Minimum Near-Convex Decomposition for Robust Shape Representation

Zhou Ren*, Junsong Yuan
Nanyang Technological University, Singapore
{renzhou, jsyuan}@ntu.edu.sg

Chunyuan Li, Wenyu Liu
Huazhong Univ. of Sci. & Tech., P.R. China
chunyuan.li@hotmail.com, liuwy@hust.edu.cn

Abstract

Shape decomposition is a fundamental problem for part-based shape representation. We propose a novel shape decomposition method called Minimum Near-Convex Decomposition (MNCD), which decomposes 2D and 3D arbitrary shapes into minimum number of "near-convex" parts. With the degree of near-convexity a user specified parameter, our decomposition is robust to large local distortions and shape deformation. The shape decomposition is formulated as a combinatorial optimization problem by minimizing the number of non-intersection cuts. Two major perception rules are also imposed into our scheme to improve the visual naturalness of the decomposition. The global optimal solution of this challenging discrete optimization problem is obtained by a dynamic subgradient-based branch-and-bound search. Both theoretical analysis and experiment results show that our approach outperforms the state-of-the-art results without introducing redundant parts. Finally we also show the superiority of our method in the application of hand gesture recognition.

1. Introduction

It is natural to represent an object by its parts and there has been strong evidence for part-based representations in human vision [15]. According to Siddiqi *et al*. [15],

> Part-based representations allow for recognition that is robust in the presence of occlusion, movement, deletion, or growth of portions of an object. In the task of forming high-level object-centered models from low-level image-based features, parts serve as an intermediate representation.

Given an arbitrary shape, it is thus of great interests to decompose it into a number of natural parts, where each part satisfies certain geometric constraint. The most popular constraint is convexity constraint, because (1) a convex part is visually natural and geometrically simple [3, 17], thus can serve as a satisfactory primitive for recognition; (2) many operators, which are too complicated to be applied on the original objects, can be easily applied to its convex parts [4, 13]. To this end, strict convex decomposition has been a well studied problem in computational geometry [7, 8].

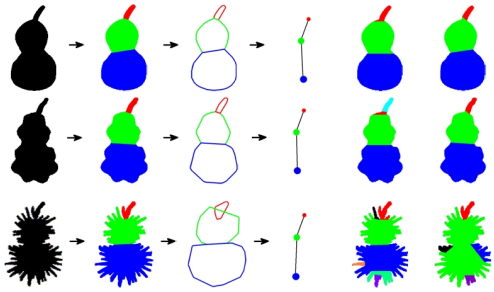

Figure 1. A robust shape representation via our method. The first column shows the same objects with different degrees of local distortions; the second column shows the near-convex decomposition results using our method; the third column shows the shape representations by replacing each part with its convex hull; and the fourth column shows the graph representations by replacing each part with a node. Despite severe local distortions, as our method decomposes a shape into minimum number of near-convex parts, it avoids introducing redundant parts and thus brings consistent decomposition results. The last two columns are the results of existing near-convex decomposition methods: [10] and [12], respectively.

However, in practice, strict convex decomposition is not robust because it is sensitive to small variations of the shape, such as the local distortions on the contour, which are commonly caused due to imperfect image segmentation and shape deformations. In such cases, to satisfy the strict convex requirement, it usually results in a large number of redundant small parts, thus does not lead to consistent shape representation.

To handle this problem, near-convex decomposition has been proposed. As illustrated in Fig.1, instead of requiring each part to be strictly convex, it allows near-convex parts. In [10, 11], Lien *et al*. proposed a greedy strategy for near-convex decomposition, which exhaustively partitions the most concave feature in the shape until all the parts satisfy the convexity constraint. A recent method proposed by Liu *et al*. [12] formalized the near-convex decomposition as a linear programming problem by minimizing the total length of cuts, and obtained an approximate optimal solution. Generally, by tolerating local non-convex distortions,

*Part of this work was done while the author was at Huazhong Univ. of Sci. & Tech.

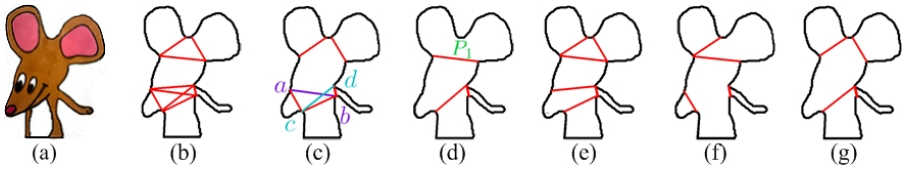

Figure 2. Illustration of near-convex decomposition. (a) The original image. (b) The extracted shape with some sampled candidate cuts inside. (c) An incorrect near-convex decomposition which does not satisfy *the non-overlapping constraint*, as the purple line ab intersects with the cyan line cd causing the part abc overlaps with the part bcd. (d) An incorrect near-convex decomposition which does not satisfy *the convexity constraint*, as $concave(P_1) > \psi$. (e) A near-convex decomposition of 7 parts. (f) A minimum near-convex decomposition of 5 parts. (g) Another minimum near-convex decomposition of 5 parts, but looks more natural.

near-convex decomposition leads to more robust shape representation.

Despite previous works in near-convex shape decomposition, there still remain two unsolved problems. First, the existing methods cannot avoid introducing redundant parts. For example, the greedy algorithm proposed in [10, 11] inevitably results in redundant parts. Also by only optimizing the total cut length, [12] can produce redundant parts as well. Thus, these methods cannot generate robust shape decomposition, as illustrated in the last two columns of Fig.1. Secondly, without any priori knowledge of the object, it is difficult to obtain visually natural parts through unsupervised decomposition.

