CVPR 2011

From Partial Shape Matching through Local Deformation to Robust Global Shape Similarity for Object Detection

Tianyang Ma and Longin Jan Latecki
Dept. of Computer and Information Sciences, Temple University, Philadelphia, USA
tianyang.ma@temple.edu, latecki@temple.edu

Abstract

In this paper, we propose a novel framework for contour based object detection. Compared to previous work, our contribution is three-fold. 1) A novel shape matching scheme suitable for partial matching of edge fragments. The shape descriptor has the same geometric units as shape context but our shape representation is not histogram based. 2) Grouping of partial matching hypotheses to object detection hypotheses is expressed as maximum clique inference on a weighted graph. 3) A novel local affine-transformation to utilize the holistic shape information for scoring and ranking the shape similarity hypotheses. Consequently, each detection result not only identifies the location of the target object in the image, but also provides a precise location of its contours, since we transform a complete model contour to the image. Very competitive results on ETHZ dataset, obtained in a pure shape-based framework, demonstrate that our method achieves not only accurate object detection but also precise contour localization on cluttered background.

1. Introduction

Compared to other image cues, the outline contour (silhouette) is invariant to lighting conditions and variations in object color and texture. More importantly, it can efficiently represent image structures with large spatial extents [20]. Because of these advantages, contour information is widely used in object detection and recognition methods. Recently, several contour-based methods have been demonstrated to work well on the task of object detection and recognition, such as [8], [7], [20] and [21].

Given a gray scale image, edge pixels are obtained by an edge detector, such as Canny [4] or Pb [14]. Then edge pixels are grouped to edge fragments in a bottom up process using an edge-linking algorithm, e.g., [10]. An example of obtained edge fragments is shown in Fig. 1(b), where each edge fragment is marked with a different color. These fragments usually form the input to a contour-based object detection algorithm. Given the contour of the target object as a model, the goal of contour-based object detection is to select a small subset of edge fragments that match well to the model contour. The processes of selection and matching are challenged by the following problems with extracted edge fragments in real images: (1) Edge fragments representing part of the target object are missing, e.g., lower part of the legs in Fig. 1(b). (2) Edge fragments are broken into several pieces. In our example image in Fig. 1(b) both contours of the woman and the swam are broken in many pieces. (3) Part of the true contour of the target object object may be wrongly connected to part of a background contour resulting in a single edge fragment. An example is given in Fig. 1(c), where the yellow edge fragment contains part of the true contour of the swan neck and its reflection in water, which obviously does not belong to the true contour of the swam.

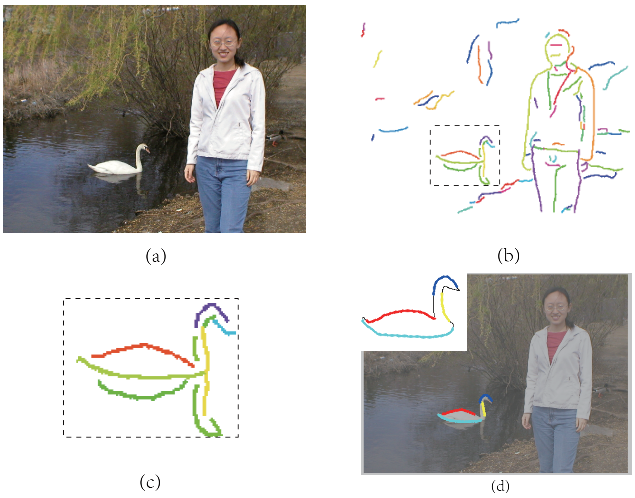

Figure 1. (b,c) show edge fragments obtained from (a), which usually are the input to shape based object detection algorithms. (d) shows a detection example of the proposed approach; the corresponding parts in model and image have the same colors.

These problems are unavoidable in real applications, since a perfect edge detector does not exist [14]. In addition (1) may also result from partial occlusion of the target object, which is common in cluttered scenes. Therefore, any object detection approach must address problems (1-3). Assuming that the contour of the target object is given, problems (1, 2) imply that edge fragments can only match parts of the object contour. The situation is significantly more complex due to (3), which implies that only part of an edge fragment may match to part of the object contour. While all recent approaches , e.g. [21] [12], address the problems (1,2), they suffer from problem (3), since they treat the edge fragments as nonseparable building blocks of the target contours. This may result in missing the target object in the image or locating the object inaccurately, e.g., if the entire yellow fragment is assigned to the swam, the detected bounding box will be larger than the ground truth. To our best knowledge, only the approach in [19] explicitly addresses problem (3) by introducing an efficient partial matching schema based on integral image [22].

However, the final detection evaluation in [19] is appearance based (SVM on HOG features), which demonstrates weakness in the discriminative power of their partial matching schema. There are at least two main reasons for this, one is the selection of the best matching fragments in the integral image framework and the other is simply weak discriminative power of their shape descriptor, which is only angle based.

