

PAPER

Segmenting Shape Using Deformation Information

Ruiqi GUO[†], *Nonmember*, Shinichiro OMACHI^{††}, *Member*, and Hirotomo ASO^{††}, *Fellow*

SUMMARY To segment a shape into parts is an important problem in shape representation and analysis. We propose in this paper a novel framework of shape segmentation using deformation models learned from multiple shapes. The deformation model from the target image to every other image is then estimated. Finally, normalized-cut graph partition is applied to the graph constructed based on the similarity of local patches in the target image, and a segmentation of the shape is carried out. Experimental results for images from MPEG7 shape database show the effectiveness of the proposed method.

key words: shape segmentation, shape matching, deformation model, graph partition

1. Introduction

A significant problem in many computer vision and computer graphics applications is the retrieval of a well-structured shape model by segmenting it into a set of simpler shape segments, such that each segment can be semantically meaningful and easier to model. Complex shapes are often characterized as a collection of parts. Many of these objects undergo articulated motion in which each part moves independently and rigidly in its own motion model, stretching or rotating with respect to the other parts of the object. Shape segmentation finds its uses in many fields of applications. Such techniques are employed in medical imaging for segmenting shape of an organ, in computer-aided design for identifying segments of mechanical parts, reverse engineering of CAD model and in computer graphics for constructing geometric models, etc. Also, a proper segmentation of a shape can be useful in shape matching, shape analysis and object detection in that it provides a representation with semantic contexts.

In this paper, we propose a novel framework of shape segmentation using deformation models learned from multiple shapes. Our framework contains three steps. Firstly, we roughly divide a target shape into a number of patches. This is done by randomly sampling points in the internal of the target shape. Then they are clustered with k -means method. Next, correspondences between the target and reference images are found using the Shape Context shape matching [1], then the Thin Plate Spline [2] transformation model from the target shape to each of the different reference shapes

is estimated. Using the transformation models, the deformational pattern of each patch under the transformation is calculated. Finally, a graph is created, with patches as vertices and the affine-error between two patches as the weight of edges between them. This graph captures the relation of local patches by evaluation of pair-wise similarity between them. Normalized cut graph partition algorithm is then applied to the graph by taking the graph into several subsets. By merging the patches that belong to a same subset, the final segmentation is computed. The complete framework is shown in Fig. 1. To address the effectiveness and the application of the proposed method, shape segmentation experiment with a number of shape categories in MPEG7 database is shown in the latter part of the paper.

2. Related Work

There are literatures related to the shape segmentation method. Current approaches using geometry, topology and dynamic analysis models had been established and studied. Siddiqi et al. proposed a theory for representing shapes by shocks which is called shock graph model [3]. They also applied the theory to a shape matching problem by introducing a novel tree matching algorithm. Wu and Levine proposed a physics-based segmentation method [4]. An object is viewed as a charged conductor and the part boundaries are detected based on the electrical charge density distribution. A survey article [5] gives more detailed information on this issue.

In more recent works, Dey et al. proposed a method for defining features of shapes by stabilizing shape induced flows [6]. They also developed a shape matching algorithm based on the features. Gorelick et al. proposed a segmentation method by thresholding Poisson equation solution [7] as a product of Poisson equation based shape representation. Reniers and Telea proposed a hierarchical shape segmentation method by skeletonizing the shape [8]. Kageyama et al. proposed a method to decompose a shape into parts with a hierarchical structure by categorizing the edges of the graph obtained by the medial axis transform of the shape [9].

All of these methods try to segment a shape using one single image. Unfortunately, however, the ideal segmentation containing rich information from one single image can be hard. The most concerning problem of the shape segmentation lies in setting a proper criterion of geometric segmentation, which is often ambiguous and sometimes can only be assumed routinely. Utilizing only the geometric features

Manuscript received September 4, 2008.

Manuscript revised January 30, 2009.

[†]The author is with the School of Engineering, Tohoku University, Sendai-shi, 980-8579 Japan.

^{††}The authors are with the Graduate School of Engineering, Tohoku University, Sendai-shi, 980-8579 Japan.

