Toward Perception-based Shape Decomposition

Tingting Jiang, Zhongqian Dong, Chang Ma, Yizhou Wang

National Engineering Lab for Video Technology,
Key Lab of Machine Perception(MoE),
School of EECS, Peking University, Beijing, China

Abstract. The aim of this work is to decompose shapes into parts which are consistent to human perception. We propose a novel shape decomposition method which utilizes the three perception rules suggested by psychological study: the Minima rule, the Short-cut rule and the Convexity rule. Unlike the previous work, we focus on improving the convexity of the decomposed parts while minimizing the cut length as much as possible. The problem is formulated as a combinatorial optimization problem and solved by a quadratic programming method. In addition, we consider the curved branches which introduce "false" concavity. To solve this problem, we straighten the curved branches before shape decomposition which makes the results more consistent with human perception. We test our approach on the MPEG-7 shape dataset, and the comparison results to previous work show that the proposed method can improve the part convexity while keeping the cuts short, and the decomposition is more consistent with human perception.

1 Introduction

Part-based shape representation is a popular representation for objects in the community of computer vision (*e.g.*, [1]). However, there is a question always hovering around us – *What is an optimal set of parts for representing an object (shape)?* In this paper, we propose to address this question by referring to the clues adopted by human perception, as human beings are apt at identifying shape parts very easily and quickly, as well as generally consistently.

Psychological studies show that when presenting a single shape, people tend to use generic and simple perception rules to decompose it into parts, *e.g.*, the Minima rule [2], the Short-cut rule [3] and the Convexity rule [4, 5]. *The Minima rule* considers the signed curvature of the shape boundary, and enforces that a shape is divided at places where the curvature is local minimum. *Convexity* is also an important perceptual clue to determine visual parts. Both rules reflect the "saliency" of a part *w.r.t.* the global shape and the easiness to extract and identify parts for shape perception [6]. The *Short-cut rule* takes the cut length as a constraint and optimizes the decomposition by minimizing the total cut length. This shows that human vision prefers to use the shortest possible cuts to parse shapes, leading to a "compact" part-based representation.

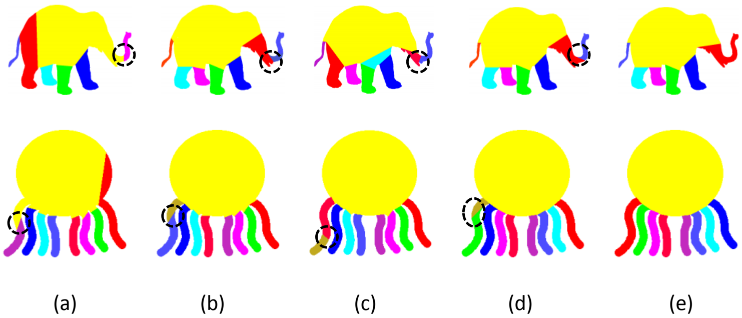

Fig. 1. Comparison of our decomposition results to previous work. (a) [7]'s result. (b) [8]'s result. (c) [11]'s result. (d) Our result without straightening. (e) Our result with straightening.

In this paper, we propose a novel shape decomposition method by jointly considering the three generic perception rules to find an optimal shape decomposition consistent to human perception. Specifically, the problem is formulated as selecting an optimal subset of candidate cuts, from which object parts are derived. The Minima rule is used to propose prospective candidate cuts. The cut length is the cost of choosing one cut which corresponds to the Short-cut rule. On the other hand, the benefit of choosing one cut is designed as the improvement of the convexity of generated parts which corresponds to the convexity rule. Therefore, the three generic perception rules are integrated naturally into a quadratic integer programming formulation that optimizes the totality of the perception criterion. To further improve the decomposition, we first detect curved branches and apply a straightening process before the decomposition. This can avoid the superfluous cuts due to the "false concavity" of curved branches (shown as the highlighted parts of Fig. 1(a)-(d)) and make the decomposition more consistent to human perception.

Besides the shape decomposition method, another contribution of the paper is that we propose a quantitative measure for the first time to evaluate decomposition results. Most of previous methods justified improvement by showing a limited number of selected good decomposition results and compared with the results from other methods side by side only by eyeballing. This kind of evaluation methodology is subjective and could be inaccurate. In this paper we propose a quantitative evaluation measure based on statistical data from a considerable amount of decomposition results. This makes the comparison more convincing.

To verify our method, we design an experiment which asks people to decompose the given shapes manually into perceptual meaningful parts. By comparing the human decomposition to different results generated by our algorithm as well as previous work, it can be seen that our results are closer to human decomposition which proves the consistency between our method and human perception.

Related Work Most previous work on generating shape parts can be classified into two strategies: one is "bottom up" strategy which is grouping small shape elements into large shape parts [9], and the other is "top down" strategy which is shape decomposition. For the former strategy, people use bottom-up grouping method to learn parts as hierarchical shape vocabularies from a large number of shape instances, e.g. [9, 10]. This type of approaches consider the joint statistics between the object and curve fragments at different levels of hierarchies, whereas ignore shape perceptual properties such as convexity and cut length at all.