We utilize the well-known geometric relations of shape context as shape descriptor, but without any histogram representation. One of our main contributions is the selection of the best matching contour fragments in the integral image framework, which by the virtue of the problem is very different from image matching frameworks. As the result we obtain a powerful shape matching framework particularly tailored for partial shape matching. This framework allows us to solve problem (3), since the partial shape matching automatically selects parts of edge fragments that best match to parts of model contour, we essentially generates a new sets of edge fragments. We observe that each of these new edge fragments has a known correspondence to part of the model contour. Thus, partial shape matching is utilized not only to establish the correspondence of edge fragments to model contour parts but also as edge fragment filter.

Given the set of filtered edge fragments and their correspondences to parts of the contour model, our next step is to infer the possible locations of the target object in the image. The inference must simultaneously perform selection and grouping of the edge fragments so that the similarity to the model contour is maximized. We first construct a graph whose nodes are the partial correspondences and edges represent the compatibilities of these correspondences. The location hypotheses are determined as maximal cliques in this graph, i.e., as subgraphs of the weighted graph with maximal affinity of all pairwise connections. To infer the maximal cliques we utilize a recently proposed algorithm [11]. It is very robust in a noisy affinity graph and the number of nodes in a dense-subgraph is automatically determined. These features make it extremely suitable for our task, because the number of fragments to be grouped is unknown and varies a lot depending on the quality of edge fragments. Moreover, the shape of single edge fragments in the image is usually not very discriminative. Each object location hypothesis is identified by several partial correspondences. For example, in Fig. 1(d), four partial correspondences identify the target object. We stress that we not only selected the edge fragments in the image but also the corresponding parts of the model contour. Therefore, we can perform a holistic evaluation of the location hypothesis with global shape similarity, i.e., we score each detection hypothesis with a global shape similarity of grouped edge fragments to the model contour.

However, the target object in the image may be distorted, e.g., due to view point change or nonrigid deformation. In addition, as stated above some parts of the model contour do not have any correspondence in image due to missing edge fragments. Therefore, the shape similarity measure must tolerate deformations and missing parts. However, this makes it less discriminative and increases the risk of "hallucinating" the target object in the background. It follows that it is impossible to tolerate deformations and at the same time keep high discriminative power to avoid hallucinating. This is a very important problem that has not been explicitly addressed by most of the existing approaches.

We address this problem by performing a nonrigid deformation of the model contour according to each detection hypothesis. A nonrigid deformation transformation is obtained by a composition of local affine transformations. Our intuition is that if a detection hypothesis is correct, the deformed model will become more similar to the selected edge fragments, while at the same time it remains similar to the original model. If a detection hypothesis is wrong, the composition of local affine transformations will likely result in a completely deformed model that resembles neither the original model nor the configuration of the selected edge fragments. However, the key benefit of the proposed local affine transformation is its high capability in estimating the position of missing model parts (i.e., parts that do not correspond to any selected edge fragments). This not only results in a robust scoring of the detection hypotheses but also allows us to put the deformed model contour on the image.

2. Related Work

In recent years a large number of contour-based object detection and recognition methods has been proposed.

Many methods achieve state-of-the-art performance by only utilizing edge information. For example, Shotton et al. [20] and Opelt et al. [16] first learn codebooks of contour fragments, then use Chamfer distance to match learnt fragments to edge images. Ferrari et al. [8] [7] build a network of nearly straight adjacent segments (kAS). In [23], Zhu et al. formulate the shape matching of contour in clutter as a set to set matching problem, and present an approximate solution to the hard combinatorial problem by using a voting scheme. They use a context selection scheme by algebraically encoding shape context into linear programming. Ravishankar et al. [18] use short segments to approximate the outer contour of objects. They decompose the model shapes into segments at high curvature points. Dynamic programming is used to group the matched segments in a multi-stage process which begins with triples of segments. Lu et al. [12] first decompose the model into several part bundles. They use particle filters as inference tool to simultaneously perform selection of relevant contour fragments in edge images, grouping of the selected contour fragments, and matching to the model contours. To address the non-rigid object deformation, Bai et al. [1] use the skeleton information to capture the main structure of an object, and use Oriented Chamfer Matching [20] to match the model parts to images. Most recently, Srinivasan et al. [21] address the contour grouping problem as many-to-one matching, and use this scheme in both training and testing phases. For purpose of improving detection and score ranking, a sophisticated training process is designed in which latent SVM is used to guarantee the many-to-one score is tuned discriminatively. Besides of literature mentioned above, edge information is also utilized in [19, 15, 3].

3. Shape Descriptor

We propose a novel shape descriptor that is particularly suitable for shape matching of edge fragments in images to model contours of target objects. Its basic geometric units are the same as in shape context [2]. Shape context (SC) appears to be one of the best performing shape descriptor and definitely the most popular one. Given a planar set X composed of a finite number of points, for every point $x \in X$ we consider both the length and direction of the vector from x to other points in X. However, different from SC, we do not build any histograms representing the lengths and directions.

