grouped into $\mathcal{B}=\{B_k\}_{k=1}^{b}$ part bundles and image segments
   $E=\{e_1,\ldots,e_n\}$;
3: Select $K$ most similar image segment to each model
   segment $s_i$ according the tangent distance;
4: Construct a graph where each vertex is a pair of segments
   $(s_i,e_j)$; The edge weights are determined by the Shape
   Context distance between two pair of segments: $(s_i,s_m)$ and
   $(e_j,e_n)$;
5: Initialize at each vertex with its neighbors; Obtain the
   indicator vector for each initialization; Merge small dense
   subgraph into a large one;
6: Compute evaluation score for each hypothesis (the dense
   subgraph) by the Shape Context distance among selected
   model and image segments.
7: Output: The ordered object detection hypothesis for the
   image.
```

The paper is organized as follow. Related methods are discussed in Section 2. The preprocessing step is described in Section 3. We

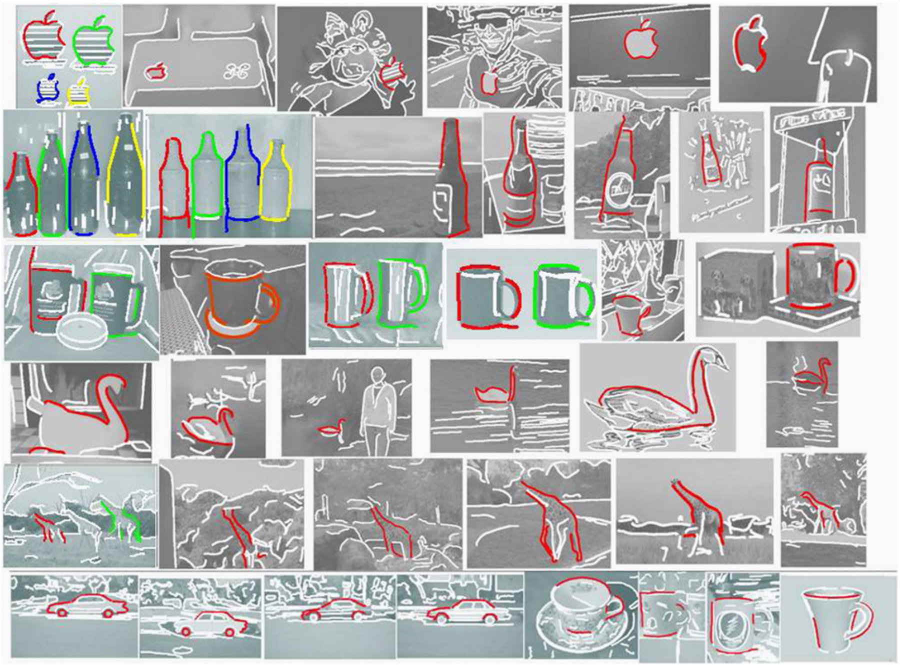

Fig. 3. Example detections on ETHZ dataset and, in the last row, on two Caltech-101 classes: car-side and cups. The detections from different dominant sets are shown in different colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

introduce our approach based on dominant sets in Section 4. The optimization method for finding the dominant sets is described in Section 5. We view the dominant sets as object detection hypotheses, which are then evaluated with global shape similarity in Section 6. In Section 7, we evaluate the performance of the proposed approach on the challenging ETHZ shape dataset [22,7], which features large variations in scale and cluttered background. Besides, we also evaluate our approach on a subset of Caltech 101 dataset [23].

2. Related work

As there exist a lot of papers on object detection and recognition, we only review the most related ones. Ferrari et al. [22] propose to use kAS, the k connected roughly straight contour segments, with Hough voting to detect objects. Later, Ferrari et al. [7] extend their work to learn the contour model from the image. Both of these methods rely on Hough voting strategy to detect objects. Different from them, the proposed method formulates object detection as a dominant set computation problem. Instead of object's contour, Trinh and Kimia [24] use skeleton-based generative shape model with modified dynamic programming to detect objects. Bai et al. [25] also utilize skeleton to constrain the detection process. These algorithms use chamfer matching distance to evaluate the detection hypotheses, which is more tolerant to the deformation than matching of contour segments but more sensitive to the noise. All the above methods require multiple initializations and they enumerate all possible object sizes (scales) to get the optimal results. Different from them, the proposed method is able to detect multiple objects at different scales in one pass without enumerating scales. The approach by Ravishankar et al. [26] is also scale invariant, but it is very different from our method. [26] utilizes a multi-stage contour based detection with dynamic programming.

Similar to the proposed method, several methods have formulated contour based object detection as a matching process between model and image segments. Zhu et al. [8] utilize Shape Context [27] to evaluate the distance between model and image segments. They formulate the shape matching of contours as a set–set matching problem and solve it by linear programming, which is fundamentally different from our approach. Though they have reduced the computational complexity by relaxing the problem into linear programming, their algorithm is still less efficient than ours. Similar to our method, Lu et al. [11] formulate object detection as a segment correspondence problem. However, their inference framework is very different, where they utilize particle filter to solve the label assignment problem. Due to the limitation of their inference, they cannot detect multiple objects.

To refine detection candidates, many algorithms conduct a verification stage to obtain final results. To solve the problem of scales in Hough voting, Ommer and Malik [28] propose a weighted, pairwise clustering of voting lines to obtain globally consistent hypotheses. Then, a verification stage is used to re-rank the hypotheses. Maji and Malik [29] integrate Hough transform based features of codebooks into kernel classifiers to detect objects. Unlike them, we use a purely shape based method and do not utilize any classifiers like SVM to rank the hypotheses.

Besides detecting objects with shape information, various techniques have been explored. Gu et al. [30] utilize region segmentation to detect target objects. Recent approach by [31] utilizes region information in addition to contours. As expected, the region information improves the detection results. Felzenszwalb et al. [32] detect objects by using part based models, which increases the discriminative ability a lot. Zhang et al. [33] localize the objects with proper shapes that describes the objects, instead of just bounding box. Their algorithm is based on the widely used bag of visual words technique, which requires a supervised learning process to extract descriptive features. It is based on supervised learning with properly selected texture features. In contrast, we only use shape information to detect objects and we do not take advantage of classifiers.

