A Method of Perceptual-based Shape Decomposition

Chang Ma, Zhongqian Dong, Tingting Jiang, Yizhou Wang, Wen Gao
National Engineering Lab for Video Technology,
Key Lab of Machine Perception (MoE),
School of EECS, Peking University, Beijing, China
{machang_pku, zqdong, ttjiang, yizhou.wang, wgao}@pku.edu.cn

Abstract

In this paper, we propose a novel perception-based shape decomposition method which aims to decompose a shape into semantically meaningful parts. In addition to three popular perception rules (the Minima rule, the Short-cut rule and the Convexity rule) in shape decomposition, we propose a new rule named part-similarity rule to encourage consistent partition of similar parts. The problem is formulated as a quadratically constrained quadratic program (QCQP) problem and is solved by a trust-region method. Experiment results on MPEG-7 dataset show that we can get a more consistent shape decomposition with human perception compared with other state-of-the-art methods both qualitatively and quantitatively. Finally, we show the advantage of semantic parts over non-meaningful parts in object detection on the ETHZ dataset.

1. Introduction

Many psychological studies have shown the important role of parts that plays in object perception and recognition (*e.g.*, [8, 19]). Part-based object representation becomes popular due to its flexibility and robustness to account for the compositional structures of objects.

Object parts are usually generated by two types of methods, the learning-based methods and the rule-based methods. The former learns parts by grouping small curve segments into larger structures (as parts) according to the statistics of a dataset (*e.g.*, [5, 22]). However, most of the datasets are of limited size and prone to be biased, hence, the learned parts usually are dataset-dependent and quite different from human perception. In addition, the learning can be computationally expensive.

We argue that perceptual meaningful parts have the advantage in many vision tasks, such as object detection, because such parts are usually more stable in different environments. This is because that the human vision system (HVS) has evolved for millions of years to adapt to the visual structure of natural scenes from a huge amount of examples in the physical world, which is adept at capturing the statistically stable structures of objects in the presence of deformation and articulation. In addition, the semantic parts are useful in judging the affordance of objects and they are the key to transferring the knowledge between different objects via the shared parts [1].

In order to obtain semantic parts, perception-rule-based methods are often adopted to decompose a shape into a number of parts (*e.g.*, [7, 9, 14, 17]). The perception rules are usually generic and easy to compute. Some generic perception rules have been examined in psychological studies, *e.g.*, the Minima rule [8], the Short-cut rule [20] and the Convexity rule [11, 21]. The Minima rule suggests the shape should be divided at loci of negative minima of curvature along the contour. The Convexity rule requires a part to be convex. The Short-cut rule suggests to decompose shapes into parts using the shortest possible cuts.

Lien and Amato [13] and Gopalan *et al*. [7] proposed methods for approximate shape decomposition based on the Convexity rule. Liu *et al*. [14] formulated the convex shape decomposition as a linear programming problem and considers the Short-cut rule. Ren *et al*. [17] and Jiang *et al*. [9] proposed methods for perception-based shape decomposition which add the Minima rule in their works.

In this paper, we propose a new method to acquire semantic parts via perception-based shape decomposition. We observe that objects often have similar parts (*e.g.*, the four legs and feet of a camel in Fig. 3). Even these parts may look different due to deformation, people still try to extract them by making them as similar as possible. Therefore, besides the existing rules adopted by [9, 14, 17], we add a new rule to encourage the consistent decomposition of similar parts. In addition, most of the works based on the Convexity rule tend to generate redundant parts in order to satisfy the convexity constraint. In this work, we add a quadratic constraint to avoid this problem.

We formulate the problem as a combinatorial optimization problem by selecting an optimal subset of candidate

cuts. We solve this constrained optimization problem by a trust-region method designed for nonlinear programming.

Extensive experiments have been done on the MPEG-7 [12] shape dataset. We propose a novel quantitative measure to evaluate the decomposition consistency of the proposed method w.r.t. human decomposition. Experiment results demonstrate significant improvement in this respect both qualitatively and quantitatively.

To prove the advantage of semantic parts in the vision tasks, we conduct the object detection experiment on the ETHZ dataset [6]. We use the semantic parts obtained by the proposed decomposition method to detect objects in natural scenes and compare the detection rate with the same detection method using a set of non-semantic random parts. The result shows that the detection rate of using the semantic parts is much better than the random parts, especially in the cases of objects with articulation.

The rest of the paper is organized as follows. Section 2 introduces the proposed method. Section 3 shows the experiment results. Finally, Section 4 concludes the paper.