Given two sequences of points $P = \{p_1 \cdots p_m\}$ and $Q = \{q_1 \cdots q_n\}$ representing two contour fragments in 2D, we compute two matrices, one representing all lengths and the second representing all pairwise orientations of vectors from each $p_i \in P$ to each $q_j \in Q$. As a special case when $P = Q$, the matrices describe the shape of the contour fragment P. The distance $D^{(P,Q)}(i,j)$ from p_i to q_j is defined as Euclidean distance in the log space

$$D^{(P,Q)}(i,j) = \log(1 + ||\vec{p}_i - \vec{q}_j||_2) \qquad (1)$$

We add one to Euclidean distance to make the $D^{(P,Q)}(i,j)$ positive. The orientation $\Theta^{(P,Q)}(i,j)$ from p_i to q_j is defined as the orientation of vector $\vec{p}_i - \vec{q}_j$:

$$\Theta^{(P,Q)}(i,j) = \angle(\vec{p}_i - \vec{q}_j) \in [-\pi, \pi]. \qquad (2)$$

The relative geometric relation of two contour fragments P and Q is encoded in two $m \times n$ matrices $D^{(P,Q)}$ and $\Theta^{(P,Q)}$. An example is given in Fig. 2.

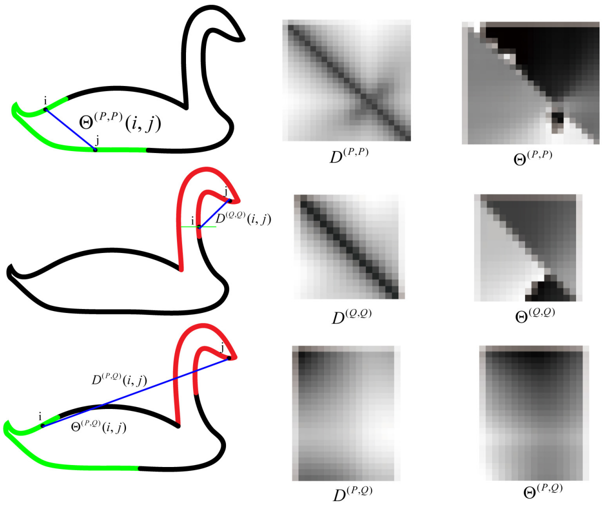

Figure 2. Shape descriptor.

Given another two contour fragments T and U consisting of the same number of points as P and Q, respectively, we define two affinity matrices that measure the similarity of the two fragment configuration (P,Q) to the other two fragment configuration (T,U). The first affinity matrix is based on comparison of distances between two pairs of corresponding pairs of points

$$A_D(P,Q,T,U) = \exp(-\frac{(D^{(P,Q)}(i,j) - D^{(T,U)}(i,j))^2}{(D^{(P,Q)}(i,j)\ \sigma)^2}). \qquad (3)$$

where σ represents the tolerance of distance differences (it is set to 0.2 in all our experiments).

To make the value of $A_D(P,Q,T,U)$ invariant to scale, we divide each distance difference by the distance between the first pair of points. The second affinity matrix is based on angle comparison of vectors connecting the corresponding pairs of points

$$A_\Theta(P,Q,T,U) = \exp(-\frac{(\Theta^{(P,Q)}(i,j) - \Theta^{(T,U)}(i,j))^2}{\delta^2}), \qquad (4)$$

where the difference of angles is taken modulo π, i.e., it is the angle between vectors $\vec{p}_i - \vec{q}_j$ and $\vec{t}_i - \vec{u}_j$, and δ represents the tolerance of angle differences (it is set to $\frac{\pi}{4}$ in all our experiments). Since both A_D and A_Θ are normalized, we can simply add them to obtain the affinity matrix

$$A(P, Q, T, U) = A_D(P, Q, T, U) + A_\Theta(P, Q, T, U). \quad (5)$$

We observe that A is $m \times n$ matrix representing the similarities of corresponding point pairs in (P, Q) and (T, U). The similarity of two configurations of contour fragments (P, Q) and (T, U) is defined as

$$\Psi(P, Q, T, U) = \frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} A(P, Q, T, U). \quad (6)$$

As a special case of Eq. (6), we obtain a similarity between two contour fragments P and T defined as

$$\Psi(P, T) = \Psi(P, P, T, T) \quad (7)$$

(here we slightly abuse the notation for the sake of simplicity). When Q is the same as P in Eq. (1) and (2), the matrices $D^{(P,P)}$ and $\Theta^{(P,P)}$ represent all pairwise distances between all pair of points of P and corresponding angles of the vector connecting the points. Thus, two matrices form a shape descriptor of the contour fragment P and similarly for T. Hence $\Psi(P, T)$ simply compares the shape descriptor of the contour fragments P and T.

4. Partial Matching between Edge Fragments and Model Contour

Given an image I, using edge-linking software [10], a set of edge fragments $E = \{e_1 \cdots e_K\}$ is generated. Each fragment e_k is a list of N_k points (i.e., pixels) $\{q_1, \cdots, q_{N_k}\}$. According to our descriptor, the geometry of fragment e_k is encoded in two $N_k \times N_k$ matrices: A_D and A_Θ. Similarly, two $M \times M$ matrices are used to fully represent the contour of a model $\mathcal{M}$ composed of points $\{p_1, \cdots, p_M\}$.