3. Preprocessing

As we formulate the object detection as a correspondence problem between image segments and model segments, we need to construct image segments from image edge maps as well as define shape models composed of contour segments. We utilize shape similarity of these segments to perform object detection. Given an image I and the edge map, we use an open source edge linking method provided by Kovesi [34] to group edge pixels into edge fragments. If a junction point exists on the edge fragment, the corresponding edge fragments are split at the junction point. We obtain a set of image edge segments $E=\{e_1,\ldots,e_n\}$ for the image I. An example is shown in Fig. 4(a), where each edge segment is shown in a different color.

The model segments $S=\{s_1,\ldots,s_m\}$ are manually designed so that they represent meaningful contour parts. As junction points normally exist at high curvature points in the edge maps, we also decompose the model template at high curvature points. Moreover, since the image segments are noisy and some part of object boundary may be missing, we need to add shorter model segments in addition to longer ones. Then, the segments are grouped into different part bundles $\mathcal{B}=\{B_k\}_{k=1}^{b}$, where each part bundle represents the same visual part of the modeled contour. The part selection was tuned on example images different from test images. The first principle of model design was that the contour obtained by selecting one part from each bundle still resembles the original model contour. The second principle was that contour should be broken into parts at high curvature points, since these are the points where edge fragments in the image are likely to be broken.

An example is shown in Fig. 4(b).

4. Problem formulation

With preprocessing introduced in Section 3, we can obtain a set of image segments $E=\{e_1,\ldots,e_n\}$ for the image I and a set of model segments $S=\{s_1,\ldots,s_m\}$ for a model contour template. We formulate the object detection as a labeling problem, labeling the model segments with the image segments. Our goal is to find the segment correspondence so that the image segments corresponding to the model segments maximize the global shape similarity of all selected model segments to all selected image segment.

To reach this goal, we first build a graph G whose nodes are $\{c_1,\ldots,c_m\}$, where $c_i=(s_i,e_{i'})\in S\times E$, and $i=1,\ldots,m$. A tangent distance $TD(s_i,e_{i'})$ between two segments s_i and $e_{i'}$ is computed. We first build the mapping between the sample points p_k. Using sequences of tangent directions along s_i and $e_{i'}$ as their shape descriptors, we perform sequence matching with the Smith–Waterman algorithm [35], see Fig. 5. Since we sample all the segments to the same number of sample points K_s, TD is scale invariant. Once the correspondences $C:\{1,\ldots,K_s\}\rightarrow\{1',\ldots,K_s'\}$ are constructed, the tangent distance for the two segments is defined as the sum of the absolute values of tangent differences among corresponding points

$$TD(s_i,e_{i'})=\sum_{k=1}^{K_s} Dist(T(p_k),T(p'_{C(k)}), \tag{1}$$

where $p_k\in s_i$, $p'_{C(k)}\in e_{i'}$ and $T(\cdot)$ represents the tangent direction at a point as tangent angle in the range $[0,2\pi)$. $Dist$ is defined as $Dist(t_i,t_j)=(t_i-t_j)\ modulo\ 2\pi$, where t_i, t_j are two tangent angles.

For each model segment s_i, we use TD to find K most similar image segments. Hence graph G has $M=m\times K$ nodes and each node represents a correspondence c_k for $k=1,\ldots,M$. The weight of an edge connecting nodes i and j is defined as

$$w_{ij}=\mathcal{N}(SC(s_i\cup s_j,e_{i'}\cup e_{j'})), \tag{2}$$

where $c_i=(s_i,e_{i'})$ and $c_j=(s_j,e_{j'})$ are two correspondences, $\mathcal{N}$ is a Gaussian, and SC is the Shape Context [27] distance. The mean of Gaussian $\mathcal{N}$ is defined as 0 and the standard deviation is defined as a quarter of the average distance between all pairs of correspondences.

Hence w_{ij} represents shape similarity between shape constructed of two model segments $s_i\cup s_j$ and two image edge segments $e_{i'}\cup e_{j'}$. The adoption of Shape Context has several advantages. First, it is a descriptive shape similarity method and it can be easily used for discrete points sets. Second, SC performs automatic scale normalization. Consequently, w_{ij} is scale invariant.

However, not all correspondences in graph G are compatible. For example, two edge segments that are far away from each other in a given image cannot both belong to the contour of a

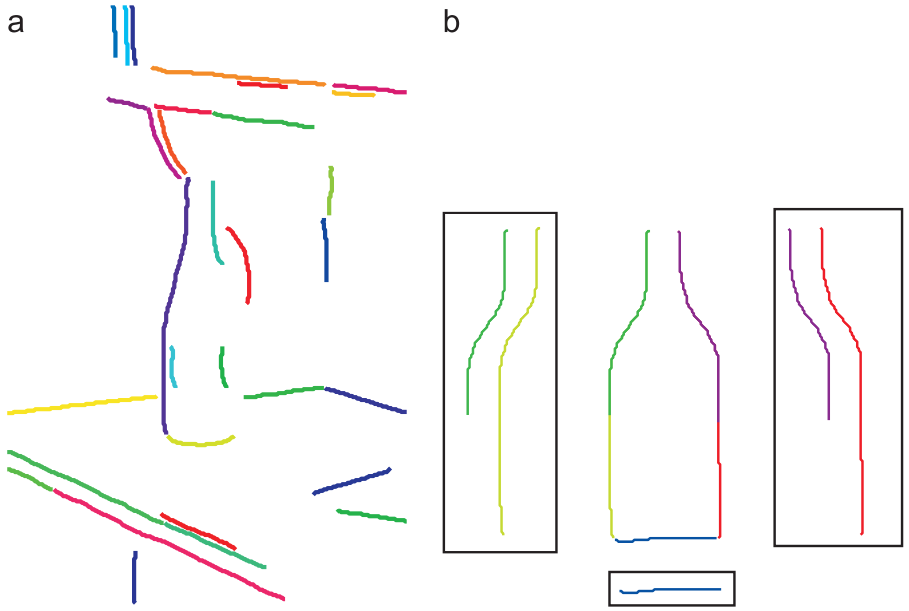