2. Perceptual-based Shape Decomposition

2.1. Perception Rules

The three perception rules usually adopted by humans to decompose a shape include the *Minima rule* [8], the *Short-cut rule* [20] and the *Convexity rule* [11, 21]. The Minima rule suggests that the endpoints of a cut usually locate at the places where the curvature is local minimum. The Short-cut rule prefers to minimize the total cut length, and the Convexity rule requires parts to be convex.

Apart from these rules, we also observe that an object often has similar parts, *e.g*. four legs and feet of a camel (as shown in Fig. 3). Even they may appear differently on a projected profile due to pose variation and deformation, people tend to identify them equally since they share the same perceptual meaning/function in understanding a shape. We name this rule as *part-similarity rule*.

2.2. Formulation

Perceptual-based shape decomposition is to decompose a shape into non-overlapping parts consistently with human perception. In our problem, parts are generated by a set of cuts on a shape. Each cut is a line segment, both its endpoints lie on the boundary of the shape. So the problem can be formulated as selecting an optimal subset of candidate cuts which can derive perceptual-based parts and do not intersect with each other.

We define C_p as a set of candidate cuts and C^* as the selected optimal cuts. We introduce a binary vector x to indicate the selection of the candidate cuts. $x_i = 1$ means C_i is selected; otherwise, $x_i = 0$.

Then our problem can be formulated as

$$\min_x \{L^T x - aI^T x + x^T H x - bx^T H_{sim} x\} \quad \textit{s.t. } P_i \text{ is convex } \forall i. \tag{1}$$

According to the Short-cut rule [20], L is the cut length vector *s.t.* L_i is the length of cut C_i. $H_{n\times n}$ is a penalty matrix *s.t.* if C_j and C_k intersect, $H(i,j) = +\infty$; otherwise, $H(i,j) = 0$. $I()$ is a function that measures the improvement of shape convexity by a cut. $H_{sim}()$ encourages consistent cuts for similar parts. a and b are two free parameters. $\{P_i\}$ are parts derived by C^*.

In the following we will first explain how to formulate the "convex" constraint so as to obtain near-convex parts $\{P_i\}$, then, introduce the definitions of $H_{sim}()$ and $I()$.

2.3. Convexity Constraint

As *strict convex* shape decomposition can generate many spurious parts due to the noise (*e.g*., small bumps) or small deformation (*e.g*., bending) on shape, we consider *near-convex* shape decomposition, *i.e.*, given a threshold ε, we want to ensure the concavity of a decomposed part is less than ε.

In this project, we obtain near-convex parts through cuts. In the following, we first introduce the concavity measurement of two points and a part as proposed in [14], then present the method of generating cuts to ensure the convexity of parts, followed by the formulation of the convexity constraint in Eq. 1.

2.3.1 Mutex pair and Concavity measures

Given an arbitrary direction (*e.g*., the vertical direction shown in Fig. 1), the concavity of two points within a shape w.r.t. the given direction is defined as:

$$\text{Concavity}_f(p_1,p_2) = f(S_f(p_1,p_2)) - \max(f(p_1), f(p_2)), \tag{2}$$

where $f()$ is the Morse function, which is defined as the projection of a point on the given direction, *i.e.*, $f(p) = < d, p >$ where d is the unit vector representing the direction and $< . >$ is the inner product. $S_f(p_1,p_2)$ is the lowest of the point set along the given direction composed of points which are the highest on the path connecting a given pair of points (*e.g*., p_1 and p_2) within the shape.

Fig. 1(a) gives an example, of which $S_f(p_1,p_2)$ and $S_f(p_2,p_3)$ are the same and we denote them as S. Thus $\text{Concavity}_f(p_1,p_2) = f_S - f_1$ is less than ε, whereas, $\text{Concavity}_f(p_2,p_3) = f_S - f_2$ is greater than ε. Any pair of points whose concavity is more than ε is defined as a *mutex pair* [14]. So, $\langle p_2, p_3 \rangle$ is a *mutex pair*. And $S_f(p_2,p_3)$ is called a *saddle point* of the mutex pair w.r.t. the given direction.