To handle these two problems, we present a novel near-convex decomposition method called Minimum Near-Convex Decomposition (MNCD) to decompose arbitrary 2D and 3D shapes. By finding a collection of candidate cuts to partition the shape into near-convex parts, we formulate the shape decomposition problem as a combinatorial optimization problem by selecting the best subset of candidate cuts that has both a minimum size and high visual naturalness. Two major perception rules, the minima rule [6] and the short cut rule [18], are also imposed to improve the visual naturalness of the decomposition. The global optimal solution of this challenging discrete optimization problem is found by a dynamic subgradient-based branch-and-bound search.

Extensive experiments on both 2D and 3D shapes show that our algorithm is robust to local distortions and shape deformation. The comparisons with the state-of-the-art results validate the advantages of our method in terms of reducing the number of redundant parts. With less and better recognition primitives, we also demonstrate the superiority of our method in the application of hand gesture recognition. The contributions of our method are summarized as follows:

- It decomposes an arbitrary shape into minimum number of near-convex parts, and can be easily extended to decompose 3D shapes.
- With the degree of near-convexity a user specific parameter, it can well handle local shape distortions and shape deformation.
- It imposes perception rules in cognitive science to guide the shape decomposition, thus is visually more natural.

2. Problem Formulation

2.1. Overview

In near-convex decomposition, each decomposed part may not be strictly convex, thus the user has to specify a parameter ψ which indicates the near-convex tolerance of the decomposed parts. Formally, a ψ-near-convex decomposition of a shape S, $\mathrm{D}_\psi(S)$, is defined as a decomposition that only contains ψ-near-convex non-overlapping parts, i.e.:

$$\mathrm{D}_\psi(S) = \{P_i | \bigcup_i P_i = S, \forall_{i \neq j} P_i \cap P_j = \emptyset, concave(P_i) \leq \psi\}, \quad (1)$$

where P_i denotes the decomposed part; $concave(P_i)$ is the concavity of P_i. We say P_i is ψ-near-convex if $concave(P_i) \leq \psi$. P_i is strictly convex if $concave(P_i)$=0. According to the definition, near-convex decomposition has two constraints: *the non-overlapping constraint*, $\forall_{i \neq j} P_i \cap P_j = \emptyset$; *the convexity constraint*, $\forall P_i$, $concave(P_i) \leq \psi$.

The partition $\{P_i\}$ is formed by some cuts. For any two vertices p, q on the contour, if the line connecting p and q locates inside the shape, line pq is a cut. As shown in Fig.2(b), the red lines are some sampled cuts. We denote the complete set of all possible cuts in shape S as the candidate cut set, $C(S)$. Therefore, as shown in Fig.2, a near-convex decomposition of S is to select a subset of cuts from $C(S)$ to form $\{P_i\}$ such that the two constraints in Eq.1 are satisfied: (1) as illustrated in Fig.2(c), to ensure the non-overlapping constraint, the selected cuts cannot intersect with each other; and (2) as illustrated in Fig.2(d), to ensure the convexity constraint, we restrict $\forall P_i$, $concave(P_i) \leq \psi$.

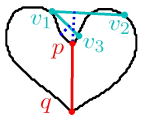

Figure 3. At the concave contour, some lines (such as v_1v_2, v_1v_3) intersect with the contour or locate outside the contour, which form the mutex pairs; while vertices v_2, v_3 are not a mutex pair.

In order to measure $concave(P_i)$, we apply the shape feature $mutex\ pair$ in [12]: for any two vertices on a shape contour, v_1 and v_2, if the connecting line between v_1 and v_2 intersects with the contour or locates outside the contour, (v_1, v_2) is a mutex pair. As shown in Fig.3, (v_1, v_2) and (v_1, v_3) are two mutex pairs. The concavity of a part P_i is

defined as the maximal concavity of the mutex pairs in the part:

$$concave(P_i) = \max_{(v_1,v_2)\in P_i} \{concave_m(v_1, v_2)\}, \quad (2)$$

where (v_1, v_2) denotes the mutex pair in P_i; $concave(P_i)$ denotes the concavity of P_i; $concave_m(v_1, v_2)$ is the concavity of mutex pair (v_1, v_2).

Hence, we can measure $concave(P_i)$ by measuring $concave_m(v_1, v_2)$. We use the same method proposed in [12] to measure $concave_m(v_1, v_2)$: by projecting the shape contour in multiple Morse functions, the concavity of a mutex pair is defined as the maximal perpendicular distance between line v_1v_2 and the corresponding concave contour. As in Fig.3, $concave_m(v_1, v_2)$ and $concave_m(v_1, v_3)$ are shown in the blue dotted lines, and $concave_m(v_1, v_2) > concave_m(v_1, v_3)$.

To ensure the convexity constraint: $\forall P_i$, $concave(P_i) \leq \psi$, according to Eq.2, the concavities of all the mutex pairs in each part P_i must be smaller than ψ. Therefore, for a ψ-near-convex decomposition, we need to separate all the mutex pairs in S whose concavities are greater than ψ into different parts to ensure $concave(P_i) \leq \psi$. As shown in Fig.3, cut pq separates the heart shape into two parts, and the mutex pair (v_1, v_2) as well as (v_1, v_3) are separated. Thus $concave_m(v_1, v_2)$ and $concave_m(v_1, v_3)$ will not affect the concavities of these two parts.