Fig. 4. (a) The edge segments in one image. Different colors represent different segments. (b) Model segments for category bottle, which is shown in the middle. Each segment is shown in a different color and the segments in one box form a part bundle. Thus, there are three part bundles for the bottle. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

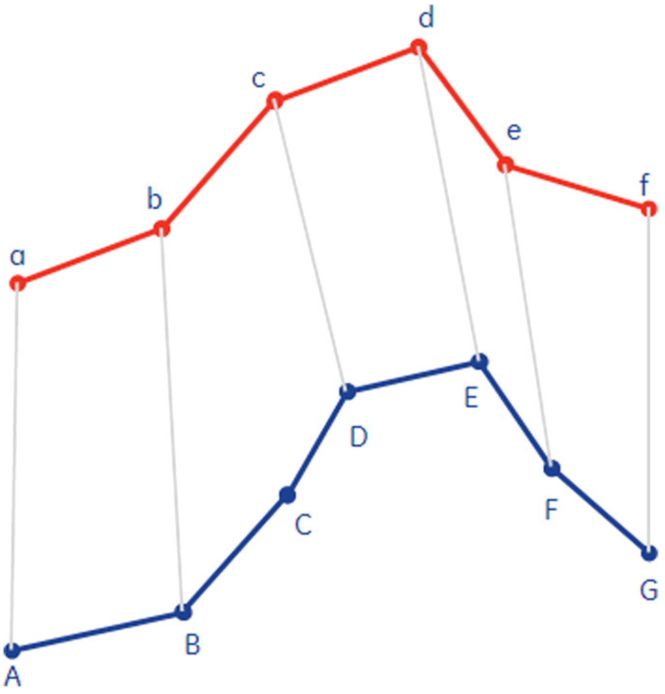

Fig. 5. This is a toy matching example of two segments. Each point is represented by its tangent direction. The correspondences found are (a,A), (b,B), (c,D), (d,E), (e,F) and (f,G).

target object whose diameter is smaller than their distance. Therefore, we will define now a binary relation that allows us to efficiently remove such correspondences from G. We observe that when computing the shape distance $TD(s_i,e_{i'})$, we also obtain the correspondence of sample points of s_i to sample points of $e_{i'}$. It allows us to determine the scale factor so that the model bounding box can be properly re-scaled. Then the re-scaled bounding box is placed on the image; since we know the position of the model bounding box relative to segment s_i, the position of the bounding box in the image is determined by the position of $e_{i'}$.

Let us denote two re-scaled and relocated model bounding boxes in the image with $bbx(i)$ and $bbx(j)$, which are based on $c_i=(s_i,e_{i'})$ and $c_j=(s_j,e_{j'})$, respectively. Of course, if both correspondences are correct the two bounding boxes in the image should coincide. In the left image of Fig. 6, the color segments are the image segments being considered and the numbers show the indices of the corresponding model segments, which are 1 and 3. The two estimated bounding boxes are shown in red and green. Although the bounding box estimation is not perfect, it can roughly determine that the two image edge segments can belong to the same contour. On the other hand, the right image shows a wrong correspondence that leads to two disjoint bounding boxes. Therefore, if the area of the intersection of both bounding boxes in the image is small, then $e_{i'}$ and $e_{j'}$ cannot be both parts of the contour of the target object. To capture this property, we define a binary relation

$$R_I(i,j)=\begin{cases}1, & \frac{area(bbx(i)\cap bbx(j))}{area(bbx(i)\cup bbx(j))}>C,\\ 0, & otherwise,\end{cases} \tag{3}$$

where C is the area intersection threshold that is set to 0.1 in all our experiments.

We also define another binary relation that relates model segments of two correspondences. Since the shape constructed by two segments $s_i \cup s_j$ that belong to the same part bundle B_k for $k=1,\ldots,b$ is not particularly informative, because they represent the same model part, we do not allow correspondences that involve such model segments. We define $R_M(i,j)=0$ if $s_i,s_j \in B_k$ for some $k=1,\ldots,b$ and $R_M(i,j)=1$ otherwise.

The weighted adjacency matrix A of graph G is an $M\times M$ matrix defined as

$$A_{i,j}=\begin{cases}0, & i=j \text{ or } R_I(i,j)=0 \text{ or } R_M(i,j)=0,\\ w_{ij}, & otherwise.\end{cases} \tag{4}$$

The binary relations R_I and R_M help us to make the graph G sparse, which significantly reduces the computation cost. It is obvious that A is symmetric and nonnegative, since this is the case for w_{ij}, R_I and R_M.

We are interested in finding subgraphs H of G that are local maxima of the average affinity score S_a defined as

$$S_a(H)=\frac{1}{|H|^2}\sum_{i\in H,j\in H}A_{ij}=\vec{x}^TA\vec{x}, \tag{5}$$

where $|H|$ is the number of nodes of H and $\vec{x}$ is a column vector such that $x_i=1/|H|$ if $i\in H$ and $x_i=0$ otherwise for $i=1,\ldots,M$.

For unweighted graphs, the Motzkin–Straus theorem [36] has established a connection between the maximal cliques and the local maximizers of the quadratic function

$$\text{maximize} \quad f(\vec{x})=\vec{x}^TA\,\vec{x} \quad \text{subject to } \vec{x}\in\Delta, \tag{6}$$

where $\Delta=\{\vec{x}\in\mathbb{R}^m:\vec{x}\geq 0 \text{ and} |\vec{x}|_1=1\}$ is the standard simplex in $\mathbb{R}^m$. Eq. (6) means that a subgraph H of G is a maximal clique if and only if its characteristic vector $\vec{x}^H$ is a local maximizer of this equation, where $x_i^H=1/|H|$ if $i\in H$, and $x_i^H=0$ otherwise. Recently, Pavan and Pelillo [16] generalized the Motzkin–Straus theorem to weighted graphs. They also introduced the notation of dominant sets of vertices as a generalization to weighted graphs of the concept of a maximal cliques in unweighted graphs. In unweighted graphs dominant sets are equivalent to (strictly) maximal cliques. They showed that each (strict local) solution of the quadratic program Eq. (6) determines a **dominant set**, which we take as a definition of the dominant set in this paper.