The latter strategy exploits perceptual cues to decompose shapes, it is in the same vein of the proposed method. Based on different cues, computer vision scientists have developed various optimization algorithms for shape decomposition. For example, based on convexity, Gopalan *et al.* [7] proposed an algorithm to do approximate convex decomposition. Liu *et al.* [8] considered both convexity and the Short-cut rule to optimize the shape decomposition. Based on their work, Ren *et al.* [11] further encoded the Minima rule and the number of shape parts into the objective function during optimization.

Besides these two classes, there are some previous work using symmetry [12] and Relatability [13] to find shape parts.

In this paper, the way we formulate the problem is different from previous work, including [8, 7, 11]. Both [8] and [7] only applied one or two rules. Although [11] considered all three rules, their focus is to minimize the number of parts which has overlap with the Short-cut rule, and to optimize the "visual naturalness" which is different from our goal. In addition, they did not provide a rigorous way to define and evaluate the "visual naturalness". Compared to the previous work, we optimize the totality of the perception criterion by encoding the contribution of each cut as "cut-income" and formulating the problem as a quadratic programming problem, instead of requiring each part's convexity simply above a threshold as [8, 11]. Thus our decomposition results will be sensitive to the improvement of convexity and consistent to human perception.

The rest of the paper is organized as follows. Section 2 formulates the shape decomposition problem as an optimization problem. Section 3 reviews the related preliminary work. Section 4 introduces our approach and Section 5 shows the experimental results. Finally, Section 6 concludes the paper.

2 Problem Formulation

Given a planar shape S which is simply connected, a partition of S is defined as

$$S = \bigcup_i P_i, \;\; s.t., \forall P_i, \;\; P_i \subset S; \;\; \forall i, j, \;\; P_i \bigcap P_j = \emptyset, \tag{1}$$

which means that S is composed of several parts $\{P_i\}$ and these parts do not overlap each other. On the other hand, a partition of S is associated with a set of cuts $\{C_j\} = \{\overline{p_{j1}p_{j2}}\}$ where each cut is a line segment $\overline{p_{j1}p_{j2}}$ and both points p_{j1} and p_{j2} lie on the boundary of S. These cuts do not intersect each other. The boundary of P_i is composed of the boundary of S and a subset of $\{C_j\}$.

Based on the above notations, we can explain the three rules as follows:

- The Minima rule [2] suggests that the cut points $\{p_{j1}, p_{j2}\}$ are located at the points where the curvature is local minimum.
- The Short-cut rule [3] suggests to minimize the total length of the cuts, i.e., $\min \sum_j L(C_j)$ where $L(C_j)$ denotes the length of cut C_j which is usually computed as the Euclidean distance between points p_{j1} and p_{j2}.
- The Convexity rule [4, 5] suggests to maximize the convexity of parts or minimize the concavity of parts. Based on this rule, Rosin [14] proposed a weighted average convexity to evaluate the decomposition quality as follows,

$$\text{Convexity}(\{P_i\}) = \sum_i \frac{A_i}{A} \text{Convexity}(P_i). \tag{2}$$

where A_i denotes the area of P_i and A is the area of shape S or $A = \sum_i A_i$.

Based on these rules, the goal of shape decomposition can be formulated as

$$\min(\sum_j L(C_j) + \sum_i \frac{A_i}{A} \text{Concavity}(P_i)) \tag{3}$$

s.t. cut points have local minimal curvatures and cuts do not intersect.

3 Preliminary Work

Given the above problem formulation, the remaining questions include : (i) how to measure *Concavity*(P_i)? (2) how to encourage the cut points $\{p_{j1}, p_{j2}\}$ to have the local minimal curvatures? (3) how to optimize the objective function based on the measurements? We will introduce the preliminary work related to these questions in the following sections.

3.1 Convexity/Concavity Measurement

There are several choices for the measurement of convexity. One classical definition of convexity is the ratio of the area of the part to the area of its convex hull [14]. But this convexity measure is criticized due to its insensitivity to deep (but thin) protrusions of the boundary because it is area-based.

Inner Distance Recently, the inner distance (ID) is becoming a popular measure for convexity [15, 7] because it is sensitive to the deep protrusions. The convexity of a part P_i is defined as the minimal ratio of the Euclidean distance (ED) over ID of a pair of points within this part, and the concavity of P_i is defined as follows,

$$\text{Convexity}(P_i) = \min_{p,q \in P_i} \text{ED}(p,q)/\text{ID}(p,q), \text{Concavity}(P_i) = 1 - \text{Convexity}(P_i). \tag{4}$$