DOI: 10.1587/transinf.E92.D.1296

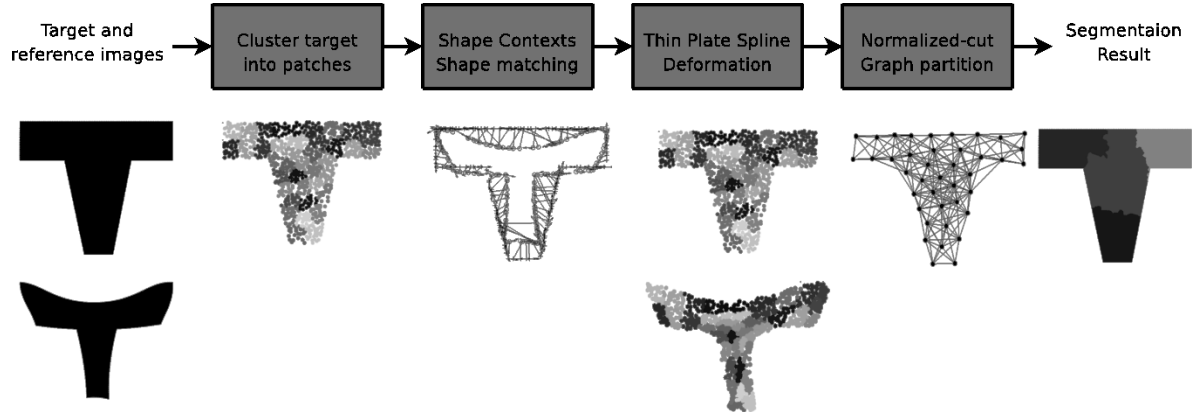


Fig. 1 An overview of the proposed shape segmentation framework. *k*-means provides an initial clustering. Shape Contexts matching results are fed to produce the TPS deformation models. A graph of local patches and their dissimilarity relations is partitioned using Normalized-cut method. Then the final segmentation result is obtained.

from one static, isolated shape, these frameworks are inherently incapable of processing semantic information that can be extracted from multiple shapes of a category.

Many of shape analysis tasks that involve perceptual shape grouping and segmentation deal with huge amount of shapes. Thus, it is reasonable to assume that multiple instances of a prototype exist for the same shape category. For example, when segmenting the shape of an organ, it is highly possible that at least one reference shape of the organ is available and useful information can be extracted from the comparison between the target shape and the reference ones. Motivated by this idea, the proposed method utilizes the deformation information with multiple images to help segmenting the target shape. We aim at segmentation of a shape into a number of parts, of which the dissimilarity between each component is most significant.

3. Methodology

3.1 Rough Initial Clustering

First, we randomly sample a number of points s_i ($i = 1, \dots, N$) from the internal of the target shape. And a pair of points that are too close to each other are removed so that the points distribute uniformly within the shape. Next, using *k*-means [10] approach, we cluster these sampled points into K clusters, $\{\mathcal{P}_i | i = 1, \dots, K\}$, where K is relatively large compared to the desired number of segmentation. Each cluster \mathcal{P}_i can be seen as a patch. Since all segmentation in later stages are merged from the initial clusters, the result cannot be more “refined” than the initial one. The final result is not sensitive to the initial clusters if the number of initial cluster is large. As a constraint, it is required that the minimal number of points in a cluster is 3, since the affine transformation estimation needs at least 3 points.

3.2 Transformation Recovery

In this step, we compute the transformation model of the target shape to each of the reference shapes. First we use Shape Contexts descriptor [1], which is one of the most popular and effective approaches of shape matching. In this stage, we deal with a shape based on the sample points taken randomly from the contour. The sample points on the contour has to be dense enough to approximate the shape. Once shape matching completes, Thin Plate Spline (TPS) deformation models are estimated for each pair of matched shapes. TPS is found to be successful in modeling morphing in various types of biological forms and equally performs well in our cases.

3.2.1 Shape Matching by Shape Context Descriptor

Shape Context is a descriptor defined on each sampled point $S_c = \{p_i | i = 1, \dots, n\}$ on the contour of the shape, and it contains a rough histogram h_i of the relative position vector from the point to all other points. Shape Context feature on point p_i is given as:

$$h_i(k) = \#\{q \in S_c | q \neq p_i, (q - p_i) \in \text{bin}(k)\},$$

where bins are in a log-polar space, which makes the descriptor more sensitive to nearer points. The distance function of two Shape Context local descriptors is defined as:

$$C_{ij} \equiv C(p_i, q_j) = \frac{1}{2} \sum_{k=1}^{N_c} \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)},$$

where N_c is the number of bins.

Note that both rotation invariant and variant versions of Shape Context can be used, depending on the application requirement. Once descriptors are calculated, matching process based on distance of descriptors is carried out using bipartite graph matching: the nodes on one side are

from the target shape and nodes from reference image on the other side. Well-known Hungarian algorithm can solve the matching problem in $O(n^3)$ time, and there are also more efficient methods of [11] that can be implemented in $O(n^2)$.

The result of the matching algorithm is a set of one-to-one correspondences between points on the target shape and one reference image. Suppose we have found in the matching step that $p_i = (x_i, y_i)$ corresponds to $u_i = (x'_i, y'_i)$, we can use the matched pairs to estimate the TPS deformation such that $T(p_i) = u_i$. The deformation function is in the form of $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$:

$$T(x, y) = (f_1(x, y), f_2(x, y)),$$

where $f_1(x, y)$ and $f_2(x, y)$ are in form of spline functions:

$$f_k(x, y) = a_0^k + a_1^k x + a_2^k y + \sum_{j=1}^n w_j^k U(\|(x, y) - (x_j, y_j)\|),$$

where $U(r) = r^2 \log r^2$, and n is the number of sample points on the contour. We can see that each f_k is specified by a group of parameters a_0^k, a_1^k, a_2^k and w_j^k ($j = 1, \dots, n$). When we solve for f_k as a spline function, let v_i be the x -coordinate of u_i for $k = 1$, or y -coordinate of u_i for $k = 2$. We consider the deformation with regularization:

$$H[f_k] = \sum_{i=1}^n (v_i - f_k(x_i, y_i))^2 + \lambda I(f_k),$$

where $I(f)$ is defined for a function f as

$$I(f) = \iint_{\mathbb{R}^2} \left(\frac{\partial^2 f}{\partial x^2} \right)^2 + 2 \left(\frac{\partial^2 f}{\partial x \partial y} \right)^2 + \left(\frac{\partial^2 f}{\partial y^2} \right)^2 dx dy.$$