As has been observed in [16,19], Eq. (6) maximizes the same quadratic function as the spectral methods in [17,18]. The only difference is the constraints on $\vec{x}$: the spectral methods require $|\vec{x}|_2=1$ while Eq. (6) requires $|\vec{x}|_1=1$. This minor difference changes dramatically the properties of obtained solutions. Instead of partitioning all data, as is the case for spectral methods, Eq. (6) only selects highly correlated data and ignores outliers. Consequently, the proposed object detection system can automatically select contours of the target object in an edge image and at the same time ignore the vast majority of the background segments.

Many other graphs can also be utilized to recognize objects, such as Delenay graphs, LG graphs [37]. However, they are sensitive to outliers and noises in the graph, which is exactly the main difficulty for object detection. Compared to them, the proposed method is robust to anomalies, which makes it proper for detecting objects in clutters.

5. Optimization

The main challenge we face now is to determine all local maxima of the quadratic program Eq. (6). Following [16], once an initialization $\vec{x}(1)$ is given at discrete time step 1, the discrete replicator equation [38] can be used to obtain a local solution $\vec{x}^*$

$$\vec{x}_i(t+1)=\vec{x}_i(t)\frac{(A\,\vec{x}(t))_i}{\vec{x}(t)^TA\,\vec{x}(t)}, \tag{7}$$

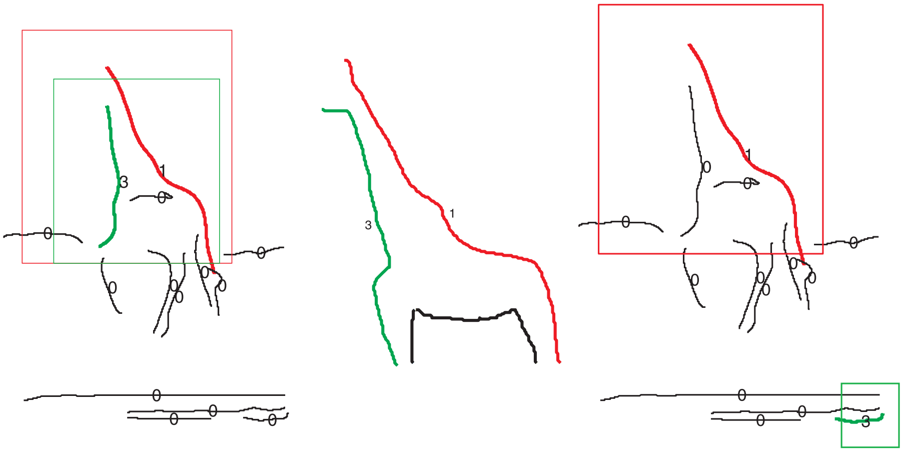

Fig. 6. Left: the estimated bounding boxes of two correct correspondences. Right: the estimated bounding boxes of two wrong correspondences. Middle: the shape model with corresponding segments marked in colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

for $i=1,\dots,M$ indexing the coordinates of vector $\vec{x}$. As is proven in [16], each strict local solution of (Eq. (6)) determines a dominant set.

A key question for our approach is how to enumerate the initialization vectors $\vec{x}(1)$ so that we can obtain all local maxima $\{x^*\}$. To solve this problem, Pavan and Pelillo [16] propose to detect dominant sets iteratively, i.e., after finding a dominant set, they remove its vertices from the graph G, and then rerun the algorithm on the remaining vertices. However, their method may miss some local maxima due to the fact that some dominant sets have nonempty intersection.

Recently [19] proposed a novel initialization strategy, which we follow in our approach. They suggest initializing $\vec{x}(1)$ in the neighborhood $N(v)\bigcup\{v\}$ of every vertex $v\in G$, where $N(v)$, the neighborhood of v is determined by thresholding the row v of matrix A. This strategy is based on the assumption that each dominant set that contains v must be a subset of $N(v)\bigcup\{v\}$. This assumption is satisfied in our setting for dominant sets that correctly determine a target shape composed of edge segments in the image, since all correspondences that extend the correspondence v to form the target shape have relatively large affinity with v. Consequently, this initialization strategy does not eliminate any correct solution in our setting.

However, this initialization strategy produces many duplicate solutions. Therefore [19], proposed merging two dominant sets if their indicator vectors $\vec{x}^*$ and $\vec{y}^*$ have large correlation defined as $(\vec{x}^*)^T y^*$. This reflects the fact that although different dominant sets may have nonempty intersection, their overlap is usually small. This merging strategy significantly reduces the set of solution hypotheses that we need to examine with global shape similarity as described in the next section.

6. Final evaluation

All different dominant sets obtained as solutions to Eq. (6) are potential object detection hypotheses in our system. Our final step is to use global shape similarity to evaluate these hypotheses. Although computing the global shape similarity is computationally expensive, it is tractable in our application. Usually the graph of correspondences G may have several hundred vertices, but we obtain less than 20 different dominant sets as solutions to Eq. (6).

Since each coordinate x_i^* of $\vec{x}^*$ represents the probability that correspondence i has been selected, we simply select the correspondences with probability bigger than 0 as the elements of the dominant set L determined by $\vec{x}^*$. We obtain $L=\{c_{i_1},\dots,c_{i_l}\}$ for some $l\ll M$, where $c_{i_k}=(s_{i_k},e_{i'_k})$. The global shape distance is computed with shape context as

$$SC(L)=SC\left(\bigcup_{k=1}^{l} s_{i_k},\bigcup_{k=1}^{l} e_{i'_k}\right). \qquad (8)$$

Thus, we simply compare the shape formed by combining all selected image edge segments $\bigcup_{k=1}^{l} e_{i'_k}$ to the corresponding model segments $\bigcup_{k=1}^{l} s_{i_k}$.