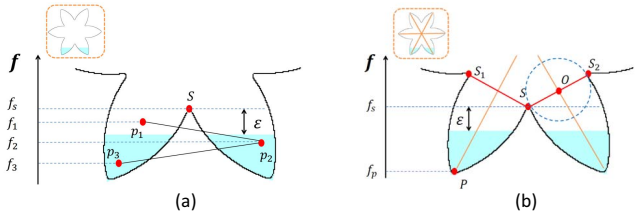

Figure 1. (a) Vertices p_2 and p_3 is a mutex pair while p_1 and p_2 is not under the threshold ε. S is a saddle point and f is Morse function which both correspond to the given direction. The blue regions are mutex pair of regions. (b) Red lines $\overline{SS_1}$ and $\overline{SS_2}$ are two candidate cuts generated by S, the orange lines are the skeleton of the shape.

If we consider all the directions, then the concavity of a pair of points is defined as

$$\text{Concavity}(p_1, p_2) = \max_f \text{Concavity}_f(p_1, p_2), \quad (3)$$

For a shape part P, its concavity is defined as

$$\text{Concavity}(P) = \max_{p_1 \in P, p_2 \in P} \text{Concavity}(p_1, p_2), \quad (4)$$

where p_1 and p_2 are two arbitrary points in P. If the concavity of every pair of points in a part is less than ε, then the concavity of the part is less than ε. We call such a part a "ε-convex part". To ensure all the decomposed parts are ε-convex, we shall separate all the mutex pairs of a shape by cuts, although some of these cuts are spurious. Our goal is to find the optimal set of cuts for shape decomposition.

In order to extend the concept of mutex pair from point to point set, two concavity measures of two point sets R_1 and R_2 are defined as

$$w(R_1, R_2) = \min_{p_1 \in R_1, p_2 \in R_2} \text{Concavity}(p_1, p_2), \quad (5)$$

$$W(R_1, R_2) = \max_{p_1 \in R_1, p_2 \in R_2} \text{Concavity}(p_1, p_2). \quad (6)$$

If $w(R_1, R_2) \geq \varepsilon$, every pair of points from R_1 and R_2 forms a mutex pair. R_1 and R_2 is called a mutex pair of regions (*e.g.*, the blue regions shown in Fig. 1(a)). The concavity of R_1 and R_2 is defined as $W(R_1, R_2)$. We use the method in [4, 14] to find mutex pairs of regions in our implementation and select a subset of candidate cuts to satisfy them in order to get ε-convex decomposition.

2.3.2 Generating candidate cuts to separate the mutex pairs

By considering the Minima rule introduced above, we propose candidate cuts using all the saddle points of a shape. Specifically, we use a saddle point as one of the endpoints of a cut. In order to find the other, we utilize the idea of local symmetry (A part is locally symmetric w.r.t. its skeleton.). A skeleton point has two symmetric points on the shape boundary. The two points are the contacting points between the shape contour and its maximal disk centered at the skeleton point [18]. For example, in Fig. 1(b), the orange lines are the skeleton of the shape. S and S_2 are the two corresponding contour points of a skeleton point O. Hence, to get a candidate cut, we first compute the skeleton of a shape using method in [18], then find the symmetric points of the saddle point S based on its skeletons, *i.e.*, S_1 and S_2. Then the cuts $\overline{SS_1}$ and $\overline{SS_2}$ are two candidate cuts generated by S.

2.3.3 Formulating the convexity constraint

The candidate cuts generated by the above method are usually surplus. We shall select an "optimal" set of cuts that is able to separate all the mutex pairs, and hence generates near-convex parts. (The meaning of the optimality encoded in Eq. 1.)

To achieve this, a binary matrix A is defined, which signifies the separation relationship between the mutex pairs ($MP = \{mp_1, mp_2, \ldots, mp_m\}$) and the candidate cuts C_p. A is a matrix of size $m \times n$, where m is the number of mutex pairs, n is the number of candidate cuts. If a mutex pair mp_i can be separated by a cut C_j, then $A(i, j) = 1$; otherwise 0. So if we constrain $A(i, :)x \geq 1$, then mutex pair i is separated at least once by the optimal set of cuts, which is also used in [14].

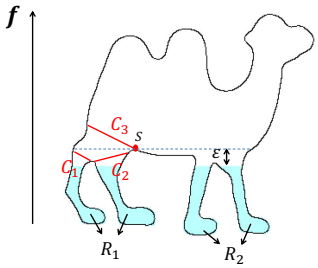

Figure 2. An example of a mutex pair satisfied by two cuts. The mutex pair R_1 and R_2 can be separated by the combination of cuts C_1 and C_2, so there is no need for C_3.

Pruning redundant cuts Although we can satisfy all the mutex pairs using the above constraint, it can produce redundant cuts. This is due to the double counting of mutex pairs. As shown in Fig. 2, R_1 and R_2 are a mutex pair of regions w.r.t. saddle point S. C_3 serves to separate them. However, the combination of two lower level cuts C_1 and C_2 is also able to separate R_1 and R_2; in addition, they can separate the two rear legs of the camel as well. Hence C_3 becomes redundant.