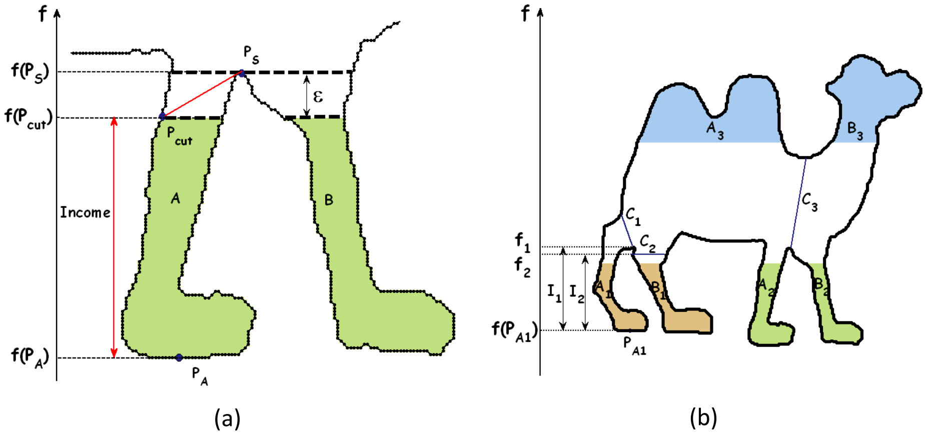

Fig. 2. (a) Illustration of a mutex pair of regions (A and B), a candidate cut (red line) and its income. f is the Morse function. Point p_s is the saddle point which corresponds to the mutex pair (A and B). p_{cut} is the cut point. p_A is the lowest point in part A w.r.t. the direction of Morse function f. $f(p_{cut})$ and $f(p_A)$ are the Morse function values of points p_{cut} and p_A. The income of the red cut is $f(p_{cut}) - f(p_A)$. (b) One cut can satisfy two mutex pairs of regions. Cut C_3 can satisfy two mutex pairs of regions (A_2 and B_2, A_3 and B_3). One mutex pair of regions A_1 and B_1 can be separated by two different cuts C_1 and C_2 with income I_1 and I_2 respectively.

Morse Function Besides the above measure, Liu *et al.* proposed a new measure [8] for concavity. By this measure, for each point pair (p, q) within part P_i, their concavity $concave(p, q)$ is defined by a path which can minimize the maximal perpendicular distance between the line passing (p, q) and the projected contour points between p and q on this path w.r.t. all Morse functions. Based on this, the concavity for a part P_i is defined as follows:

$$\text{concave}(P_i) = \max_{p,q \in P_i} \text{concave}(p, q). \tag{5}$$

Although this definition of concavity is different from Eqn. 4, we can prove that the two definitions are inherently consistent for shapes without holes:

Theorem : Given a shape S without holes, for any point pair $(p, q) \in S$, the path R which corresponds to the inner distance between p and q is also the path which can minimize the concavity as defined in Eqn. 5.
The proof is provided in the supplemental material.

3.2 Mutex Pair and Candidate Cuts

The shape decomposition problem can be also viewed as a selection problem. Since there are infinite cuts inside a shape, the goal is to select a subset of cuts which can optimize an objective function. How to propose qualified candidate cuts is a challenging problem. Liu *et al.* [8] proposed a way to generate candidate

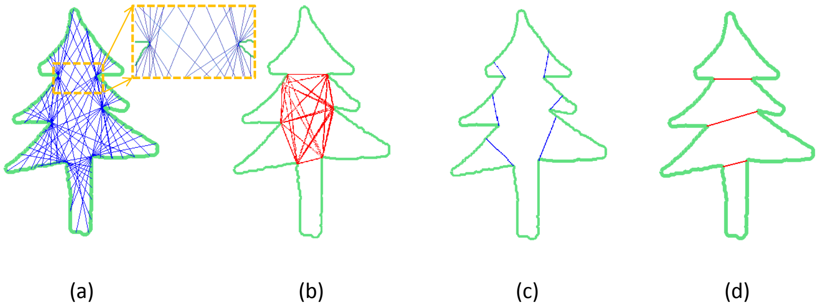

Fig. 3. Comparisons of the candidate cuts by previous work and our new cuts as well as the resulted shape decompositions. (a) Candidate cuts by [8]. (b) New candidate cuts. (c) Shape decomposition by [8]. (d) Our decomposition result.

cuts. The idea is to find pairs of components which cannot be kept together otherwise the concavity of the part containing both components will be high. Each pair of such components is defined as a "mutex pair" of regions. Specifically, it is a pair of regions A and B with

$$m(A, B) = \min_{p \in A, q \in B} \text{concave}(p, q) \tag{6}$$

above a threshold ϵ (See Fig. 2(a)). Given a fixed threshold ϵ, let MP denote all the mutex pairs to be separated and $|MP| = n_{mp}$. The motivation of generating candidate cuts is to separate these mutex pairs of regions. Fig. 3 (a) shows an example of generated candidate cuts by this method. Due to the limited number of Morse functions being sampled (16 directions in the experiments), the "best" cut which can separate the bottom piece is missing.

4 Our Approach

4.1 New Candidate Cuts

In Liu *et al.* [8]'s work, they only consider the Convexity rule to propose a set of candidate cuts $\mathbf{C}_L$. However, the Minima rule can also help to propose useful candidate cuts. We add a new set of candidate cuts $\mathbf{C}_M$ *s.t.* both cut points of each new cut have local minimum curvatures. Fig. 3 (b) shows a set of new cuts generated by this rule. By combining $\mathbf{C}_L$ and $\mathbf{C}_M$, it can be seen that the set of candidate cuts $\mathbf{C}_p = \mathbf{C}_L \cup \mathbf{C}_M$ is more comprehensive and complete which will improve the final solution (See Fig. 3 (c) and (d)).

4.2 Cut Income

If the length of a cut is thought as the cost for choosing this cut, the contribution of a cut for reducing the concavity of parts can be viewed as its "income". For each mutex pair mp and each cut C, let $I(mp, C)$ denote the income of C for

mp. For example, in Fig. 2(a), without the red cut, the concavity of A and B is $f(p_s) - f(p_A)$ by the definition of Eqn. 5. With the red cut, the concavity of the left part becomes $f(p_s) - f(p_{cut})$. So the reduction of the concavity is $f(p_{cut}) - f(p_A)$ which is the income of this cut for mutex pair A and B.