Although in Eq. (4), we have removed the connection between the model segments from the same part bundle, L can still contain different model segments from the same part bundle. Since each part bundle can only provide at most one part for one detection, the multiple model segments from the same part bundle generate multiple candidate detections at the same position. In other words, one candidate detection is composed of one and only one segment from each bundle existing in the dominant set. This exclusion property of the segments in the same bundle leads to several candidate detections in one dominant set. The final evaluation Eq. (8) is conducted over all the candidates and the one with the best score is selected as the detection represented by the dominant set. Moreover, we only consider dominant sets L that contain correspondences from a certain minimal number of part bundles, which is set to 3 in all our experimental results. The global shape distance in Eq. (8) is used to rank the object detection hypotheses. We stress that our method is purely based on shape similarity without using any training or classifier.

7. Experimental results

To evaluate our approach, we choose the challenging ETHZ Shape Dataset [7,22] containing five different categories with 255 images in total. Each image contains one or more instances with significant background clutter. All categories have significant scale difference and intra-class variation. Similar to Zhu et al. [8] and Lu et al. [11], we use a single manually constructed, contour model for each shape class. The detection performance is measured based on the standard PASCAL VOC criterion [39].

Furthermore, we also selected two classes from Caltech-101 dataset [23] to evaluate our approach: cups and car-sides. They contain substantial intra-class variations and missing edge segments. Similar to [22] an equal number of negative images is selected from the Caltech-101 background set. Then, the test set for each class consists of all positive images and the equal number of negative images. The contour model for cups is the same as the model for ETHZ mugs, and we manually created a contour model for car-sides.

The time for processing images varies due to the different number of segments in the image. On average, in our experiments, it takes about 30–40 s per image. Moreover, the current method is sensitive to the viewpoint change, which is a common problem for shape based object detection methods. One main reason is that shapes are not always meaningful in some viewpoints, such that the shapes of top viewed images of bicycle are just a couple of straight lines. Some works have been done to solve this problem, such as [40]. They have made some progress, but still not stable for large viewpoint changes. This will be our future task.

7.1. ETHZ dataset

In Fig. 3, some example detections on ETHZ dataset are shown. These examples demonstrate the proposed method can handle multiple detections and scale variation in one pass. Moreover, our method is robust to missing segments, even if a whole meaningful object part is missing, which cannot be solved by relaxation labeling and many other methods. Moreover, some false positive detection results are given in Fig. 7.

We also demonstrate the benefit of our algorithm by comparing to other methods. A large number of methods have been tested on ETHZ dataset [30,29,11,7,8,22,28]. It is difficult to compare all of them, thus, we only compare to some shape based methods. We first compare to [22,7,28] by plotting the detection rate (DR) against false positive per image (FPPI), see Fig. 8. Besides the curves, the detection rates at 0.3 and 0.4 FPPI are also shown in Table 1. We stress that only the method by [28] and the proposed method are truly scale independent. The other two methods [22,7] enumerate scales in a certain scale range.

Table 1
Detection rates at 0.3/0.4 FPPI for ETHZ dataset. The best results are highlighted in bold.

Category	Clustering lines [28]	kAS [22]	Full system [7]	Our method
Apples	**95.0**/**95.0**	50.0/60.0	77.7/83.2	80.0/80.0
Bottles	89.3/89.3	**92.9**/92.9	79.8/81.6	**92.9**/**95.9**
Giraffes	70.5/75.4	49.0/51.1	39.9/44.5	**76.92**/**79.21**
Mugs	**87.3**/**90.3**	67.8/77.4	75.1/80.0	83.3/84.85
Swans	**94.1**/**94.1**	47.1/52.4	63.2/70.5	90.9/**94.1**
Average	**87.2**/**88.8**	61.4/66.8	67.2/72.0	84.8/86.79

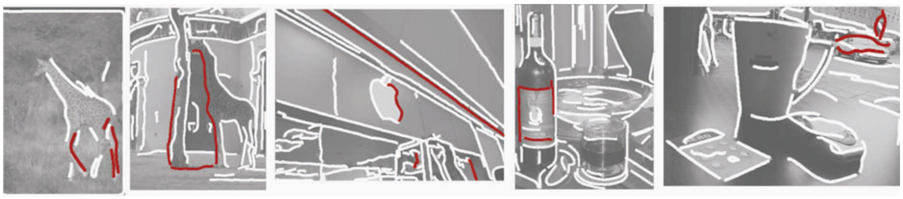

Fig. 7. Example of false positive detections on ETHZ dataset. From left to right, the model is apple, bottle, giraffe, mugs and swan.

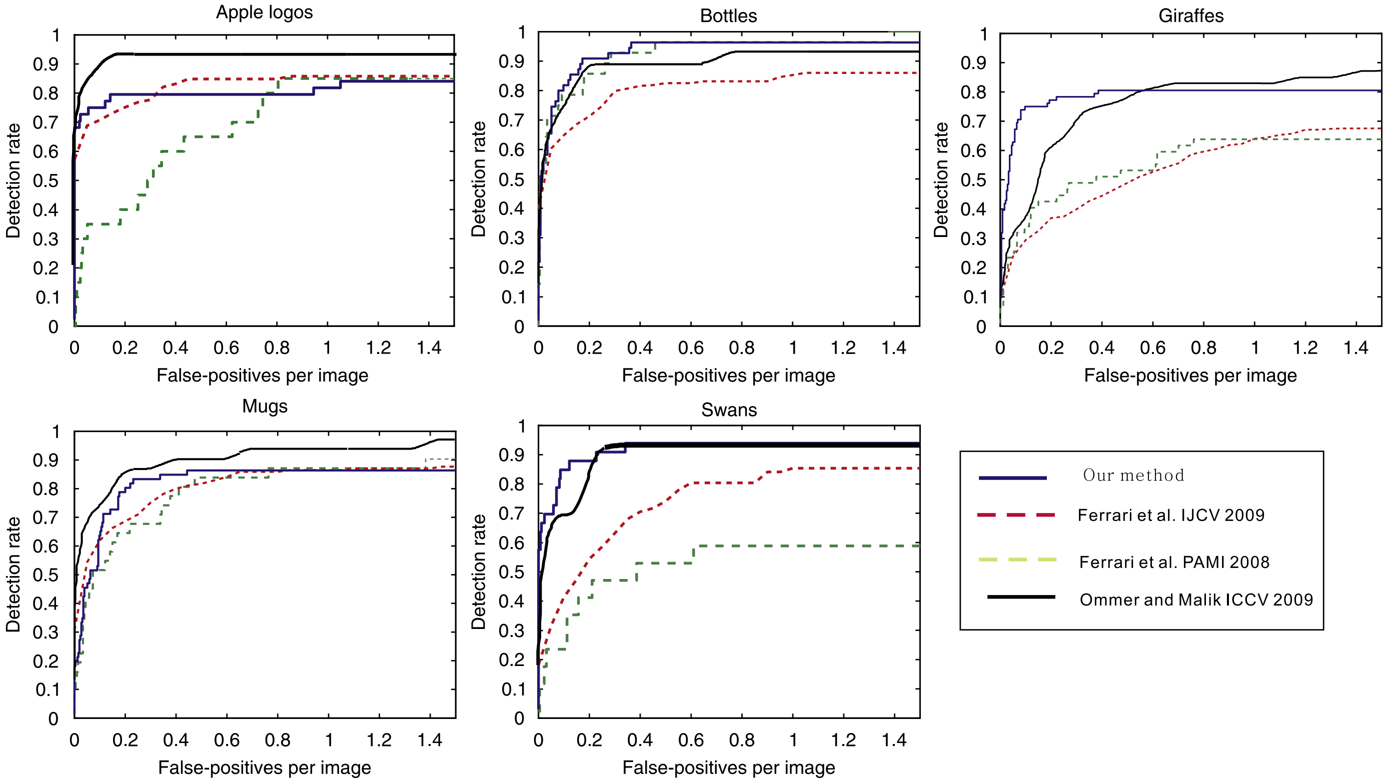

Fig. 8. DR/FPPI curves on ETHZ dataset with comparison to methods in [28,22,7].