Our goal is to find the best matching between a part of image edge fragment e_k with a part of model fragment p_m. Thus, we need to find a part $\mathcal{M}(i, l) = \{p_i, \cdots, p_{i+l-1}\} \subseteq \mathcal{M}$, where i is the starting point of the part and l is its length. (The indices are modulo M if the model contour fragment is a closed curve.) Since cannot expect that the whole image fragment participates in the matching, we need to simultaneously select part $e_k(j, l) = \{q_j, \cdots, q_{j+l-1}\} \subset e_k$, where j is the starting point of the fragment part and its length is also l.

Our goal can be expressed as finding two corresponding subblocks of their shape matrices with the maximum similarity Ψ defined in (7). To achieve this goal we construct a 4D tensor matrix

$$\Gamma(i, j, l, k) = \Psi(\mathcal{M}(i, l), e_k(j, l)) \quad (8)$$

and observe that $\Gamma(i, j, l, k)$ can be computed efficiently by utilizing the integral image algorithm, since it allows to access any element in the 4D matrix in constant time [5, 22].

Intuitively, when very few points are involved in a matching, the shape similarity is neither reliable nor discriminative enough. Therefore, we set a threshold τ on the minimal number of matching points and set $\Gamma(i, j, l, k) = 0$ if $l < \tau$. We then take the maximum of the 4D matrix along different l, and suppress it to

$$S(i, j, k) = \max_{l} \Gamma(i, j, l, k) \quad (9)$$

We observe that the index of the maximal value of S determines a pair of best matching subsegments of $\mathcal{M}$ and e_k:

$$G(i, j, k) = \arg\max_{l} \Gamma(i, j, l, k) = (\mathcal{M}(i, l), e_k(j, l)). \quad (10)$$

Based on these local observations, the most popular method to form object location hypothesis is using Hough voting, such as in [19]: local maxima of $S(i, j, k)$ for certain fragment e_k are identified, and corresponding fragment correspondences are used to estimate object location by Hough voting. However, Hough voting seems not to be an optimal choice here. When each part correspondence independently cast a vote, the cluttered background is more likely to get a larger score, since single edge fragments are unlikely to carry discriminative shape information.

More discriminative shape information can be obtained by considering all pairwise shape relations of several edge fragments. We introduce a graph-based clustering method to find location hypothesis through which shape dependency of local edge fragments is naturally captured.

5. Object Localization as Maximal Clique Computation in a Weighted Graph

Each vertex $v \in V$ of our graph corresponds to a partial match $G(i, j, k)$ (10), i.e., v represents a model segment $\mathcal{M}(i, l)$ selected as best matching to part $e_k(j, l)$ of the edge fragment e_k. To limit the number of vertices $G(i, j, k)$, for each point i in model $\mathcal{M}$, we only choose the best K matches as vertices according to their corresponding similarity $S(i, j, k)$. Therefore, for a given model $\mathcal{M}$ contour with M points, the number of vertices is equal to $M \times K$.

Given two pairs of matches, i.e., two vertices $v_i = \{\mathcal{M}(i_1, l_1), e_m(j_1, l_1)\}$ and $v_j = \{\mathcal{M}(i_2, l_2), e_n(j_2, l_2)\}$, if $v_i \neq v_j$ we define the edge weight as

$$A(i, j) = \Psi(\mathcal{M}(i_1, l_1), \mathcal{M}(i_2, l_2), e_m(j_1, l_1), e_n(j_2, l_2)), \quad (11)$$

which measures the shape similarity of the configuration of two model segments $\mathcal{M}(i_1, l_1)$ and $\mathcal{M}(i_2, l_2)$ to a corresponding configuration $e_m(j_1, l_1)$ and $e_n(j_2, l_2)$ of two parts of edge fragments. As a special case, we define

$$A(i, i) = \Psi(\mathcal{M}(i_1, l_1), e_m(j_1, l_1)), \quad (12)$$

which measures the shape similarity of a single model segment $\mathcal{M}(i_1, l_1)$ to a corresponding edge part $e_m(j_1, l_1)$.

To sparsify the affinity matrix A, we observe that $e_m(j_1, l_1)$ and $e_m(j_2, l_2)$ can only correspond to $\mathcal{M}(i_1, l_1)$ and $\mathcal{M}(i_2, l_2)$ if they are relatively close to each other. In practice, we compare the average value of distance matrix $D^{(e_m(j_1,l_1),e_m(j_2,l_2))}$ to average value of $D^{(\mathcal{M}(j_1,l_1),\mathcal{M}(j_2,l_2))}$. If the difference is larger than a reasonable value, we set $A(i, j) = 0$ (for instance in our experiment, it is the square root of model size multiply the scale).

Meanwhile, partial matching v_i and v_j may refer to the corresponding of the similar position of model only with a few pixels offset. We do not want to have these kind of partial matches co-occur in a solution of clustering, since for a true positive configuration of an object hypothesis, it is impossible that several fragments in image corresponding to the same part of model. Based on $f = \frac{|\mathcal{M}(i_1,l_1)\cap\mathcal{M}(i_2,l_2)|}{|\mathcal{M}(i_1,l_1)\cup\mathcal{M}(i_2,l_2)|}$, we tell if v_i and v_j get the same part of model involved in. If $f < t$, we set $A(i, j) = 0$. In experiment, t equals to 0.5.