A cut can satisfy multiple mutex pairs and a mutex pair can also be satisfied by multiple cuts (Fig. 2(b)). Let mp denote a mutex pair and $\mathbf{C}(mp)$ denote the set of candidate cuts which can satisfy mp. Let $\mathbf{M}(C)$ denote the set of mutex pairs which can be satisfied by C. So the real income of a candidate cut is determined by the mutex pairs it satisfies in the final shape decomposition. However, which mutex pairs it can really satisfy is unknown before the decomposition is finalized. If multiple cuts satisfying mp are chosen in the final solution, only the cut C^* which maximizes $I(mp, C)$ can get its income $I(mp, C^*)$ and other cuts have no income. For example, in Fig. 2(b), both cuts C_1 and C_2 can satisfy mutex pair A_1 and B_1, but the income of C_1 is larger. Thus if both cuts are chosen, only C_1 makes income for this mutex pair.

We estimate the expected income of each candidate cut by stochastic analysis. Here we make an assumption that for any final solution, whether one cut is chosen or not is independent of other cuts. So the probability of choosing one candidate cut is 1/2. For each mutex pair, we rank $\mathbf{C}(mp)$ by $I(mp, C)$ and get a sequence of candidate cuts as $C_1^{mp}, C_2^{mp}, \ldots, C_k^{mp}$ *s.t.* $I(mp, C_1^{mp}) \geq I(mp, C_2^{mp}) \geq \ldots \geq I(mp, C_k^{mp})$. Let $r(mp, C)$ be the ranking of the cut C w.r.t. mp. For example, $r(mp, C_1^{mp}) = 1$. If a subset of $\mathbf{C}(mp)$ is chosen in the final solution, only the cut with maximum $I(mp, C)$ counts, the others do not make any income for this mutex pair. The probability for cut C being counted for mutex pair mp depends on its ranking $r(mp, C)$. If $r(mp, C) = 1$, the probability for C being counted for mp is 1/2. Because once C is chosen, it will be counted for mp no matter whether any other cut is chosen. The probability of choosing C is 1/2. In general, it is easy to show the probability for C being counted for mp is

$$Pr(C \text{ counted for}\, mp) = Pr(C \text{ is 1st chosen in ranked } \mathbf{C}(mp)) = \frac{1}{2^{r(mp,C)}}. \tag{7}$$

It is expected that the summation of this probability over all candidate cuts which can satisfy this mutex pair is 1. If cut $I(mp, C) = 0$ or C does not make any income for mp, $Pr(C \text{ is counted for } mp) = 0$. Based on this observation, we can estimate the expected income of a cut C as follows:

$$\overline{I(C)} = \sum_{mp \in \mathbf{M}(C)} \frac{1}{2^{r(mp,C)}} I(mp, C). \tag{8}$$

4.3 Optimization

Assume that there are n candidate cuts in the proposed candidate cut set $\mathbf{C}_p = \{C_j\}_{j=1}^n$. The final decomposition chooses a subset of $\mathbf{C}_p$, denoted by $\mathbf{C}^*$. With the expected income for each candidate cut, we can reformulate Eqn. 3 as:

$$\min_{\mathbf{C}^*}\{\sum_{C_j \in \mathbf{C}^*} [L(C_j) - \overline{I(C_j)}]\} \tag{9}$$

because minimization of the part concavity is equivalent to maximization of the reduction of concavity, i.e., the expected income of chosen cuts.

Design a binary vector $\mathbf{x}$ *s.t.*: $\mathbf{x}_j = 1 \iff C_j \in \mathbf{C}^*$. Let vector $\mathbf{L}$ represent the cut length of $\mathbf{C}_p$ *s.t.* $\mathbf{L}_j = L(C_j)$. Let vector $\mathbf{I}$ be the expected income of $\mathbf{C}_p$ *s.t.* $\mathbf{I}_j = \overline{I(C_j)}$. Design a penalty matrix $\mathbf{H}_{n\times n}$ *s.t.* if C_j and C_k intersects $\mathbf{H}(j,k) = +\infty$. So Eqn. 9 can be rewritten as

$$\min_{\mathbf{x}} \mathbf{L}^T\mathbf{x} - a\mathbf{I}^T\mathbf{x} + \mathbf{x}^T\mathbf{H}\mathbf{x} \qquad s.t.\ \ \mathbf{A}\mathbf{x} \geq 1,\ \ \mathbf{x} \in \{0,1\}^n, \tag{10}$$

where $\mathbf{A}$ is a matrix of size $n_{mp} \times n$. It denotes the relationship between the mutex pairs $MP = \{mp_k\}_{k=1}^{n_{mp}}$ and the candidate cuts $\mathbf{C}_p$. If a mutex pair mp_k can be satisfied by cut C_j, then $\mathbf{A}(k,j) = 1$, otherwise it is zero. a is parameter to adjust the impact of cuts' income.