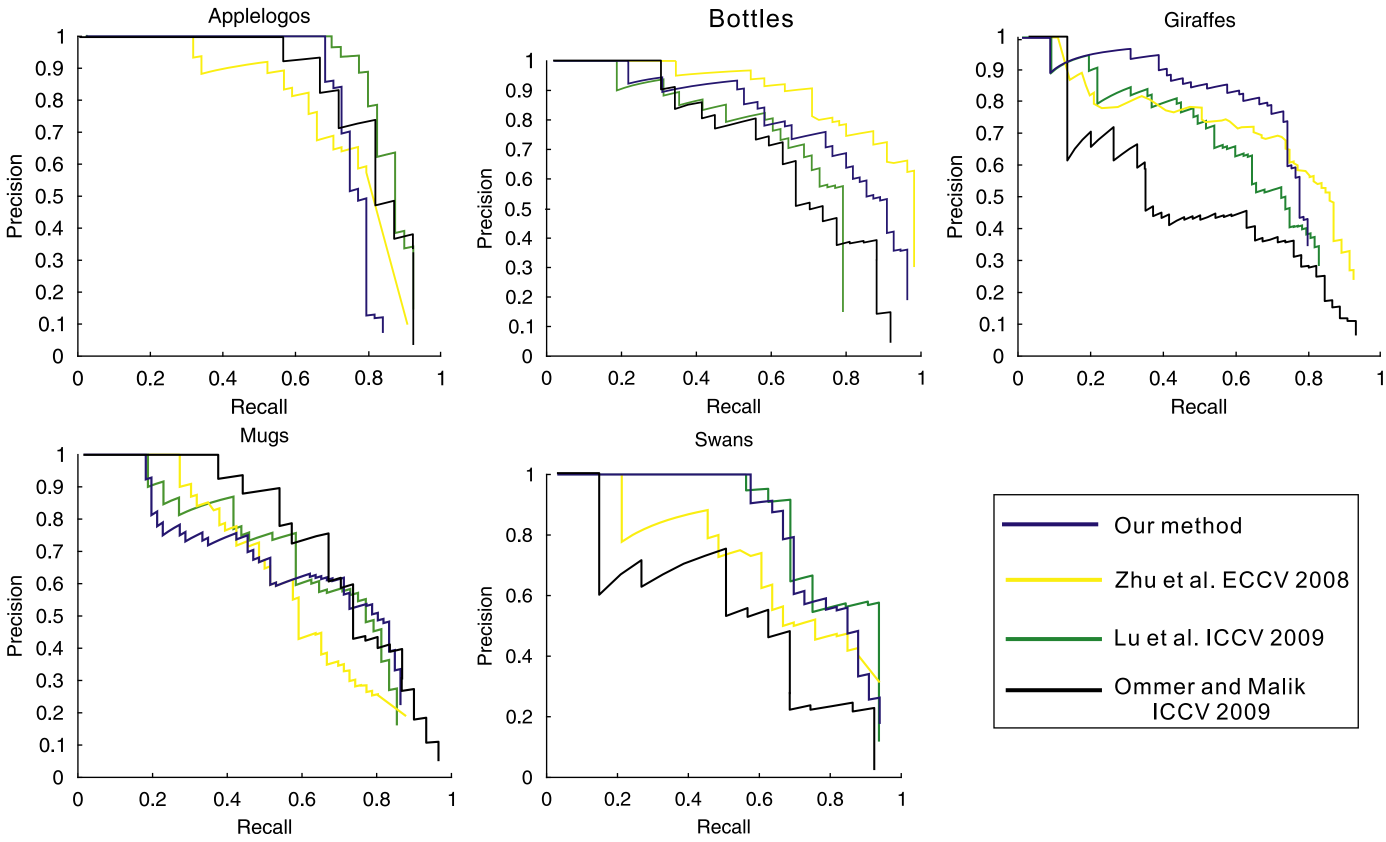

Fig. 9. Precision/recall curves on ETHZ dataset with comparison to [28,8,11].