The obtained weighted affinity graph is denoted as $G = (V, A)$. Our goal is to find all maximal cliques in this graph. As stated in [17], a maximal clique is a subset of V with maximal average affinity between all pairs of its vertices, which is equivalent to the fact that the overall similarity among internal elements is higher than that between external and internal elements. In our case, given a shape model and corresponding partial matches in the image, clustering is expected to find several pairs of matches with high values of all pairwise similarities. To formally state our goal, we introduce an indicator vector **x** over the vertices V, i.e., has $M \times K$ coordinates. A vertex $v \in V$ is selected as belonging to a maximal clique if and only if $x_v > 0$, where x_v denotes the v coordinate of x. Then each maximal clique is defined as the solution of the following quadratic program

$$\begin{aligned} &\text{maximize} \;\; f(\mathbf{x}) = \mathbf{x}^T A \mathbf{x} \\ &\text{subject to} \;\; \mathbf{x} \in \triangle, \end{aligned} \tag{13}$$

where $\triangle = \{\mathbf{x} \in R^{M\times K} : \mathbf{x} \geq 0 \;\text{ and }\; ||\mathbf{x}||_1 = 1\}$ is the simplex in $R^{M\times K}$.

Each maximal clique corresponds to a local solution of Eq. (13). We are using the recently proposed algorithm in [11] to compute the local solutions. Each solution **x**, i.e., maximal clique, is treated as an object detection hypothesis. It consists of several model contour segments and the corresponding parts of edge fragments. The final evaluation of the hypotheses is presented in the next section.

6. Evaluation of Detection Hypotheses

By considering the partial matches as a whole, a detection hypothesis is expressed as the correspondence between a subset of points on the model and a subset of edge points in image. We denote the subset of model points as $\mathcal{M}_a \subset \mathcal{M}$, and subset of image edge points as $E_a \subset E$. Clearly there exists a bijection T between $\mathcal{M}_a \subset \mathcal{M}$ and $E_a \subset E$, i.e., if $x \in \mathcal{M}_a, T(x) \in E_a$. For each hypothesis, there are usually some points in the model that have no correspondence in the image, i.e., $\mathcal{M}_b = \mathcal{M} \setminus \mathcal{M}_a \neq \varnothing$. The mapping $T : \Re^2 \to \Re^2$ can be regarded as affine-transformation Z which consists of scaling, translation and rotation. Here, we intend to extend T to conclude the transformation Z for $x \in \mathcal{M}_b$. Therefore, we define T for $x \in \mathcal{M}$ as following:

$$\begin{aligned} T(x) = &\;\; \mathcal{M}_a \to E_a, \;\text{ if }\; x \in \mathcal{M}_a \\ = &\;\; xZ, \qquad \text{if }\; x \in \mathcal{M}_b \end{aligned} \tag{14}$$

For each point among $\mathcal{M}_b$, our goal is to determine the appropriate affine-transformation based on existing mapping relations $\mathcal{M}_a \to E_a$. We attempt to locally estimate Z for every $x \in \mathcal{M}_b$. This is motivated by the observation that affine transformations of points belong to the same part of model are usually consistent, e.g., the points on swan neck. Based on the distance of indices in the model points sequence, we find the a certain number of close points of $x \in \mathcal{M}_b$, and denote them by $N(x) \subset \mathcal{M}_a$. The reason that we define distance as difference between points indices instead of their geometry closeness is: $\mathcal{M}$ is an ordered points set, point connectedness is more important than the closeness in geometry. Then Z is computed as:

$$Z = \min_{Z^*} \;\; d(T(N(x)), N(x)Z^*) \tag{15}$$

Here, function d is simply computing the accumulate square distance between $T(N(x))$ and $N(x)Z^*$. Thus, Eq. (15) is turned into

$$\begin{aligned} T(x) = &\;\; \mathcal{M}_a \to E_a, \;\text{ if }\; x \in \mathcal{M}_a \\ = &\;\; x \min_{Z^*} \;\; d(T(N(x)), N(x)Z^*), \qquad \text{if }\; x \in \mathcal{M}_b \end{aligned} \tag{16}$$

By applying mapping T on every point $x \in \mathcal{M}$, a set of points $T(\mathcal{M})$ corresponding to model points is obtained. It is used for later scoring.

6.1. Scoring and Ranking

As mentioned above, the confidence for a hypothesis is evaluated from two aspects.

$$S(T(\mathcal{M})) = \Psi(\mathcal{M}, T(\mathcal{M})) \times \Psi(T(\mathcal{M}), T'(\mathcal{M})) \tag{17}$$

The first score indicates how well M is corresponded to $T(\mathcal{M})$ considering the geometric arrangement, which is simply computed using Eq. (6).