2.2. Minimum Near-Convex Decomposition

As illustrated in Fig.2(e), Fig.2(f) and Fig.2(g), in order to decompose a shape into minimum number of parts with high visual naturalness, we need to optimize the selection of cuts. Assume there are in total n possible cuts in a shape S, namely $C(S) = \{cut_1, ..., cut_n\}$. The final decomposition consists of a subset of the cuts from $C(S)$, denoted by $C'(S) \subseteq C(S)$. We assign a binary variable x_i to each cut_i in $C(S)$ where:

$$x_i = \begin{cases} 1 & cut_i \in C'(S), \\ 0 & cut_i \notin C'(S). \end{cases} \quad (3)$$

Thus $\mathbf{x}_{n\times 1} = (x_1, x_2, ..., x_n)^\top$ is a binary vector indicating the selection/non-selection of cuts from $C(S)$.

With the two constraints in Eq.1, by minimizing the number of cuts and imposing perception rules, we formulate the ψ-MNCD as follows:

$$\begin{aligned} &\mathbf{min} \quad \| \mathbf{x} \|_0 + \lambda \mathbf{w}^\top \mathbf{x}, \\ &s.t. \ \ \mathbf{Ax} \geq \mathbf{1}, \quad \mathbf{x}^\top \mathbf{Bx} = 0, \quad \mathbf{x} \in \{0, 1\}^n, \end{aligned} \quad (4)$$

where $\|\mathbf{x}\|_0$ is the zero-norm of vector $\mathbf{x}$, which counts the number of the selected cuts in $C'(S)$. $\lambda \geq 0$ is a parameter introducing the visual naturalness regularization $\mathbf{w}^\top \mathbf{x}$ to the decomposition, in order to regularize the cuts selection by favoring the cuts with high visual naturalness. We will discuss λ in Section 3.1. Now we explain our formulation.

The visual naturalness regularization: $\mathbf{w}^\top \mathbf{x}$

We employ both the minima rule [6] and the short cut rule [18] to ensure high visual naturalness of the decomposition. A cost is assigned to each $cut_i \in C(S)$ to evaluate its own visual naturalness, and a smaller cost means a higher visual naturalness:

$$w_{pq} = \frac{dist(pq)}{1 + \beta \cdot |\min\{cur(p), 0\} + \min\{cur(q), 0\}|}, \quad (5)$$

where cut pq is a candidate cut in $C(S)$; $dist(pq)$ is the normalized distance between vertices p and q. This corresponds to the short cut rule: a shorter cut has a smaller cost. $cur(p)$ denotes the normalized curvature of the vertex p, which corresponds to the minima rule: a cut resolving at positions with greater negative curvatures has a smaller cost. We normalize the negative curvature among concave vertices and ignore the convex vertices. β is a parameter balancing these two rules. As these two rules both are critical for natural decomposition, we set $\beta = 1$ in our experiments.

We denote $\mathbf{w}_{n\times 1} = (w_1, w_2, ..., w_n)^\top$ as the costs of n candidate cuts. From Eq.5, we know that the cuts separating at positions with greater negative curvatures and with shorter lengths have smaller costs. Thus by minimizing $\mathbf{w}^\top \mathbf{x}$, those cuts with higher visual naturalness are more likely to be selected.

The convexity constraint: $\mathbf{Ax} \geq \mathbf{1}$

As mentioned in Section 2.1, to ensure the convexity constraint: $\forall P_i$, $concave(P_i) \leq \psi$, we need to separate all the mutex pairs in S whose concavities are greater than ψ into different parts. So we first obtain the ψ-mutex set of S, $M^\psi(S)$, which is defined as the set of mutex pairs whose concavities are greater than ψ. Then we separate all the mutex pairs in $M^\psi(S)$ with the selected cuts from $C(S)$. A candidate cut may separate several mutex pairs, such as the cut pq in Fig.3. For every candidate cut in $C(S)$, cut_i, the mutex pairs it can separate form a subset of $M^\psi(S)$, denoted by M'_i. In this way, we obtain $\{M'_i,\ i = 1, ..., n\}$.

Suppose there are m mutex pairs in the ψ-mutex set, $M^\psi(S) = \{mp_1, ..., mp_m\}$. For each mutex pair in $M^\psi(S)$, mp_i, among all the cuts that can separate it, at least one cut must be in set $C'(S)$. Thus, for each mp_i, this gives a constraint:

$$\sum_{j=1}^{n} a_{ij} x_j \geq 1, \text{ where } a_{ij} = \begin{cases} 1 & mp_i \in M'_j, \\ 0 & mp_i \notin M'_j. \end{cases} \quad (6)$$

Let us denote $\mathbf{A}_{m\times n} = (a_{ij} | i = 1, ..., m;\ j = 1, ..., n)$, $\mathbf{1}_{m\times 1} = (1, ..., 1)^\top$. Consider all the m mutex pairs in $M^\psi(S)$, we have the convexity constraint: $\mathbf{Ax} \geq \mathbf{1}$, which is also used in [12].

	ACD [10]	CSD [12]	MNCD
Objective	a NCD without optimization	a NCD with the minimum length of cuts	a NCD with minimum number of parts and high visual naturalness
Candidate cut set	complete set of all possible cuts	incomplete set from Reeb graph	complete set of all possible cuts
Perception rules	minima rule and short cut rule	short cut rule	minima rule and short cut rule
Constraints	non-overlapping constraint convexity constraint	convexity constraint	non-overlapping constraint convexity constraint
Solution	greedy algorithm	approximation algorithm	global optimal algorithm

Table 1. The comparison among ACD, CSD and MNCD, where NCD denotes near-convex decomposition.