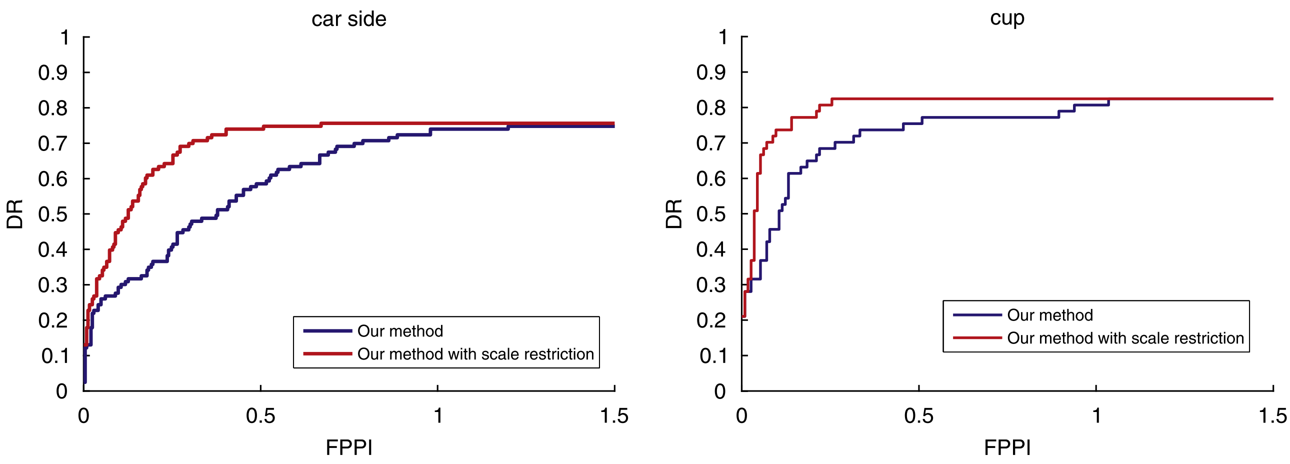

Fig. 10. DR/FPPI curves on class car-side and cups of Caltech-101 dataset.

Ferrari et al. [22,7] train their detector for each category on half of the positive examples on that class. In [22], they also use the negative images for training. In comparison to [7], it turns out that our algorithm can obtain better results on the whole ETHZ dataset except the class apple logos, where we perform equally well. Our method improves the average detection rate by 17.7% and 14.9% at 0.3 and 0.4 FPPI, respectively. Similarly, we are comparable to [22] on the class bottle and better on all the other classes. The average detection rate has increased by 23.5% and 20.1% at 0.3 and 0.4 FPPI, respectively.

Furthermore, we also compared to the recent work by [28], which also uses half of the positive examples as training. Our method performs better on classes giraffes and bottles and is comparable on class swans. However, we perform worse on mugs and apples. The average detection rates at 0.3 and 0.4 FPPI are about 2% lower compared to [28].

We stress that our algorithm is purely shape based and we do not have any post-processing phase to refine the results. In [28], the SVM classifier is run in sliding window mode over a grid of locations around each initial detection, which boosts the results a lot. Without the help of the verification step, it performs much worse than our method. For example, the detection rates at $FPPI=1.0$ for all the fives classes are 80.0% for apple, 89.3% for bottle, 80.9% for giraffes, 74.2% for mugs and 68.6% for swans.

Then, the average over the whole dataset is 78.6%. All of them are much lower than our algorithm.

To make the comparison to [28] complete, we also compare the precision/recall curves in Fig. 9. Unlike the previous comparison in DR/FPPI, the actual values for precision/recall are not reported in [28]. Thus, we can only compare the curves visually. Our results on swans and bottles are better than [28] and the precision of giraffes is better with a little lower recall value. The precision of apple logos is comparable, but the recall is worse. Our results of mugs are worse than [28]. We also compared to methods in [8,11] using precision and recall. Similar to [28], both of them do not report actual values and we can only compare by visual estimation. With comparison to [8], it is apparent that our method performs better on swans and equally well on apple logos, mugs and giraffes, but it is outperformed on bottles. Compared to [11], we perform better on giraffes and bottles and we are comparable on classes mugs and swans. The only class we are worse is apple logos.

7.2. Subset of Caltech-101

We also test on two classes of Caltech-101 dataset: cups and car-sides. Since there are no given edge maps like for ETHZ [22], we use Canny Edge detector to obtain edges. Four examples for each class are shown in the last row in Fig. 3.

The blue DR/FPPI curves in Fig. 10 illustrate the performance of our algorithm. We obtain detection rate 76.5% and 51.2% at 0.4 FPPI on class cups and car-sides, respectively. The high false positive rates for class car-side are due to the low resolution of the images, which makes edge maps not reliable. Consequently, the image segments may be messy making the obtained edge segments different from the model, so that the final global shape similarity may not be able to distinguish the false positives. However, the reasonable detection rate, about 75%, shows that even when the edges are not reliable, the proposed method can still accurately localize the objects.

In order to show the influence of predetermined scale range, we use the scale range as a constraint for our detection results on the two classes. If the scale of a detection hypothesis (which is automatically determined by shape similarly) is not within the scale range, we discard the hypothesis. As shown by red curves in Fig. 10, it is obvious that the restriction of scale range improves the performance. It is mainly due to removing the false positives. With the scale restriction, the detection rate at 0.4 FPPI on class cups and car-sides increases to 82.3% and 73.2%, respectively. The increase in detection rate on car-sides is by 22%. We observe that the detection rate of 82.3% on cups is better than the 78.6% reported in [22]. This is particularly impressive, since our contour model for cups is the ETHZ mug model without any modification, while [22] trained a cup classifier on half of cup images.

8. Conclusion

In this paper, we presented a simple yet effective purely shape based approach for object detection. We formulate object detection as a matching problem between image and model segments. To solve the problem of missing segments, we transform the matching problem into finding dominant sets in the correspondence graph. With this transformation, the algorithm is also robust to outlier and noise (background clutter) in the image. Besides, the proposed method can detect multiple objects at multiple scales in one pass, which reduces the complexity a lot compared to standard sliding window and Hough voting approaches. The next step of our work is learning the shape bundle model from training images without any human supervision.

Acknowledgments

The work was supported by the DOE Award 71498-001-09, the AFOSR Grant FA9550-09-1-0207, and by the NSF under Grants IIS-0812118, BCS-0924164, OIA-1027897.

References

[1] B. Leibe, A. Leonardis, B. Schiele, Combined object categorization and segmentation with an implicit shape model, in: Proceedings of the Workshop on Statistical Learning in Computer Vision, Prague, Czech Republic, 2004.