Moreover, we also need to measure if $T(\mathcal{M})$ is consistent with the contour cues in image. This is indicated by the second score. For this purpose, we first calculate tangent direction θ for both points in $T(x), x \in \mathcal{M}_b$ and edge points E in image. This makes each point to be 3D data, i.e., $[x, y, \theta]$. In this 3D space, for each point in $T(x), x \in \mathcal{M}_b$, we use kd-tree algorithm to find the closest point in E. All these closest points from E are aggregated, together with the points in E_a, are denoted by $T'(\mathcal{M})$. We measure the similarity between $T(\mathcal{M})$ and $T'(\mathcal{M})$ using Eq. (6). Finally, we rank all obtained hypothesis according to the confidence $S(T(\mathcal{M}))$.

7. Experimental Results

We present results on the ETHZ shape classes [8] which features five diverse classes (bottles, swans, mugs, giraffes, apple-logos) and contains a total of 255 images. For all categories, there are significant inner-class variations, scale changes, and illumination changes. Most importantly, the dataset comes with ground truth gray level edge maps, which is computed by Pb edge detector [14]. This makes it possible to have a fair comparison of contour-based object detection methods.

Depending on the way of selecting shape models for each category, we follow two different experiment protocols. First, we utilize single hand-drawn shape model for each class, and testing is done on all 255 images. Second, we follow the protocol in [7]. We 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. In our approach we only use the ground truth outlines of objects present in the first half of images for each class. We apply our shape descriptor to compute pairwise similarity of the outlines, and use affinity propagation clustering algorithm [9] to automatically obtain several prototype shape models. Thus, our training is only used to select prototype contour models.

For the purpose of detection evaluation, we follow the PASCAL criteria, i.e., a detection is deemed as correct if the intersection of detected bounding box and ground truth over the union of the two bounding boxes is larger than 50%.

To convert the gray level edge map to binary edge map, we set all pixels with their values larger than 0 as edge pixels. This means we do not adjust the threshold to get better edges. During detection, 5 different scales are searched for every image. Non-maximum suppression is used to remove duplicate hypothesis.

We focus on comparison to the state-of-the-art contour-based object detection methods, in particular to [7, 21, 12]. We plot the precision/recall (PR) curves in Fig. 3. Table 1 shows the interpolated average precision (AP) value for 5 methods. Our method achieves the best mean AP and the best AP for category Swans. Our AP is comparable to the best ones in the other four classes. The mean AP of our method is slightly better than [21] and much better than the other contour-based methods.

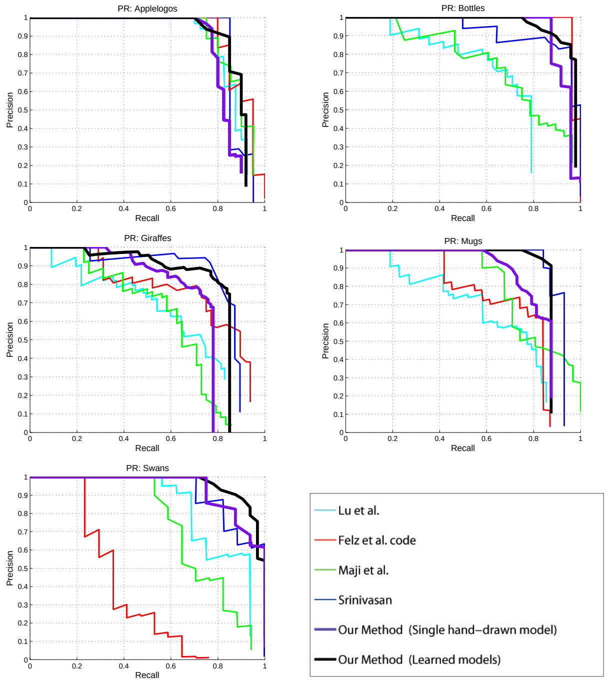

Figure 3. Precision/Recall curves of our method compared to Lu et al. [12], Felz et al. [6], Maji et al. [13], and Srinivasan et al. [21] on ETHZ shape classes. We report both the results with single hand-drawn model and with learned models.

We also show the false positives per image (FPPI) vs. detection rate (DR) in Fig. 4. Table 2 compares the detection rates at 0.3/0.4 FPPI. Our method also achieve comparable result to [21], but the mean value of [21] is slightly better than ours for this measure. We observe that our method is the only one with no difference in detection rates at 0.3 FPPI and 0.4 FPPI. The curve of our methods increases sharply at the beginning and reaches the peak of the detection rate before 0.3/0.4 FPPI.

Besides the presented evaluation of the object detection accuracy, which is based on bounding box intersection, accuracy of localizing the boundary of detected objects is extremely important in many applications. Since our final detection evaluation includes nonrigid deformation of a contour model and positioning the deformed model on the edge image, we are able not only to precise localize the boundary but also to complete the missing contours. This fact is illustrated by our example detection results shown in Fig. 5.

To qualitatively evaluate the contour detection accuracy, we use the coverage and precision measure defined in [7]. The coverage value shows what percentage of true boundaries have been successfully detected. The precision values measures how many detected edge points are inside the true boundaries. We compare the coverage/precision of our

	Applelogos	Bottles	Giraffes	Mugs	Swans	Mean
Our method	0.881	0.920	0.756	0.868	**0.959**	**0.877**
Srinivasan et al. [21]	0.845	0.916	**0.787**	**0.888**	0.922	0.872
Maji et al. [13]	0.869	0.724	0.742	0.806	0.716	0.771
Felz et al. code [6]	**0.891**	**0.950**	0.608	0.721	0.391	0.712
Lu et al. [12]	0.844	0.641	0.617	0.643	0.798	0.709

Table 1. Comparison of interpolated average precision (AP) on ETHZ Shape classes.