The non-overlapping constraint: $\mathbf{x}^\top\mathbf{B}\mathbf{x}$=0

As for two cuts in $C(S)$, cut_i and cut_j, they may intersect with each other. We define an intersection matrix, $\mathbf{B}_{n\times n}$, to indicate the intersection relations in $C(S)$:

$$b_{ij} = \begin{cases} 0 & cut_i \text{ does not intersect with } cut_j, \text{ and } i \neq j, \\ 1 & cut_i \text{ intersect with } cut_j, \text{ and } i \neq j, \\ 0 & i = j. \end{cases} \quad (7)$$

As mentioned in Section 2.1, to ensure the non-overlapping constraint $\forall_{i\neq j} P_i \cap P_j = \emptyset$, the selected cuts in $C'(S)$ cannot intersect with each other, namely $\forall\ x_i, x_j \in \mathbf{x}$, $x_i \times b_{ij} \times x_j = 0$. Thus we have the intersection constraint: $\mathbf{x}^\top\mathbf{B}\mathbf{x}$=0.

3. Our solution

3.1. Selection of parameter λ

As mentioned earlier, λ is an important parameter introducing the visual naturalness regularization to the decomposition. If we do not consider the visual naturalness of the decomposition, while only focus on the minimum number of parts, the problem can be reformulated by setting $\lambda = 0$, i.e.:

$$\mathbf{min}\ \|\mathbf{x}\|_0\ \ s.t.\ \mathbf{A}\mathbf{x} \geq \mathbf{1},\ \mathbf{x}^\top\mathbf{B}\mathbf{x} = 0,\ \mathbf{x} \in \{0,1\}^n. \quad (8)$$

The solution $\mathbf{x}$ of this formulation is not unique, but it ensures exactly minimum number of parts. Although with different objective functions, we can prove that our formulation in Eq.4 can obtain the same minimum number of parts as Eq.8 if selecting λ appropriately. Theorem 1 tells the relationship between Eq.8 and our formulation in Eq.4:

Theorem 1 minimum decomposition rule

We consider two objective functions as follows:

$$f(\mathbf{x}) = \|\mathbf{x}\|_0 + \lambda\mathbf{w}^\top\mathbf{x},\ \ s.t.\ \mathbf{A}\mathbf{x} \geq \mathbf{1},\ \mathbf{x}^\top\mathbf{B}\mathbf{x} = 0,\ \mathbf{x} \in \{0,\ 1\}^n,$$
$$g(\mathbf{x}) = \|\mathbf{x}\|_0,\ \ s.t.\ \mathbf{A}\mathbf{x} \geq \mathbf{1},\ \mathbf{x}^\top\mathbf{B}\mathbf{x} = 0,\ \mathbf{x} \in \{0,\ 1\}^n,$$

Let:

$$\mathbf{x}' = \arg\min_{\mathbf{x}} f(\mathbf{x}), \quad \mathbf{x}'' = \arg\min_{\mathbf{x}} g(\mathbf{x}).$$

We have $\|\mathbf{x}'\|_0 = \|\mathbf{x}''\|_0$ *when* $0 \leq \lambda \leq 1/\sum_{i=1}^n w_i$.

The proof of Theorem 1 is in the Appendix. $\mathbf{x}''$ is the solution of Eq.8 whose zero-norm is minimized, and $\mathbf{x}'$ is the solution of our formulation in Eq.4. Therefore, our formulation can decompose a shape into minimum number of parts as well when $0 \leq \lambda \leq 1/\sum_{i=1}^n w_i$. It is worth mentioning that although Eq.4 and Eq.8 both minimize the number of parts, their cuts are not necessarily the same subset from $C(S)$, as Eq.4 favors visually more natural cuts.

3.2. Branch-and-bound search

Algorithm 1: MNCD(S, ψ)

Input: A shape, S, and a concavity tolerance, ψ;
Output: ψ-MNCD of S, $\{P_i\}$.

```
◇ compute the candidate cut set, C(S);
◇ compute ψ-mutex set of S→M^ψ(S);
foreach mp_i in M^ψ(S) do
    foreach cut_j in C(S) do
        check whether cut_j separates mp_i → a_ij;
foreach cut_i in C(S) do
    compute its cost → w_i;
    foreach cut_j in C(S) do
        check whether cut_i intersects with cut_j → b_ij;
◇ solve the optimization problem in Eq.4;
◇ obtain its global optimal solution→ {P_i}.
```

The shape decomposition problem formulated in Eq.4 is a challenging combinatorial optimization problem, as the solution space is of size $O(2^n)$.

In order to obtain the global optimal solution efficiently, we employ a dynamic subgradient-based branch-and-bound search [2]. The main characteristic of dynamic subgradient-based branch-and-bound procedure is that at every node of the search tree, instead of using the simplex method to solve the linear programming relaxation of the given subproblem, it uses primal and dual heuristics with subgradient optimization applied to a Lagrangian dual, to generate the upper and lower bounds on the objective function.

The dynamic subgradient procedure is embedded into a branch-and-bound algorithm. We use a breadth first search strategy. The branch-and-bound algorithm does one or several of the following things: improves the lower bound; improves the upper bound; fixes some variables at 0 or 1. It gradually makes the dual solution feasible, to attempt to find a better cover, and to recursively fix as many variables as possible. The resulting partial cover is then completed to a full cover when the global optimal solution is found.

Algorithm 1 shows the overall procedure of our method. In our experiment, the $\mathbf{x}$ vector generally contains $n = 10,000 \sim 100,000$ elements. Although we have a large number of variables, because of the sparsity of the solution, i.e., typically only several or tens of cuts will be selected, the global optimal solution can be found efficiently in several seconds using the branch-and-bound search.