The above formulation considers the cut length, the expected cut income, the intersection of cuts and the mutex pairs to be separated. If we relax $\mathbf{x}$ to be $\mathbf{x}_i \in [0,1]$, this problem becomes a standard quadratic programming problem:

$$\min_{\mathbf{x}} \mathbf{x}^T\mathbf{H}\mathbf{x} + (\mathbf{L}^T - a\mathbf{I}^T)\mathbf{x} \quad s.t.\ \ \mathbf{A}\mathbf{x} \geq 1,\ \ \mathbf{x} \in [0,1]^n. \tag{11}$$

In Eqn. 11, H might not be positive definite and thus the function might be nonconvex. We use a global solver called "bmibnb" provided by YALMIP [16]. It is based on a standard spatial branch-and-bound strategy. After solving Eqn. 11, we can get a soft assignment of $\mathbf{x}$ and then incrementally choose the cut with maximal $\mathbf{x}_j$ until all mutex pairs are satisfied. Because we considered the cut intersections by H, there will be no intersecting cuts chosen in our final decomposition while this is possible in [8].

4.4 Straighten the Curved Branches

If the above method is applied on a shape with curved branches directly, it will be cut into small pieces. For example, in Fig. 4(f), the tail of the ray is cut into four pieces. To keep the decomposition of the curved branches consistent with human perception, we design a preprocessing step to straighten these branches before the shape decomposition. The idea is to first detect curved branches and then straighten them. Similar work has been done in [17] for shape classification. But they only detect the curved branches without straightening and assume that the curved branch must be attached to a single base structure. Here we do both detection and straightening for curved branches without the single base assumption. Notice that we cannot take the detected curved branches as the expected final parts because one curved branch could contain multiple parts. For example, in Fig. 6, the elephant's nose can be detected as a curved branch but this branch can be further decomposed as two parts (see final column).

To detect the curved branches, the skeleton of the shape is extracted by [18] and represented as a tree shown in Fig. 4(b). Each possible curved branch corresponds to a subtree. For a subtree, we find the longest path from the root in this subtree and define it as the "trunk". For example, in Fig. 4(b), the red curve

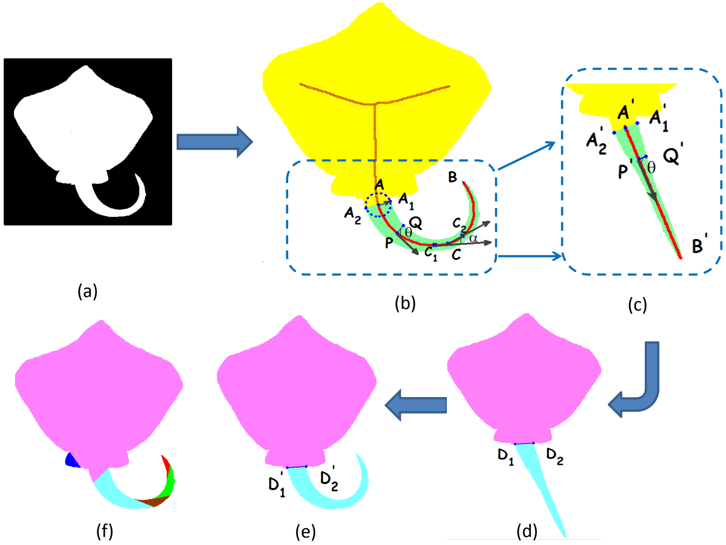

Fig. 4. Illustration of straightening process. (a) The original shape. (b)Detection of curved branches. Red curve AB is the trunk. Green area is the corresponding curved branch. A_1 and A_2 are two points on the shape contour which are equally distant from A. C_1 and C_2 are neighboring points of C on the trunk. α denotes the change of the direction of the trunk at point C. P is the closest trunk point to Q. θ is the angle between $\overline{PQ}$ and the tangent direction of the trunk at P. (c) Straightened green part. $\overline{A'B'}$, P', Q' correspond to AB, P, Q respectively in (b). (d) Decomposition on the straightened shape. $\overline{D_1D_2}$ is the cut. (e) Decomposition on the original shape. $\overline{D_1'D_2'}$ is the corresponding cut to $\overline{D_1D_2}$.(f) Decomposition without straightening.

is the "trunk" for the subtree with the root node A. A_1 and A_2 are two points on the shape contour which are equally distant from A. The green area is the corresponding part to the red trunk. Let the trunk be represented by a set of points. For each point on the trunk, connect it to the neighboring two points and get two line segments. The angle between the two line segments is defined as the change of the direction at this point. For example, in Fig. 4(b), the change of the direction at point C is α which is the angle between $\overrightarrow{C_1C}$ and $\overrightarrow{CC_2}$.

For each trunk, the following three criteria shown in Fig. 4(b) are used to check whether the corresponding branch is curved and need straightening: (1) "slimness" which can be measured by the ratio of the length of the trunk over the average width of the corresponding branch along the trunk; (2) curved angle which is the maximum change of the direction of the trunk; (3) the ratio of the branch area over the total area of the shape.