We design a series of matrices $A_2^{\{1,2,\cdots,m\}}$ of which A_2^i is a binary matrix signifying the separation of mutex pair

mp_i by all candidate cut pairs. The size of A_2^i is $n \times n$. If the mutex pair mp_i can be separated by the combination of cut C_j and C_k, then $A_2^i(j,k) = 1$; otherwise 0.

We extend the above convexity constraint to $A(i,:)x + \frac{1}{2}x^T A_2^i x \geq 1$ to enforce that the ith mutex pair must be separated either by a single cut or the combination of two. One may wonder there may exist redundancy caused by three or more cuts, however, we have not encountered such a case in our dataset.

2.4. $H_{sim}()$ – the part-similarity term

We aim to encourage a consistent decomposition of similar parts. As shown in Fig. 3, each candidate cut can separate the shape into two portions and we choose the smaller one as its corresponding part. We measure the similarity of two parts by their contours using the similar method introduced in [15].

For each sampled contour, two matrices M_D and M_θ are designed as descriptors to account for the distance and orientation for each pair of points (p_i, p_j) on the contour, respectively.

$$M_D(i,j) = \|\overrightarrow{p_i} - \overrightarrow{p_j}\|_2, \tag{7}$$

$$M_\theta(i,j) = \angle(\overrightarrow{p_i} - \overrightarrow{p_j}) \in [-\pi, \pi], \tag{8}$$

where $1 \leq i, j \leq n$. Notice that we evenly sample the same number of points on any two contours, say n, in order to compare their similarity.

Then the similarity of two contours, T_1 and T_2, is defined as

$$\phi(T_1, T_2) = (\phi_D(T_1, T_2) + \phi_\theta(T_1, T_2))/2, \tag{9}$$

where

$$\phi_D(T_1, T_2) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\exp\{-\frac{(M_D^1(i,j) - M_D^2(i,j))^2}{(M_D^1(i,j) + M_D^2(i,j))^2\sigma^2}\}, \tag{10}$$

$$\phi_\theta(T_1, T_2) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\exp\{-\frac{(M_\theta^1(i,j) - M_\theta^2(i,j))^2}{\delta^2}\}. \tag{11}$$

σ and δ are two parameters to tolerate the differences of the two contours in distance and orientation, respectively.

In Eq. 1, we define $H_{sim}(i,j) = \phi(T_i, T_j)$ to account for the similarity of a pair of contours derived from cut C_i and cut C_j. The higher the value, the more similar the two contours.

2.5. $I()$ – the cut income term

In order to improve the convexity of the decomposed parts, we employ the cut income term as proposed in [9]. The income of a cut is defined as the concavity of the separated mutex pair of regions by the cut. As shown in Fig. 1 (b), the concavity of mutex pair (blue regions) is $f_S - f_p$ based on Eq. 6. So the income of the cut $\overline{SS_1} = f_S - f_p$.

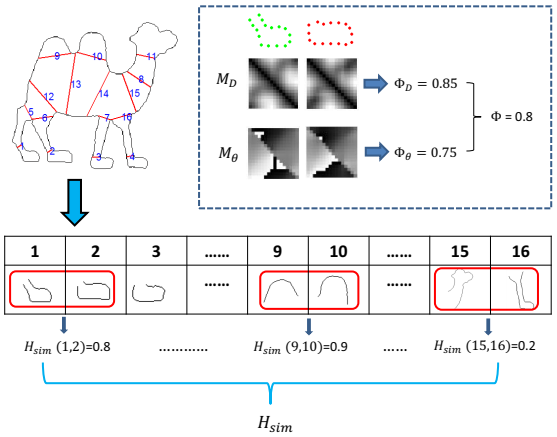

Figure 3. The procedure to construct part-similarity matrix H_{sim}. The red lines are candidate cuts of the camel. We sample the corresponding contour of each cut and compute the similarity between every two contours to get part-similarity matrix H_{sim}.

2.6. Optimization

Now Eq. 1 can be rewritten as

$$\begin{aligned} &\min_x\{L^T x - aI^T x + x^T H x - bx^T H_{sim} x\} \\ &s.t.\ A(i,:)x + \frac{1}{2}x^T A_2^i x \geq 1, \forall i\ , x \in \{0,1\}^n. \end{aligned} \tag{12}$$

To find the optimal solution, we relax x to be continuous ($x_i \in [0,1]$), and the problem becomes a standard quadratically constrained quadratic program(QCQP). We solve it using a trust-region method [3]. It's based on the interior point method, which is aimed to solve nonlinear programming problems.