3.3. Comparison to other methods

Table 1 presents a comparison among our MNCD and the state-of-the-art methods: ACD [10] and CSD [12]. Our method aims at the minimum number of parts with high visual naturalness for robust shape representation, and the global optimal solution is found by the algorithm above.

Specifically, CSD is a special case of our formulation in Eq.4 if discarding the $\|\mathbf{x}\|_0$ term and setting $\beta = 0$ in Eq.5. The $\|\mathbf{x}\|_0$ term in our formulation guarantees the minimum number of decomposed parts, which reduces all the redundant part in near-convex decomposition. This point is very essential for robust shape representation and can improve the efficiency of further applications, as shown in Fig.1. β in Eq.5 is the parameter imposing the minima rule and short cut rule into our near-convex decomposition scheme, $\beta = 0$ means discarding the minima rule. This point is essential as well because these two perception rules are introduced for high visual naturalness which guarantees better recognition primitives, the minima rule inhibits the decomposition by cuts at positions with small negative curvatures or even at convex contour.

4. Experiments

4.1. 2D Shape Decomposition

Figure 4. An example of each shape category selected from the MPEG-7 dataset [9] is displayed.

In order to evaluate our method on 2D shapes, we test the MPEG-7 shape dataset [9]. Excluding simple shapes such as the heart shape that can be easily decomposed, we select 20 complex shape categories from MPEG-7 dataset, in which each category has 20 shapes (20×20=400 shapes). Fig.4 shows an image for each selected category.

Evaluation of parameters

In our algorithm, there are 2 parameters, ψ and λ, where ψ is the user specified concavity tolerance for near-convex decomposition; λ is the parameter introducing the visual naturalness.

The parameter ψ tells how small degree of concave features the user want to ignore in near-convex decomposition.

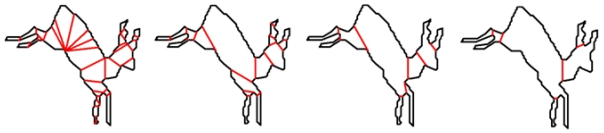

Figure 5. The decomposition results by MNCD, with ψ= $0.005R$, ψ=$0.01R$, ψ=$0.03R$ and ψ=$0.06R$, from left to right, respectively, where R is the radius of the shape's minimum enclosing disk.

Fig.5 shows the decomposition results at different value of ψ. A very small ψ means that the decomposed parts are almost strictly convex, which will introduce a large number of small parts to ensure the convexity constraint, thus is not robust to local distortions. When ψ increases, the decomposition can tolerate more severe distortions.

Figure 6. The decomposition results of MNCD when ψ=$0.03R$, with λ= 0, λ=$0.5/\sum_{i=1}^{n} w_i$, λ=$1/\sum_{i=1}^{n} w_i$, from left to right, respectively.

The parameter λ introduces the visual naturalness to the decomposition in Eq.4. Fig.6 shows the decomposition results of MNCD at different values of λ. If $0 \leq \lambda \leq 1/\sum_{i=1}^{n} w_i$, the number of parts by MNCD is minimized. But a larger λ brings a more natural decomposition as it counts more weight of the visual naturalness term in Eq.4. In our experiments below, we use λ=$1/\sum_{i=1}^{n} w_i$.

Evaluation of the number of parts

One advantage of our method is that it does not introduce redundant part as it decomposes the shape into minimum number of parts. In terms of the number of parts, table 2 presents the *average reduction rate* comparing our method with ACD [10] and CSD [12] at 4 different ψ, on MPEG-7 dataset. The average reduction rate scores are defined as:

$$\text{ACD}\downarrow = (\#\text{ACD} - \#\text{MNCD})/\#\text{ACD}\ ,$$
$$\text{CSD}\downarrow = (\#\text{CSD} - \#\text{MNCD})/\#\text{CSD}\ .$$

As it shows, we produce the least number of parts. Comparing to ACD [10], up to 32.7% number of redundant parts are reduced, and up to 30.7% number of redundant parts are reduced comparing to CSD [12]. On average, comparing to ACD 19.18% number of parts are reduced and 10.62% comparing to CSD. Thus, the efficiency of further applications on the decomposed parts can be highly improved. On the other hand, from the table, we notice that all the ACD$\downarrow$ and CSD$\downarrow$ scores are greater than 0 on every shape category and every ψ, which means that MNCD produces minimum number of parts at all time, as proved in Theorem 1.

Decomposition results

To further evaluate the visual naturalness of our decomposition, Fig.7 compares our method with the method proposed