Next a straightening procedure is applied on the detected curved branches. For each curved branch, first straighten its corresponding trunk and then use it as a reference. In Fig. 4(c), $\overline{A'B'}$ is a straight line segment corresponding to trunk AB in Fig. 4(b). For each point Q on the branch contour, find its corresponding point Q' on the straightened shape as follows. First, find its closest point P on the trunk. Denote the angle between line segment $\overline{PQ}$ and the tangent direction of the trunk at P as θ. Second, find the corresponding point P' of P on $A'B'$. Third, map Q to Q' by keeping the angle between line segment $\overline{P'Q'}$ and $\overline{A'B'}$ as θ and the length of $\overline{P'Q'}$ equal to $\overline{PQ}$. Therefore, Q' can be located.

After straightening, the proposed decomposition method above can be applied and the curved branches will not be cut into pieces (Fig. 4(d)). Last, the generated cut set on the straighten shape can be mapped back to the original shape as shown on Fig. 4(e).

5 Experimental Results

To evaluate the proposed method on 2D shape decomposition, we choose 20 categories from the MPEG-7 shape dataset [19] as Fig. 6 shows. For each category, there are 20 shapes. In total, there are 20×20 shapes for evaluation.

We propose two measures to evaluate the performance of the decomposition result: cut length and convexity which correspond to the first two items in the designed objective function Eqn. 10. The cut length is measured by the ratio of the total cut length over $\mathfrak{D}$ which is the maximum distance between points of the shape. Specifically, we choose 16 Morse functions and set the threshold ϵ for generating mutex pairs as $0.05\mathfrak{D}$. These settings are same for both our method and [8]. The parameter a in our method is set as 0.1. For straightening, the threshold for slimness is 13, the curve angle is expected between $20 - 75^\circ$, and the maximal ratio of the part area over the total shape area is set as 0.2.

5.1 Shape Decomposition

To test the performance of our method, we compare it to [8], [7] and [11]. Fig. 5 shows the comparison result. For each shape, we first obtain the four different decomposition results and then evaluate the cut length and the convexity by Eqn. 2. For each category, we calculate the mean and standard variance of the cut length across 20 instances for each method respectively. Similarly, the convexity is computed based on inner distance. From Fig. 5, it can be seen that our cut length(pink and red) are shorter than the other three methods (green, yellow and cyan) in most cases on average while our convexity is about the same as those methods. This proves that our method can achieve same convexity with shorter cuts compared to previous work. Further, straightening improves the decomposition results for several classes including camel, elephants, octopus and ray. The cut length is reduced for these classes because the superfluous cuts are removed by straightening. But the convexity does not change too much. Fig. 6 and Fig. 1 show some decomposition examples of the four methods. From these examples, we can see that our results can achieve a good balance between convexity and cut length. Compared to [8], ours can reduce the cut length while keeping a good convexity by the added new candidate cuts. For example, in Fig. 1, the elephant's nose is cut better by our method because the elephant is first straightened and won't introduce concavity during shape decomposition.

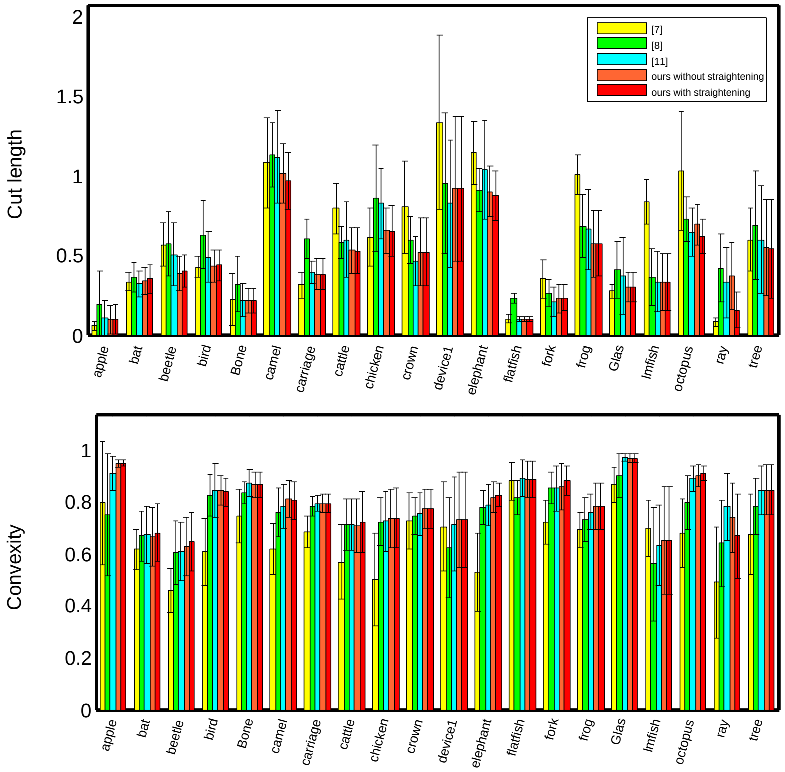

Fig. 5. Comparisons of the cut length and part convexity of the shape decomposition results from [7](yellow), [8](green), [11](cyan), our method without(pink) and with(red) straightening. The black lines denote the standard variance.