	Applelogos	Bottles	Giraffes	Mugs	Swans	Mean
Our method	0.92/0.92	0.979 / 0.979	0.854/0.854	0.875/0.875	**1 / 1**	0.926 / 0.926
Srinivasan et al. [21]	**0.95/0.95**	**1 / 1**	0.872/0.896	0.936/0.936	**1 / 1**	**0.952 / 0.956**
Maji et al. [13]	0.95/0.95	0.929 / 0.964	**0.896/0.896**	**0.936/0.967**	0.882 / 0.882	0.919 / 0.932
Felz et al. code [6]	0.95/0.95	**1 / 1**	0.729/0.729	0.839/0.839	0.588 / 0.647	0.821 / 0.833
Lu et al. [12]	0.9/0.9	0.792 / 0.792	0.734/0.77	0.813/0.833	0.938 / 0.938	0.836 / 0.851
Riemenschneider et al. [19]	0.933/0.933	0.970 / 0.970	0.792/0.819	0.846/0.863	0.926 / 0.926	0.893 / 0.905
Ferrari et al. [7]	0.777/0.832	0.798 / 0.816	0.399/0.445	0.751/0.8	0.632 / 0.705	0.671 / 0.72
Zhu et al. [23]	0.800/0.800	0.929 / 0.929	0.681/0.681	0.645/0.742	0.824 / 0.824	0.776 / 0.795

Table 2. Comparison of detection rates for 0.3/0.4 FPPI on ETHZ Shape classes.

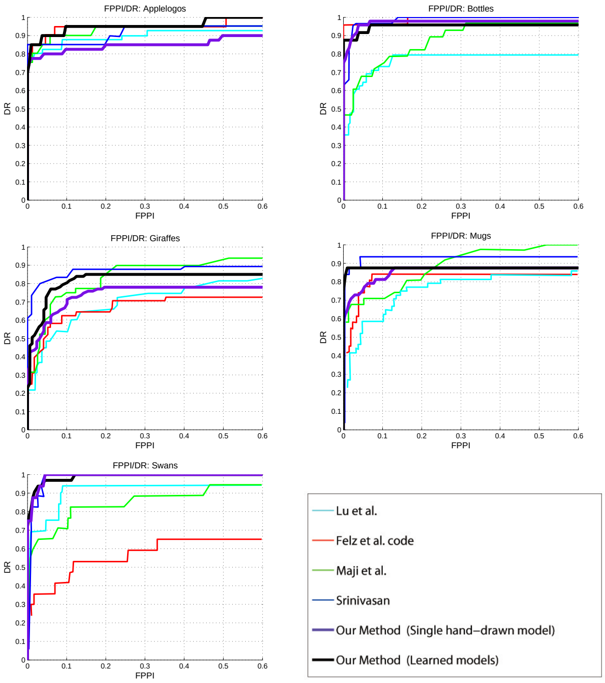

Figure 4. Comparison of DR/FPPI curves on ETHZ shape classes.

method with [7] in Table 3. Our method achieves a higher precision value on all 5 classes, especially there is a big improvement for Applelogos, Bottles, and Swans. For coverage, our method is better on 3 classes, but worse on the classes of Giraffes and Mugs. The reason is that our models for Giraffes and Mugs are very simple, in particular, we do not have the inner contour of the mug handle and the lower part of the giraffe outline as can be seen in Fig. 5. Therefore, some part of the true boundaries, such as the internal handle of mugs, are not detected.

	Our method	Ferrari et al. [8]
Applelogos	**0.923/0.948**	0.916/0.939
Bottles	**0.845/0.903**	0.836/0.845
Giraffes	0.456/0.784	**0.685/0.773**
Mugs	0.735/0.803	**0.844/0.776**
Swans	**0.848/0.909**	0.777/0.772

Table 3. Accuracy of boundary localization of the detected objects. Each entry is the average coverage/precision over trials and correct detections at 0.4 FPPI.

8. Conclusion

We present a novel framework for contour based object detection with three main contributions. First, we introduce a partial shape matching scheme suitable for matching of edge fragments, in which the shape descriptor has the same geometric units as shape context but is not histogram based. Second, we group partial matching hypotheses to object detection hypotheses via maximum clique inference on a weighted graph instead of Hough voting. Third, a unique feature of our approach is that we perform nonrigid deformation of a contour model and position the deformed model on the edge image. Our deformation is based on a local affine-transformation guided by the partial matching to edge fragments. By combining these components, we obtain an effective purely shape-based object detection