MPEG-7 dataset	ψ=0.005R		ψ=0.01R		ψ=0.03R		ψ=0.06R	
	ACD↓	CSD↓	ACD↓	CSD↓	ACD↓	CSD↓	ACD↓	CSD↓
bat	14.3%	8.9%	20.8%	11.3%	16.2%	6.8%	8.6%	6.5%
beetle	23.8%	10.3%	22.9%	9.0%	21.9%	16.0%	19.3%	14.4%
bird	18.5%	13.6%	23.8%	12.5%	12.8%	7.6%	17.4%	10.6%
butterfly	4.4%	5.8%	13.1%	7.2%	16.9%	8.8%	32.7%	12.9%
camel	16.1%	10.5%	15.2%	3.3%	21.1%	9.5%	21.3%	4.8%
carriage	5.5%	3.7%	13.8%	9.2%	15.6%	9.5%	18.4%	13.3%
cattle	24.9%	14.6%	24.5%	10.7%	27.4%	8.9%	23.0%	12.3%
chicken	19.0%	10.0%	23.1%	15.2%	24.0%	10.5%	3.1%	5.2%
chopper	8.9%	7.7%	16.2%	10.4%	22.1%	10.7%	17.4%	11.3%
crown	16.0%	9.2%	20.7%	11.9%	27.8%	14.6%	19.4%	16.7%
deer	18.0%	14.5%	24.2%	10.5%	15.3%	4.2%	22.6%	13.3%
dog	23.8%	15.4%	18.8%	7.6%	24.5%	9.2%	15.7%	10.5%
elephant	24.1%	12.0%	24.0%	8.9%	24.9%	9.7%	25.2%	7.8%
fly	11.9%	9.2%	8.9%	5.6%	4.2%	3.9%	10.6%	8.4%
horse	20.1%	8.0%	23.8%	5.1%	19.8%	1.1%	18.8%	6.1%
horseshoe	26.1%	18.6%	21.9%	11.7%	23.5%	14.8%	12.2%	12.2%
lizard	18.2%	10.4%	15.9%	10.0%	27.5%	15.2%	11.7%	7.3%
Misk	29.8%	30.7%	24.2%	11.9%	25.8%	20.3%	13.2%	15.4%
Mickey	24.6%	13.4%	14.0%	10.5%	19.8%	12.9%	17.3%	8.5%
spring	22.6%	12.6%	25.1%	13.7%	24.5%	15.8%	25.7%	6.9%

Table 2. The average reduction rate of MNCD comparing with ACD [10] and CSD [12], on the MPEG-7 dataset, where R is the radius of the shape's minimum enclosing disk.

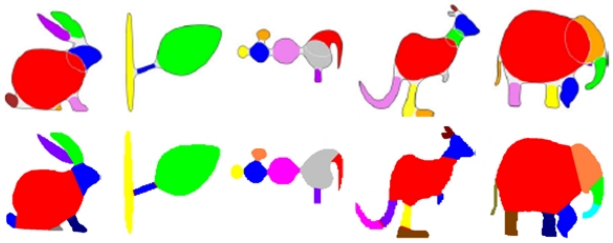

Figure 7. The first row shows the decomposition results of [14], and the second row shows the results of MNCD.

by Mi and Decarlo [14]. Mi's method is specifically designed to decompose 2D shapes into natural parts. The first row are the decomposition results of their method, and the second row are the results of MNCD. As we can see, when considering the minima rule and short cut rule in our formulation, our method decomposes shapes into parts with high visual naturalness comparable to [14], such as the legs, head and body of the animal, the leaf and stem of the tree, etc.

In Fig.11, more comparisons among ACD [10], CSD [12] and our method are provided, with ψ=0.03R. The decompositions of our method produce the least and more natural recognition primitives. At this concavity tolerance, MNCD decomposes the animals into primitives such as head, body, legs and tail, and avoid decomposing them into redundant parts as [10, 12].

Without introducing redundant parts, MNCD is robust to local distortions, as shown in the first row of Fig.12. The robustness of our method is more obvious when there are large local distortions as shown in the last row of Fig.1, while the existing decomposition methods produce many redundant noise parts. Besides, our MNCD imposes two perception rules to guide the decomposition, thus it produces more natural parts, which makes MNCD robust to shape deformation, as illustrated in the second row of Fig.12.

Figure 8. Illustration of our hand gesture recognition using the Kinect depth camera and MNCD. The first and second columns are the color and depth image in the new dataset; the third column is the image segmentations of hands; the last column is the MNCD decompositions of the hand shapes.

Thanks to the robust shape representation of our MNCD, it has a high potential for shape-based visual recognition tasks. In the next section, we apply it to hand gesture recognition.

4.2. Hand Gesture Recognition

For hand gesture recognition based HCI [5], usually the color, texture, shading, and context information are not robust for successful recognition, while the shape feature alone is often sufficient. However, the vision-based hand gesture recognition is extremely hard, because of two primary problems: 1. It is hard to segment the hand out of the image with cluttered background; 2. Even with the shape of a hand, existing representations are not robust enough for gesture recognition. For example, the contour-based and the skeleton-based representations can be affected by large local noises.

With the advent of Kinect depth camera [1], we can accurately segment the hand shape using both image and depth information, as shown in Fig.8. After that, we can use MNCD to robustly represent the hand shape for gesture recognition. With the Kinect depth camera, we collect a new hand gesture dataset with both color images and depth maps. Our dataset contains 3 hand gesture categories, namely Rock, Paper and Scissors, each category has 50 samples. For each category, an example is shown in the first two columns of Fig.8.

However, even with the help from the Kinect depth camera, the image segmentation of the hand is not perfect. Due to low-resolution, it easily introduces large local distortions or other types of noises on the contour, as shown in the third column of Fig.8. However, our MNCD is robust to handle most of the variations, and decomposes hand shapes into natural primitives such as fingers and palm. We can recognize the hand gesture among Rock, Paper, Scissors by only counting the number of parts. Suppose k is the number of

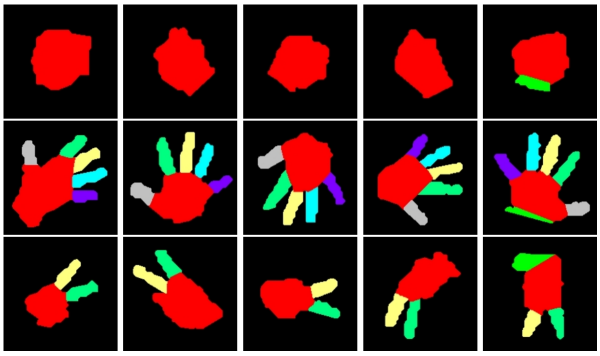

Figure 9. Hand gesture recognition using MNCD. The hand shapes are extracted in real environments using the Kinect depth camera. The first 4 columns are correct recognitions in different environments, and the last column shows some imperfect results.

parts, we classify a gesture to Rock if $k \leq 2$, Paper if $k \geq 5$, and Scissors otherwise. Fig.9 shows some recognition results using MNCD under various scale, orientation and illumination conditions. The last column is some imperfect results because of unsatisfactory hand image segmentation. Our hand gesture recognition method using MNCD is robust to local distortions, scale and orientation changes. Table 3 presents the confusion matrix of our method on our new dataset. The mean accuracy of our method recognizing these three hand gestures is 94.7%.