3. Experiments

To evaluate the proposed method, we conduct two experiments. In the first experiment, we compare our results with human decomposition on the MPEG-7 dataset. Both the quantitative and qualitative evaluations show the improved consistency to human performance compared with other shape decomposition methods. To justify the motivation of this work, we show that object detection on the ETHZ dataset using semantically meaningful parts greatly improve the detection rate than using non-meaningful parts, especially in the case of object articulation.

3.1. Shape Decomposition

3.1.1 Preprocessing

Many objects have curved parts (*e.g.*, curved tails as shown in Fig. 4). These parts greatly affect the convexity of the object shape. In order to entertain the convexity constraint, curved parts will be cut into pieces as shown in Fig. 4 (a),

which is not consistent to human perception. To solve this problem, we adopt the method proposed in [9] to straighten the curved parts as preprocessing. We first detect the tail of the ray by evaluating the "slimness" and average curved angle and then straighten it. Then we decompose the straightened shape (Fig. 4 (c)) and map the cuts back to the original shape (Fig. 4 (b)). In the straightening process, the skeleton is straightened firstly, then shift the points on the contour accordingly. For details, please refer to [9].

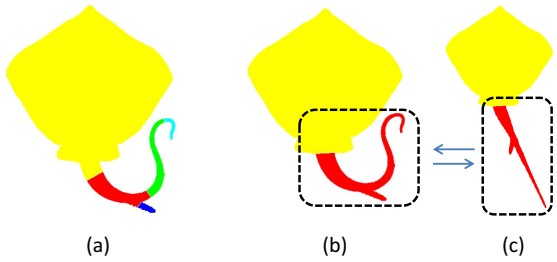

Figure 4. Straightening process. (a) The decomposition result on the original shape without straightening. (b) The mapped result of the straightened result by (c). (c) The decomposition result on the straightened shape.

3.1.2 Experiment method

To verify the consistency of the decomposition by the proposed method with the human decomposition, we choose 20 representative categories which are suitable for decomposition from the MPEG-7 shape dataset as shown in Fig. 5. Each category contains 20 shape instances. We randomly choose 10 shapes from each category and ask 12 people to decompose them according to their own perception.

Figure 5. 20 categories selected from MPEG-7 dataset.

Hence, there are two groups of decompositions, one is by the proposed method and the other is from humans. In each shape category, for each instance i, we define $G(i,1)$ to measure the decomposition similarity between the proposed method and the humans; we define $G(i,2)$ to measure the decomposition consistency among humans.

$$G(i,1) = \frac{1}{n}\sum_{j=1}^{n} g_i(0,j), \tag{13}$$

$$G(i,2) = \frac{1}{n(n-1)}\sum_{j=1}^{n}\sum_{k\neq j} g_i(j,k), \tag{14}$$

in which we index the results by the proposed method by '0' and the results by humans from '1' to 'n', where $n = 12$. $g_i(j,k)$ is the matching score function between the j-th decomposition with the k-th decomposition on the i-th shape instance. And it is defined as

$$g_i(j,k) = \sum_{q=1}^{Q} \frac{A_{ijq}}{A_i} \max_{q'\in\{1,\cdots,Q'\}} F_1(P_{ijq}, P_{ikq'}), \tag{15}$$

where A_{ijq} is the area of part P_{ijq} which denotes the q-th part of the j-th decomposition for shape i. A_i is the total area of shape i. Q and Q' are the total part number of the j-th decomposition and k-th decomposition, respectively. $F_1(P_{ijq}, P_{ikq'}) = 2 \cdot \frac{\text{precision}\times\text{recall}}{\text{precision}+\text{recall}}$ is the $F1$ score, which denotes the matching score of part P_{ijq} and part $P_{ikq'}$. The *precision* is defined as the ratio of intersection area and the area of P_{ijq}; the *recall* is defined as the ratio of intersection area and the area of $P_{ikq'}$. The higher the $F1$ score, the better the matching. If two parts are identical, the *precision* and the *recall* are both 1, and so is the F1 score. If two parts have little intersection, the $F1$ is close to 0 and we define $F1 = 0$ if two parts do not overlap.

3.1.3 Quantitative comparison

Table 1 compares the decomposition consistency between different methods w.r.t human beings (using the average of $G(i,1)$ over instance i). These methods include ACD [7], CSD [14], MNCD [17], PSD [9]. We can see that the proposed method performs the best in 10 object classes out of 20.