5.2 Human Perception

To verify whether the decomposition results are consistent with human perception, we conduct an experiment which asks people to decompose the shape examples into perceptually meaningful parts. For each of the 20 categories from the MPEG-7 shape dataset, we select 10 shapes for testing. And for each selected shape, 12 people decompose it manually. The cuts provided by the participants are taken as the set of "ground truth" cuts, called $\mathbf{C}^+$. To measure the discrepancy between our decomposition result $\mathbf{C}^*$ and the "ground truth" cuts $\mathbf{C}^+$, we define the distance between two cuts $C_1 = \overline{p_1 p_2}$ and $C_2 = \overline{p_3 p_4}$ as follows:

$$D_1(C_1, C_2) = \min\{\mathrm{ED}(p_1, p_3) + \mathrm{ED}(p_2, p_4), \mathrm{ED}(p_1, p_4) + \mathrm{ED}(p_2, p_3)\}. \qquad (12)$$

The distance between one cut C and one cut set $\mathbf{C}$ is defined as :

$$D_2(C, \mathbf{C}) = \min_{C_i \in \mathbf{C}} D_1(C, C_i) \qquad (13)$$

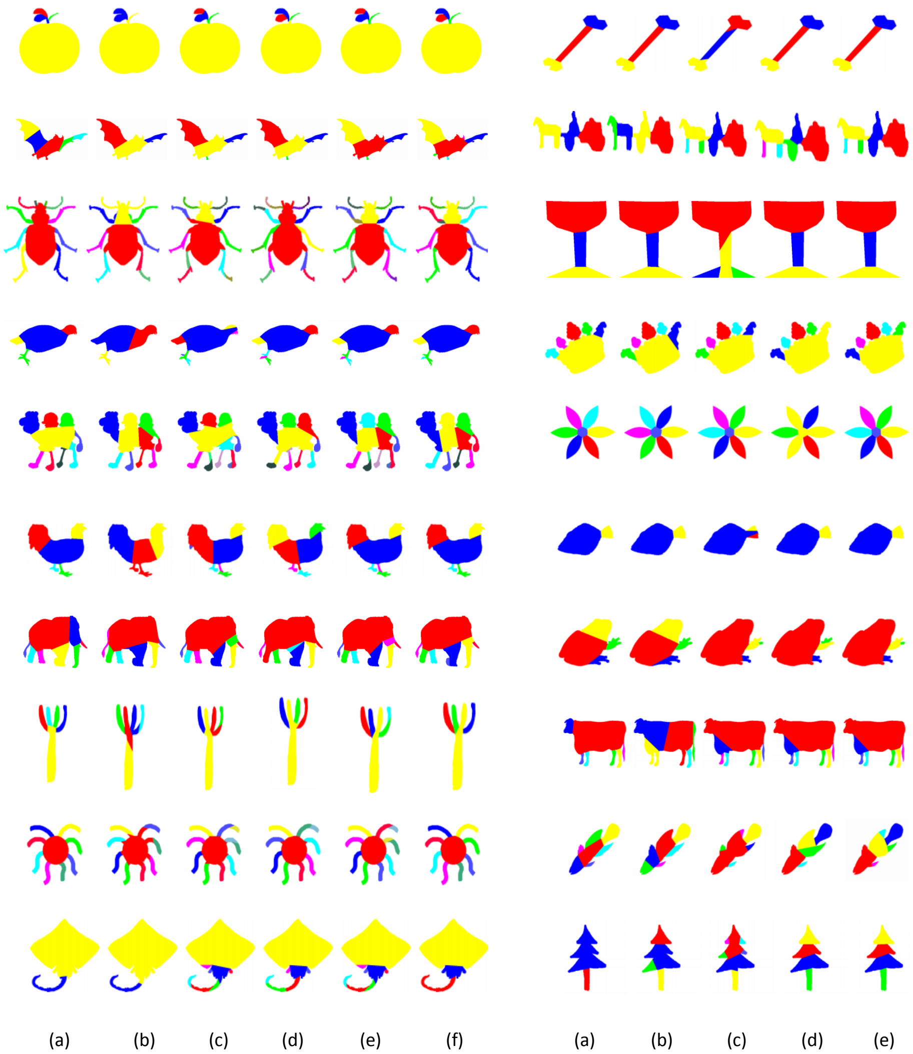

Fig. 6. Examples of decomposition results for 20 categories of MPEG-7. On the left, there are six results for each shape. One the right, there are five results for each shape because the decomposition does not change with straightening. From left to right: (a) human decomposition result, (b) [7]'s result, (c) [8]'s result, (d) [11]'s result, (e) our result without straightening and (f) our result with straightening.

which is the distance between the cut and its nearest cut in the cut sets. And the distance between two cut sets $\mathbf{C}_1$ and $\mathbf{C}_2$ is defined as:

$$D(\mathbf{C}_1, \mathbf{C}_2) = \frac{1}{2}(\frac{1}{|\mathbf{C}_1|} \sum_{C_i \in \mathbf{C}_1} D_2(C_i, \mathbf{C}_2) + \frac{1}{|\mathbf{C}_2|} \sum_{C_j \in \mathbf{C}_2} D_2(C_j, \mathbf{C}_1)). \tag{14}$$

which is the average of the average distance from $\mathbf{C}_1$ to $\mathbf{C}_2$ and that from $\mathbf{C}_2$ to $\mathbf{C}_1$. This distance is symmetric and called as "cut distance".