	Rock	Paper	Scissors
Rock	0.96	0	0.04
Paper	0	0.98	0.02
Scissors	0.1	0	0.9

Table 3. The confusion matrix of our method on the new dataset. The mean accuracy is 94.7%.

4.3. 3D Shape Decomposition

To test our algorithm on 3D shapes, we project 3D shapes into 2D planes multiple times, and obtain 3D shape features from the 2D projections. The 3D decomposition is formulated the same as 2D in Eq.4. Fig.10 shows some decomposition results of the shapes from Mcgill 3D Shape Benchmark [16]. As it shows, MNCD can decompose 3D shapes into parts with high visual naturalness, such as the fin of the whale, the legs of the bear, the lenses of the glasses, the fingers of the hand, etc.

The last row in Fig.10 illustrates the robustness of MNCD decompositions for 3D human postures. Although the human body varies significantly with different postures, the MNCD decomposition results are stable.

5. Conclusion

In this paper, we proposed a novel near-convex shape decomposition approach for robust shape representation,

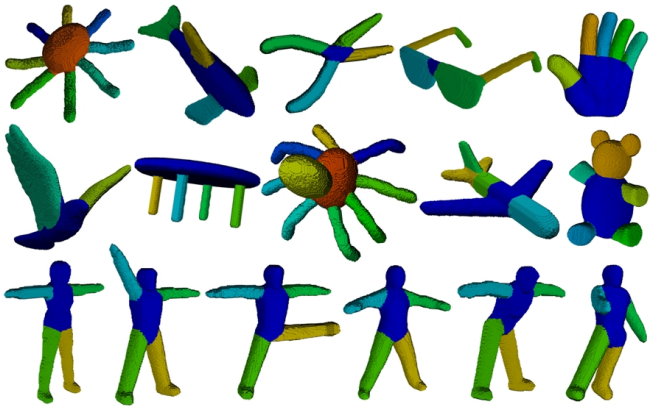

Figure 10. The 3D shape decompositions results of MNCD. The last row illustrates the robustness of our method to shape deformation.

which decomposes 2D and 3D shapes into minimum number of parts with high visual naturalness. With the convexity constraint, the non-overlapping constraint and by imposing the perception rules, we formulate the shape decomposition problem as a combinatorial optimization problem, where the global optimal solution is found by a dynamic subgradient-based branch-and-bound search. We have proved that our method can decompose the shapes into exactly minimum number of near-convex parts. Experiments on complex 2D and 3D shape datasets show that our proposed method outperforms the state-of-the-art methods in terms of the number of decomposed parts and the visual naturalness. We also demonstrate the robustness of our method in the application of hand gesture recognition with Kinect camera.

Appendix

We prove Theorem 1 here. In order to prove $\| \mathbf{x}' \|_0 = \| \mathbf{x}'' \|_0$, when $0 \leq \lambda \leq 1/\Sigma_{i=1}^{n} w_i$, first we have:

$$\min_{\mathbf{x}} f(\mathbf{x}) = \| \mathbf{x}' \|_0 + \lambda \mathbf{w}^\top \mathbf{x}' \leq \| \mathbf{x}'' \|_0 + \lambda \mathbf{w}^\top \mathbf{x}'', \tag{9}$$

$$\min_{\mathbf{x}} g(\mathbf{x}) = \| \mathbf{x}'' \|_0 \leq \| \mathbf{x}' \|_0, \tag{10}$$

As $w_i > 0$, so when $0 \leq \lambda \leq 1/\Sigma_{i=1}^{n} w_i$, $\forall\, \mathbf{x} \in \{0,\ 1\}^n$, $0 \leq \lambda \mathbf{w}^\top \mathbf{x} \leq 1$. Therefore, from Eq.9 we further have Eq.11, and from Eq.10 we further have Eq.12:

$$\| \mathbf{x}' \|_0 + \lambda \mathbf{w}^\top \mathbf{x}' \leq \| \mathbf{x}'' \|_0 + 1. \tag{11}$$

$$\| \mathbf{x}'' \|_0 \leq \| \mathbf{x}' \|_0 + \lambda \mathbf{w}^\top \mathbf{x}'. \tag{12}$$

Combining Eq.11 and Eq.12, we have:

$$\| \mathbf{x}'' \|_0 \leq \| \mathbf{x}' \|_0 + \lambda \mathbf{w}^\top \mathbf{x}' \leq \| \mathbf{x}'' \|_0 + 1.$$

As $0 \leq \lambda \mathbf{w}^\top \mathbf{x}' \leq 1$, and $\| \mathbf{x}' \|_0$, $\| \mathbf{x}'' \|_0$ are integers, thus $\| \mathbf{x}' \|_0 = \| \mathbf{x}'' \|_0$ when $0 \leq \lambda \leq 1/\Sigma_{i=1}^{n} w_i$.

Acknowledgement

This work was supported in part by the Nanyang Assistant Professorship (SUG M58040015) to Dr. Junsong Yuan and the National Natural Science Foundation of China (grant No. 60873127).