To further show that the proposed method is more statistically consistent with human decomposition than the other methods, we conduct the pairwise t-test experiment. The pairwise t-test is on the vectors G. If the decomposition consistency between the proposed method and human is less than the consistency among human beings, the testing result is 1; otherwise 0. From Table 2 we can see that on 17 object classes the decomposition consistency by the proposed method is not significantly less than human, whereas the other methods only get 7, 9, 11 and 14 classes. The t-test is performed at the 5% significance level.

3.1.4 Qualitatively visual comparison

Some visual comparisons between the proposed method and MNCD [17], PSD [9] and human decomposition are shown in Fig. 10. The first column is the result from MNCD and the second is from PSD. Both aim to decompose the shape into natural parts. The third column is the results of ours and the last one is the human decomposition.

As can be seen from the top four rows when considering the part-similarity, our method is able to decompose

	apple	bat	beetle	bird	Bone	camel	carriage	cattle	chicken	crown
ACD [7]	**0.9933**	0.7554	0.8692	0.7093	0.8861	0.6950	0.7708	0.6653	0.7363	0.9146
CSD [14]	0.9864	0.7405	0.8865	0.7987	0.8537	0.7724	0.7919	0.9060	0.7656	0.9575
MNCD [17]	0.9759	**0.7665**	0.8952	0.8398	0.9524	0.6371	0.7841	0.8553	0.7506	0.9580
PSD [9]	0.9893	0.7645	0.8840	0.8802	0.9517	0.6928	**0.7961**	0.9044	0.8131	0.9640
Ours	0.9847	0.7571	**0.9091**	**0.8846**	**0.9537**	**0.7977**	0.7809	**0.9215**	**0.8425**	**0.9647**
	device1	elephant	flatfish	fork	frog	Glas	lmfish	octopus	ray	tree
ACD [7]	0.9784	0.7923	0.9879	0.7742	0.6878	**0.9799**	**0.6622**	0.8697	0.9568	0.6887
CSD [14]	0.8997	0.8462	0.9461	0.8361	0.8209	0.8565	0.6431	0.9306	0.9303	0.6568
MNCD [17]	0.8221	0.8086	0.9908	0.8560	0.7736	0.9501	0.5945	0.9291	0.8882	0.7088
PSD [9]	0.8813	**0.8660**	**0.9910**	0.8520	**0.8232**	0.9493	0.6099	**0.9531**	0.9630	**0.7228**
Ours	**0.9861**	0.8586	0.9909	**0.8850**	0.7997	0.9136	0.6024	0.9331	**0.9634**	0.6917

Table 1. F-measure decomposition consistency result between our method and human beings comparing with ACD [7], CSD [14], MNCD [17] and PSD [9]. Our method outperforms others in 10 categories out of 20.

	ACD	CSD	MNCD	PSD	Ours
apple	0	0	0	0	0
bat	0	0	0	0	0
beetle	0	0	0	0	0
bird	1	1	1	0	0
Bone	1	1	0	1	0
camel	1	0	1	1	0
carriage	0	0	0	0	0
cattle	1	1	1	1	0
chicken	1	1	1	0	0
crown	1	0	0	0	0
device1	1	1	1	1	0
elephant	1	1	1	0	0
flatfish	0	1	0	0	0
fork	1	1	0	1	0
frog	1	0	0	0	0
Glas	0	1	0	0	1
lmfish	1	0	1	1	1
octopus	1	0	0	0	0
ray	0	1	1	0	0
tree	1	1	1	0	1
all	**7**	**9**	**11**	**14**	**17**

Table 2. Pairwise t-test of our result comparing with ACD [7], CSD [14], MNCD [17] and PSD [9]. The results by our method are consistent with human decomposition statistically in 17 categories out of 20.

shape into similar parts. This makes the decomposed parts more consistent with human perception. The fourth to seventh rows demonstrate the advantage of introducing A_2 constraint to the optimization objective function. It serves to remove redundant cuts. The last row is a failure case, which will be discussed in Section 3.3.

3.1.5 Evaluation of parameters

There are three free parameters in our formulation, ε, a and b. ε measures the tolerance to the concavity. a and b weight the cut income and part-similarity, respectively. In our experiment, we set $\varepsilon = 0.05$, $a = 0.1$ and $b = 0.1$.