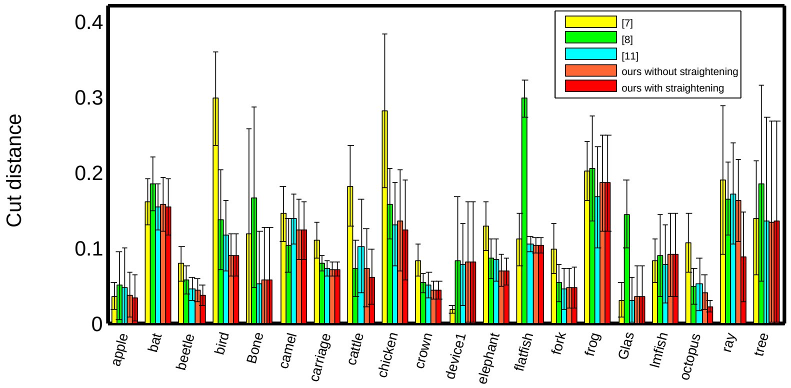

Fig. 7. Distances between the human cuts and out results without(pink)/with straightening(red), [8](green), [7](yellow) and [11](cyan) respectively for the MPEG-7 shape dataset. The black line denotes the standard variance.

Fig. 7 shows the computed "cut distance" of our results as well as [8], [7] and [11]. It is also measured by the ratio of the cut distance over $\mathfrak{D}$. It can be seen that our result is closer to the ground truth compared to the previous work for most categories. Straightening is helpful to improve the cut accuracy for some classes such as octopus and ray. This shows that combining the three generic rules and straightening is useful to learn perceptually meaningful shape parts. Fig. 6 shows examples of the decomposition results from experiments. For each category, we choose one example shape and display the decomposition results by human, [7], [8], [11] and ours. It can be seen that for most categories, our decomposition results are closer to human results.

6 Conclusion

We proposed a method to solve the shape decomposition problem for learning perceptually meaningful parts. By jointly considering three generic rules, we formulate the shape decomposition problem as an optimization problem and design a new metric "cut income" to measure the contribution of candidate cuts for improving the convexity of decomposed parts. By using this metric, the original problem is solved as a quadratic programming problem. In addition, straightening is applied to avoid the superfluous cuts. The experimental results show that our approach is promising to keep a good tradeoff between cut length and convexity, and the results are more consistent with human perception.

Acknowledgement. This work was supported in part by the National Basic Research Program of China (973 Program, 2009CB320904) and the National Science Foundation of China (61103087, 91120004, 61272027).

References

1. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. PAMI **32** (2010) 1627–1645
2. Hoffman, D., Richards, W.: Parts of recognition. Cognition **18** (1984) 65–96
3. Singh, M., Seyranian, G.D., Hoffman, D.D.: Parsing silhouettes: The short-cut rule. Perception and Psychophysics **61** (1999) 636–660
4. Latecki, L.J., Lakamper, R.: Convexity rule for shape decomposition based on discrete contour evolution. CVIU **73** (1999) 441–454
5. Walker, L.L., Malik, J.: Can convexity explain how humans segment objects into parts? Journal of Vision **3** (2003)
6. Kanisza, G., Gerbino, W.: Convexity and symmetry in figureground organization. Vision and Artifact, M. Henle, ed. New York Springer (1976)
7. Gopalan, R., Turaga, P., Chellappa, R.: Articulation-invariant representation of non-planar shapes. In: ECCV, Springer-Verlag (2010) 286–299
8. Liu, H., Liu, W., Latecki, L.: Convex shape decomposition. In: CVPR. (2010) 97–104
9. Pilu, M., Fisher, R.B.: Model-driven grouping and recognition of generic object parts from single images. Robotics and Autonomous Systems **21** (1997) 107–122
10. Zhu, L., Chen, Y., Torralba, A., Freeman, W., Yuille, A.: Part and appearance sharing: Recursive compositional models for multi-view multi-object detection. In: CVPR. (2010) 1919–1926
11. Ren, Z., Yuan, J., Li, C., Liu, W.: Minimum near-convex decomposition for robust shape representation. In: ICCV. (2011) 303–310
12. Siddiqi, K., Kimia, B.B.: Parts of visual form - computational aspects. PAMI **17** (1995) 239–251
13. Mi, X., DeCarlo, D.: Separating parts from 2D shapes using relatability. In: ICCV. (2007) 1–8
14. Rosin, P.L.: Shape partitioning by convexity. IEEE Transactions on Systems Man and Cybernetics Part a-Systems and Humans **30** (2000) 202–210
15. Ling, H.B., Jacobs, D.W.: Shape classification using the inner-distance. PAMI **29** (2007) 286–299
16. Lofberg, J.: Yalmip : A toolbox for modeling and optimization in MATLAB. In: IEEE International Symposium on Computer Aided Control Systems Design, Taipei, Taiwan (2004) 284–289
17. Temlyakov, A., Munsell, B.C., Waggoner, J.W., Wang, S.: Two perceptually motivated strategies for shape classification. In: CVPR, IEEE (2010) 2289–2296
18. Shen, W., Bai, X., Hu, R., Wang, H., Latecki, L.J.: Skeleton growing and pruning with bending potential ratio. Pattern Recognition **44** (2011) 196–209
19. Latecki, L.J., Lakamper, R., Eckhardt, U.: Shape descriptors for non-rigid shapes with a single closed contour. In: CVPR. Volume 5. (2000) 424–429