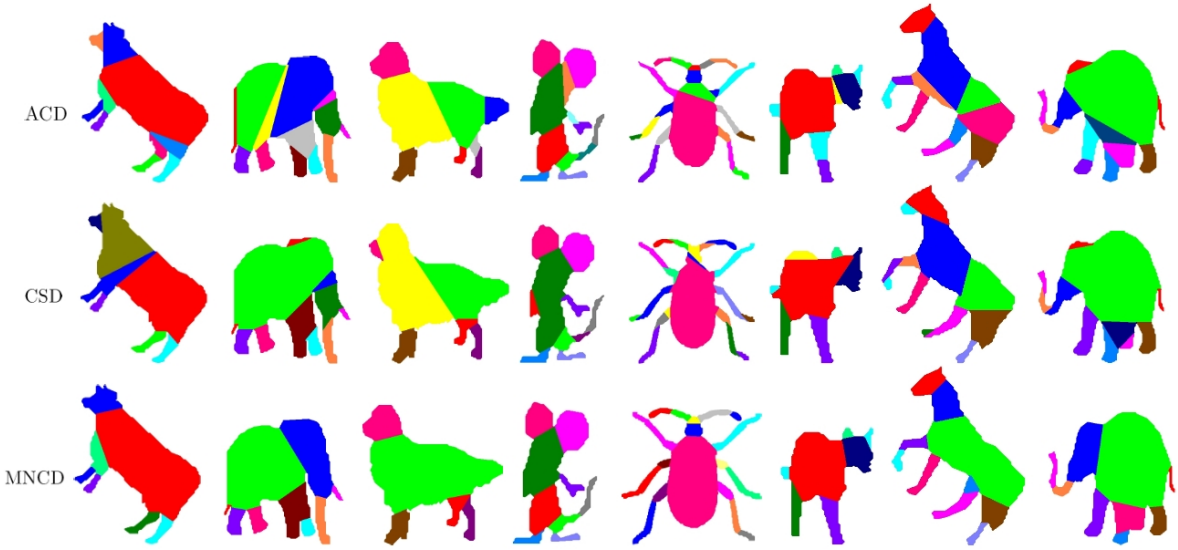

Figure 11. Some decomposition results of ACD [10], CSD [12] and MNCD. Our MNCD method produces the least number of near-convex parts and our decompositions are visually more natural.

Figure 12. The robust decomposition results of MNCD. The first row is the results of shapes with local distortions; the second row is the results of shapes with deformation. Without introducing redundant parts and by considering perception rules, MNCD is robust to local distortions and shape deformation.

References

[1] Microsoft Corp. Redmond WA. Kinect for Xbox 360. 6

[2] E. Balas and M. C. Carrera. A dynamic subgradient-based branch-and-bound procedure for set covering. *Operations Research*, 44:875–890, 1996. 4

[3] I. Biederman. Recognition-by-components: A theory of human image understanding. *Psychological Rev.*, 94:115–147, 1987. 1

[4] T. A. Cass. Robust affine structure matching for 3d object recognition. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 20:1265 – 1274, 1998. 1

[5] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly. Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding*, 108:52–73, 2007. 6

[6] D. D. Hoffman and M. Singh. Salience of visual parts. *Cognition*, 14:29–78, 1997. 2, 3

[7] J. M. Keil and J. Snoeyink. Minimum convex decomposition. www.cs.ubc.ca/∼snoeyink/demos/convdecomp. 1

[8] J. M. Keil and J. Snoeyink. On the time bound for convex decomposition of simple polygons. *International Journal of Computational Geometry and Application*, 12:181–192, 2002. 1

[9] L. J. Latecki, R. Lakamper, and U. Eckhardt. Shape descriptors for non-rigid shapes with a single closed contour. In *Proc. of Computer Vision and Pattern Recognition*, pages 424–429, 2000. 5

[10] J.-M. Lien and N. Amato. Approximate convex decomposition of polygons. *Computational Geometry*, 35:100–123, 2006. 1, 2, 4, 5, 6, 8

[11] J.-M. Lien and N. Amato. Approximate convex decomposition of polyhedra. In *Proc. of ACM Symposium on Solid and Physical Modeling*, pages 121–131, 2007. 1, 2

[12] H. Liu, L. J. Latecki, and W. Liu. Convex shape decomposition. In *Proc. of IEEE Computer Vision and Pattern Recognition*, pages 104–124, 2010. 1, 2, 3, 4, 5, 6, 8

[13] G. Lu and A. Sajjanhar. Region-based shape representation and similarity measure suitable for content-based image retrieval. *MULTIMEDIA SYSTEMS*, 7:165 – 174, 1999. 1

[14] X. Mi and D. Decarlo. Separating parts from 2d shapes using relatability. In *Proc. of International Conference on Computer Vision*, pages 1–8, 2007. 6

[15] K. Siddiqi, K. Tresness, and B. B. Kimia. Parts of visual form: Psychophysical aspects. *Perception*, 25:399–424, 1996. 1

[16] K. Siddiqi, J. Zhang, D. Macrini, A. Shokoufandeh, S. Bouix, and S. Dickinson. Retrieving articulated 3d models using medial surfaces. *Machine Vision and Applications*, 19:261–274, 2008. 7

[17] M. Singh and D. D. Hoffman. Part-based representations of visual shape and implications for visual cognition. *Advances in Psychology*, 130:401459, 2001. 1

[18] M. Singh, G. Seyranian, and D. D. Hoffman. Parsing silhouettes: The short-cut rule. *Percept and Psychophys*, 61:636–660, 1999. 2, 3