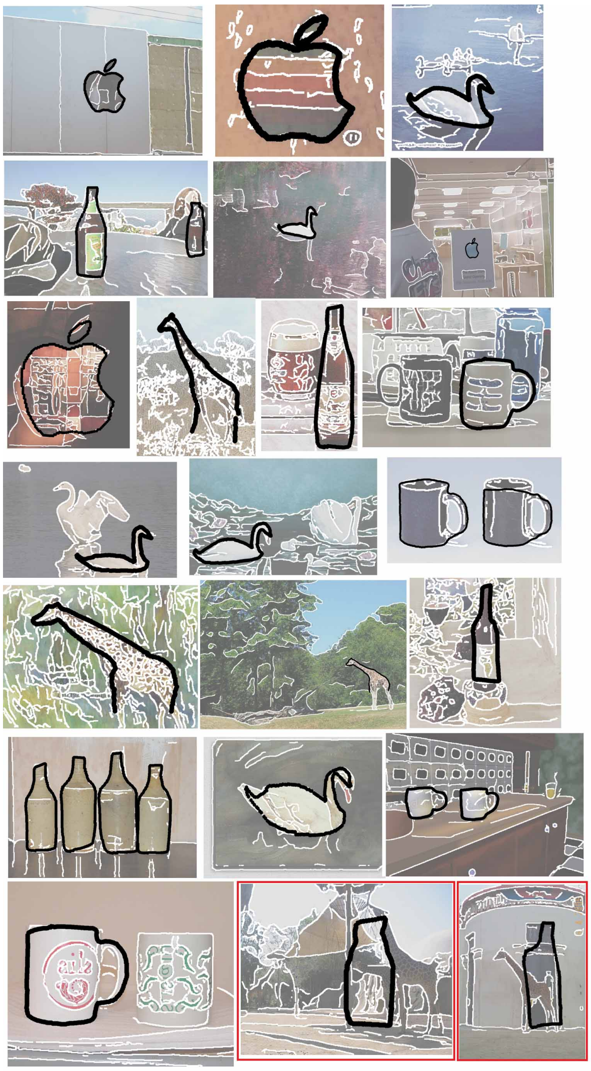

Figure 5. Some detection results of ETHZ dataset. The edge map is overlaid in white on the original images. Each detection is shown as the transformed model contour in black. The red framed images in the bottom row show two false positives.

framework. Our method compares favorable to other state-of-the-art purely shape based methods. In particular, we achieve the best average precision (AP) value averaged over all 5 classes of the ETHZ dataset. The evaluation on the ETHZ dataset demonstrates that the proposed method not only achieves accurate object detection but also precise contour localization on cluttered background.

Acknowledgments

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

References

[1] X. Bai, X. Wang, L. J. Latecki, W. Liu, and Z. Tu. Active skeleton for non-rigid object detection. *ICCV*, 2009.

[2] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. PAMI*, 24(1):705–522, 2002.

[3] A. Berg, T. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondences. *CVPR*, 2005.

[4] J. Canny. A computational approach to edge detection. *IEEE Trans. PAMI*, 6:679–698, 1986.

[5] M. Donoser, H. Riemenschneider, and H. Bischof. Efficient partial shape matchingof outer contours. *ACCV*, 2009.

[6] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. *CVPR*, 2008.

[7] V. Ferrari, F. Jurie, , and C. Schmid. From images to shape models for object detection. *International Journal of Computer Vision*, 2009.

[8] V. Ferrari, T. Tuytelaars, and L. V. Gool. Object detection with contour segment networks. *ECCV*, 2006.

[9] B. J. Frey and D. Dueck. Learning to detect natural image boundaries using local brightness, color, and texture cues. *Science*.

[10] P. D. Kovesi. Matlab and octave functions for computer vision and image processing. 2008.

[11] H. Liu, L. J. Latecki, and S. Yan. Robust clustering as ensemble of affinity relations. *NIPS*, 2010.

[12] C. Lu, L. J. Latecki, N. Adluru, X. Yang, and H. Ling. Shape guided contour grouping with particle filters. *ICCV*, 2009.

[13] S. Maji and J. Malik. A max-margin hough tranform for object detection. *CVPR*, 2009.

[14] D. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Trans. PAMI*, 2004.

[15] B. Ommer and J.Malik. Multi-scale object detection by clustering lines. *ICCV*, 2009.

[16] A. Opelt, A. Pinz, and A. Zisserman. A boundary-fragmentmodel for object detection. *ECCV*, 2006.

[17] M. Pavan and M. Pelillo. Dominant sets and pairwise clustering. *IEEE Trans. PAMI*, 29:167–172, 2007.

[18] S. Ravishankar, A. Jain, and A. Mittal. Multi-stage contour based detection of deformable objects. *ECCV*, 2008.

[19] H. Riemenschneider, M. Donoser, and H. Bischof. Using partial edge contour matches for efficient object category localization. *ECCV*, 2010.

[20] J. Shotton, A. Blake, and R. Cipolla. Multi-scale categorical object recognition using contour fragments. *IEEE Trans. PAMI*, 2008.

[21] P. Srinivasan, Q. Zhu, and J. Shi. Many-to-one contour matching for describing and discriminating object shape. *CVPR*, 2010.

[22] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *CVPR*, 1:511–518, 2001.

[23] Q. Zhu, Y. W. L. Wang, and J. Shi. Contour context selection for object detection: A set-to-set contour matching approach. *ECCV*, 2008.