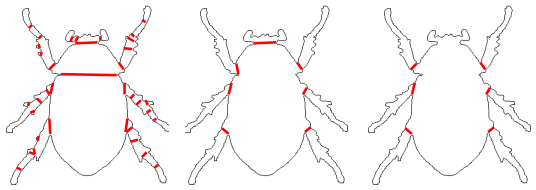

Figure 6. Evaluation of ε. The decomposition results when $\varepsilon = 0.01, 0.05$ and 0.1. In our experiment, we choose $\varepsilon = 0.05$.

Figure 7. Evaluation of b. The decomposition results when $b = 0, 0.05$ and 0.1. In our experiment, we choose $b = 0.1$.

Fig. 6 shows the decomposition results under different ε. The less ε is, the more convex the decomposed parts. So when ε increases, the decomposition will ignore smaller concave parts due to local distortions and get a relatively robust result.

Fig. 7 shows the decomposition results at different values of b. A larger b encourages more cuts with similar parts, hence, can get more semantic decomposition.

In addition, for the number of Morse functions, the more we choose, the more robust the method can be when dealing with rotation. In our experiment, the number is set 16 which is also adopted in [9, 14, 17].

3.2. Object Detection

To demonstrate the advantage of semantic parts, we conduct a part-based object detection experiment. We decompose the shapes to generate semantic parts (Fig. 8 (a)), and learn a set of part templates by clustering obtained part instances. Then we detect objects by matching the part templates on the testing images. In comparison, we simply replace the semantic parts with a set of random parts (not semantically meaningful as shown in Fig. 8 (b)), while the rest of the experiment settings are kept the same. A random part is a continuous contour fragment, which starts from a random location on the shape. The lengths of random parts are comparable to those of semantic parts. We compare the object detection rates on ETHZ dataset between the two sets of parts.

We now describe the details of our part-based object detection method. We first extract the edge map on a testing image using the method introduced in [16]. The part templates are matched on the edges to vote for the object centers in the generalized Hough transform framework [2]. In the voting space, objects are detected where the vote score is above a threshold. In particular, we use shape descriptors in [15] to measure similarities between edges and templates. Beam search [10] is adopted to speed up the matching process.

Table 3 summarizes the detection rates at 0.3 and 0.4 false positive per image (FPPI). We can see that the semantic parts can boost the performance on all categories, which demonstrates the representative power of the proposed semantic parts. In addition, the improvement on the giraffe and swan categories is over 15%. In these two categories, articulation is significant, occurring at the necks and legs. The semantic parts capture the anatomy structure and keep rigid in articulation; however, the random parts may change drastically. So the former is much more robust to handle this situation.

3.3. Limitations and Discussions

Fig. 9 shows some decomposition results of shapes with holes. Because the cup handle is a curved branch and cannot be straightened by the preprocessing method, redundant cuts are generated.

The last row of Fig. 10 shows a failure example of the proposed method – although our method generates shorter cut length than human and more similar parts, the decomposition is not consistent with human perception. This failure is due to the lack of object level semantic information. It shows the limitation of shape decomposition only based on generic perception rules.

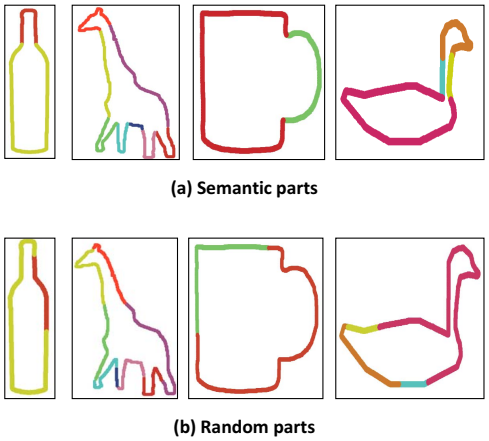

Figure 8. Semantic parts and random parts. Parts are represented by different colors.

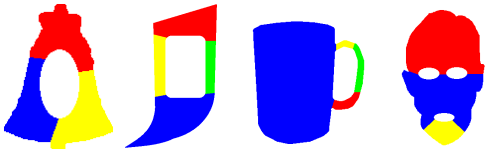

Figure 9. Decomposition results of shapes with holes.

4. Conclusion

In this paper, we propose a method to decompose a shape into semantic parts. Apart from three existing perception rules, we propose a part-similarity rule to encourage consistent cuts for similar parts. By jointly considering these perception rules, we formulate the shape decomposition problem as a quadratically constrained quadratic program problem and solve it by a trust-region method. Extensive experiments on the MPEG-7 shape dataset validates our approach is consistent with human perception. An object detection experiment is also conducted on the ETHZ dataset to demonstrate the advantage of the semantic parts over the non-meaningful parts for shape representation.