[2] J. Gall, V. Lempitsky, Class-specific hough forests for object detection, in: CVPR, 2009.
[3] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in: CVPR, 2006.
[4] J. Shotton, J.M. Winn, C. Rother, A. Criminisi, Textonboost: joint appearance, shape and context modeling for multi-class object recognition and segmentation, in: ECCV, 2006.
[5] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: CVPR, 2001.
[6] A.P. Andreas Opelt, A. Zisserman, A boundary-fragment-model for object detection, in: ECCV, 2006.
[7] V. Ferrari, F. Jurie, C. Schmid, From images to shape models for object detection, International Journal of Computer Vision 87 (3) 284–303 doi:10.1007/s11263-009-0270-9.
[8] Q. Zhu, L. Wang, Y. Wu, J. Shi, Contour context selection for object detection: a set-to-set contour matching approach, in: ECCV, 2008.
[9] M. Stark, M. Goesele, B. Schiele, A shape-based object class model for knowledge transfer, in: ICCV, 2009.
[10] T. Jiang, F. Jurie, C. Schmid, Learning shape prior models for object matching, in: CVPR, 2009.
[11] C. Lu, L.J. Latecki, N. Adluru, X. Yang, H. Ling, Shape guided contour grouping with particle filters, in: ICCV, 2009.
[12] D. Martin, C. Fowlkes, J. Malik, Learning to detect natural image boundaries using local brightness, color, and texture cues, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (5) (2004) 530–549.
[13] A. Rosenfeld, R. Hummel, S. Zucker, Scene labeling by relaxation operations, IEEE Transactions on Systems, Man and Cybernetics 6 (1976) 420–433.
[14] T. Caetano, T. Caelli, D. Schuurmans, D. Barone, Graphical models and point pattern matching, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (10) (2006) 1646–1663.
[15] A. Berg, T. Berg, J. Malik, Shape matching and object recognition using low distortion correspondences, in: CVPR, 2005.
[16] M. Pavan, M. Pelillo, Dominant sets and pairwise clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (1) (2007) 167–172.
[17] S. Sarkar, K. Boyer, Quantitative measures of change based on feature organization: Eigenvalues and eigenvectors, Computer Vision and Image Understanding 71 (1998) 110–136.
[18] M. Leordeanu, M. Hebert, A spectral technique for correspondence problems using pairwise constraints, in: ICCV, 2005.
[19] H. Liu, S. Yan, Common visual pattern discovery via spatially coherent correspondences, in: CVPR, 2010.
[20] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: CVPR, 2005.
[21] P. Hough, Method and means for recognizing complex patterns, in: U.S. Patent 3069654, 1962.
[22] V. Ferrari, L. Fevrier, F. Jurie, C. Schmid, From images to shape models for object detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (1) (2008) 36–51.
[23] L. Fei-Fei, R. Fergus, P. Perona, One-shot learning of edges and object boundaries, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (4) (2006) 594–611.
[24] B.B.K.NhonH. Trinh, Category-specific object recognition and segmentation using a skeletal shape model, in: BMVC, 2009.
[25] X. Bai, X. Wang, L.J. Latecki, Z. Tu, Active skeleton for non-rigid object detection, in: ICCV, 2009.
[26] S. Ravishankar, A. Jain, A. Mittal, Multi-stage contour based detection of deformable objects, in: ECCV, 2008.
[27] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2002) 509–522.
[28] B. Ommer, J. Malik, Multi-scale object detection by clustering lines, in: ICCV, 2009.
[29] S. Maji, J. Malik, Object detection using a max-margin hough transform, in: CVPR, 2009.
[30] C. Gu, J.J. Lim, P. Arbelaez, J. Malik, Recognition using regions, in: CVPR, 2009.
[31] A. Toshev, B. Taskar, K. Daniilidis, Object detection via boundary structure segmentation, in: CVPR, 2010.

[32] P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part based models, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (9) (2010).
[33] Z. Zhang, Y. Cao, D. Salvi, K. Oliver, J. Waggoner, S. Wang, Free-shape subwindow search for object localization, in: CVPR, 2010.
[34] P.D. Kovesi, Matlab and octave functions for computer vision and image processing. Available from 〈http://www.csse.uwa.edu.au/pk/research/matlabfns/〉.
[35] S. Waterman, Identification of common molecular subsequences, Journal of Molecular Biology 147 (1981) 195–197.
[36] T. Motzkin, E. Straus, Maxima for graphs and a new proof of a theorem of turan, Canadian Journal of Mathematics.
[37] N. Bourbakis, P. Yuan, S. Makrogiannis, Object recognition using wavelets, l–g graphs and synthesis of regions, Pattern Recognition 40 (2007) 2077–2096.
[38] J. Weibull, Evolutionary Game Theory, MIT Press, 1997.
[39] M. Everingham, L.V. Gool, C.K.I. Williams, J. Winn, A. Zisserman, The pascal visual object classes challenge 2008 (voc 2008) results 〈http://pascallin.ecs.soton.ac.uk/challenges/voc/voc2008/workshop/〉.
[40] R. Bergevin, J.-F. Bernier, Detection of unexpected multi-part objects from segmented contour maps, Pattern Recognition 42 (11) (2009) 2403–2420.

Xingwei Yang received his B.E. degree in Electronics and Information Engineering from Huazhong University of Science and Technology (HUST), Wuhan, China, in 2002. He obtained his Ph.D degree from Temple University in 2011. Now, he is a Computer Scientist in GE Global Research Center.

Hairong Liu is a postdoctoral research fellow at Embedded Video Lab National University of Singapore.

Longin Jan Latecki is a professor in the Department of Computer and Information Sciences at Temple University in Philadelphia. He is the winner of the 25th Pattern Recognition Society Award together with Azriel Rosenfeld for the best paper published in the journal Pattern Recognition in 1998. He received the main annual award from the German Society for Pattern Recognition (DAGM), the 2000 Olympus Prize. He is a member of the Editor Board of Pattern Recognition and chairs the IST/SPIE annual conference series on Vision Geometry. He has published and edited over 150 research articles and books. His main research areas are shape representation and shape similarity, object recognition, robot mapping, video analysis, data mining, and digital geometry.