Acknowledgement

This work was supported in part by the National Science Foundation of China (61103087, 61121002, 61210005, 91120004, 61272027, 61231010).

References

[1] J. Aleotti and S. Caselli. A 3D shape segmentation approach for robot grasping by parts. *Robotics and Autonomous Systems*, 60(3):358–366, 2012.

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

[3] R. H. Byrd, J. C. Gilbert, and J. Nocedal. A trust region method based on interior point techniques for nonlinear

	Bottles	Giraffes	Mugs	Swans	Mean
Random part	0.836/0.855	0.319/0.357	0.742/0.788	0.758/0.758	0.664/0.69
Semantic part	0.891/0.9	0.527/0.549	0.803/0.864	0.909/0.909	0.783/0.806

Table 3. Comparison of detection rates at 0.3/0.4 FPPI using different part templates. Objects in the apple logo category include two divided components so it is hard to generate their random parts, we do not use this category in the experiment.

Figure 10. Comparison results of ours with MNCD [17], PSD [9], and human. By considering the part-similarity and pruning the redundant cuts, our decompositions are more visually consistent with human perception.

programming. *Mathematical Programming*, 89(1):149–185, 2000.

[4] H. Doraiswamy and V. Natarajan. Output-sensitive construction of reeb graphs. *Visualization and Computer Graphics*, 18(1):146–159, 2012.

[5] V. Ferrari, F. Jurie, and C. Schmid. From images to shape models for object detection. *IJCV*, 87(3):284–303, 2010.

[6] V. Ferrari, T. Tuytelaars, and L. Van Gool. Object detection by contour segment networks. In *ECCV*, pages 14–28. Springer, 2006.

[7] R. Gopalan, P. Turaga, and R. Chellappa. Articulation-invariant representation of non-planar shapes. In *ECCV*, pages 286–299. Springer, 2010.

[8] D. D. Hoffman and W. Richards. Parts of recognition. *Cognition*, 18:65–96, 1984.

[9] T. Jiang, Z. Dong, C. Ma, and Y. Wang. Toward perception-based shape decomposition. In *ACCV 2012*, pages 188–201. Springer, 2013.

[10] G. Kim and E. P. Xing. On multiple foreground cosegmentation. In *CVPR*, pages 837–844. IEEE, 2012.

[11] L. J. Latecki and R. Lakämper. Convexity rule for shape decomposition based on discrete contour evolution. *CVIU*, 73(3):441–454, 1999.

[12] L. J. Latecki, R. Lakamper, and T. Eckhardt. Shape descriptors for non-rigid shapes with a single closed contour. In *CVPR*, volume 1, pages 424–429. IEEE, 2000.

[13] J.-M. Lien and N. M. Amato. Approximate convex decomposition of polygons. In *Proceedings of the twentieth annual symposium on Computational geometry*, pages 17–26. ACM, 2004.

[14] H. Liu, W. Liu, and L. J. Latecki. Convex shape decomposition. In *CVPR*, pages 97–104. IEEE, 2010.

[15] T. Ma and L. J. Latecki. From partial shape matching through local deformation to robust global shape similarity for object detection. In *CVPR*, pages 1441–1448. IEEE, 2011.

[16] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *PAMI*, 26(5):530–549, 2004.

[17] Z. Ren, J. Yuan, C. Li, and W. Liu. Minimum near-convex decomposition for robust shape representation. In *ICCV*, pages 303–310. IEEE, 2011.

[18] W. Shen, X. Bai, R. Hu, H. Wang, and L. Jan Latecki. Skeleton growing and pruning with bending potential ratio. *Pattern Recognition*, 44(2):196–209, 2011.

[19] K. Siddiqi, K. J. Tresness, and B. B. Kimia. Parts of visual form: Psychophysical aspects. *Perception*, 25:399–424, 1996.

[20] M. Singh, G. D. Seyranian, and D. D. Hoffman. Parsing silhouettes: The short-cut rule. *Perception & Psychophysics*, 61(4):636–660, 1999.

[21] L. L. Walker and J. Malik. Can convexity explain how humans segment objects into parts? *Journal of Vision*, 3(9):503–503, 2003.

[22] L. Zhu, Y. Chen, A. Torralba, W. Freeman, and A. Yuille. Part and appearance sharing: Recursive compositional models for multi-view. In *CVPR*, pages 1919–1926. IEEE, 2010.