When $\lambda = 0$, the deformation model realizes apparently exact interpolation, and often the model becomes distorted in this point, which is not desirable. The introduction of λ relaxes interpolation. To solve for the model with regularization, we minimize $H[f_k]$ with respect to the parameters of f_k .

3.2.2 TPS Deformation

In [2], the solution is given by linear system L , in a matrix form:

$$\begin{pmatrix} \mathbf{G} + \lambda \mathbf{I} & \mathbf{\Psi} \\ \mathbf{\Psi}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{W} \\ \mathbf{a} \end{pmatrix} = \begin{pmatrix} \mathbf{\Phi} \\ \mathbf{0} \end{pmatrix},$$

where \mathbf{W} is the column vector of w_j , \mathbf{a} is the column vector of $\{a_0, a_1, a_2\}$, $\mathbf{\Phi}$ is the column vector of v_i . And $\mathbf{G}_{ij} = U(\|(x_i, y_i) - (x_j, y_j)\|)$, $\mathbf{\Psi}$ is an $n \times 3$ matrix whose i -th row is $(1, x_i, y_i)$. [2] shows the system is nonsingular, and by inverting the system, we can solve for the weights for the spline function f_k . We combine two separate spline functions, and that gives the whole deformation $T(x, y)$.

We apply the transformation to the patches calculated in the first step to measure their corresponding positions in each reference shape. An example of shape patches applying TPS deformation is shown in Fig. 2.

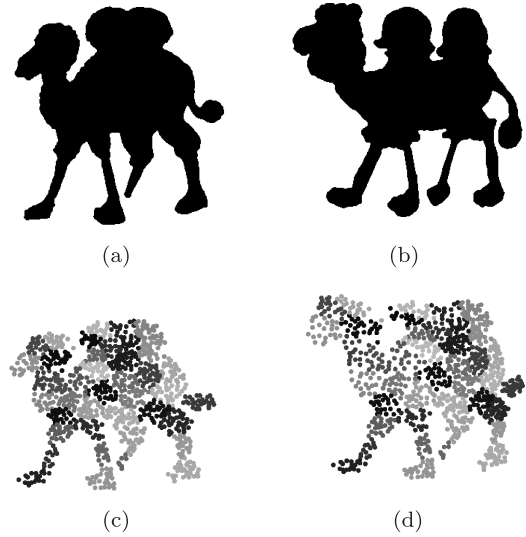


Fig. 2 (a) The target image, (b) A reference image in the same category, (c) The roughly clustered point sets, or patches in the target image, (d) Patches of the reference image under the TPS transformation estimated from (a) to (b).

3.3 Partition of Similarity Graph

The last step of our framework involves graph partition. First, a similarity graph is established based on the deformation behavior of each patch: $\mathbf{G} = \langle V, \mathbf{E}, \mathbf{W} \rangle$, which satisfies the following:

1. $V = \{\mathcal{P}_i | i = 1, \dots, K\}$, each corresponds to a patch in the target shape;
2. $\langle \mathcal{P}_i, \mathcal{P}_j \rangle \in \mathbf{E}$, namely, patch \mathcal{P}_i and \mathcal{P}_j are connected, if the Hausdorff distance between \mathcal{P}_i and \mathcal{P}_j satisfies ϵ -neighborhood or k -nearest neighbor criterion;
3. $\mathbf{W}_{ij} = \exp\left(\frac{-\varepsilon(\mathcal{P}_i, \mathcal{P}_j)^2}{2\sigma^2}\right)$, the weight on edge $\langle \mathcal{P}_i, \mathcal{P}_j \rangle$ comes from Gaussian similarity function and the affine error ε of \mathcal{P}_i and \mathcal{P}_j , which we will discuss later.

Here, the Hausdorff distance between $\mathcal{P}_i, \mathcal{P}_j$:

$$H(\mathcal{P}_i, \mathcal{P}_j) = \min\{\|p_i - p_k\| | p_i \in \mathcal{P}_i, p_k \in \mathcal{P}_j\},$$

This similarity graph essentially models the deformational patterns of patches. Because patches are small local regions, affine transformation is enough to model their deformations in reference shapes. For every pair of patches that are reasonably close, their dissimilarity in deformational patterns are reflected in their affine error. Hence, between two patches, the larger the affine error is, the more dissimilar they are. The idea behind the construction and detail of the graph is shown in Fig. 3. After the similarity graph is constructed, we perform normalized cut graph partition algorithm, which segments the graph into subsets.

3.3.1 Affine Error

We use affine error between two patches that represents the

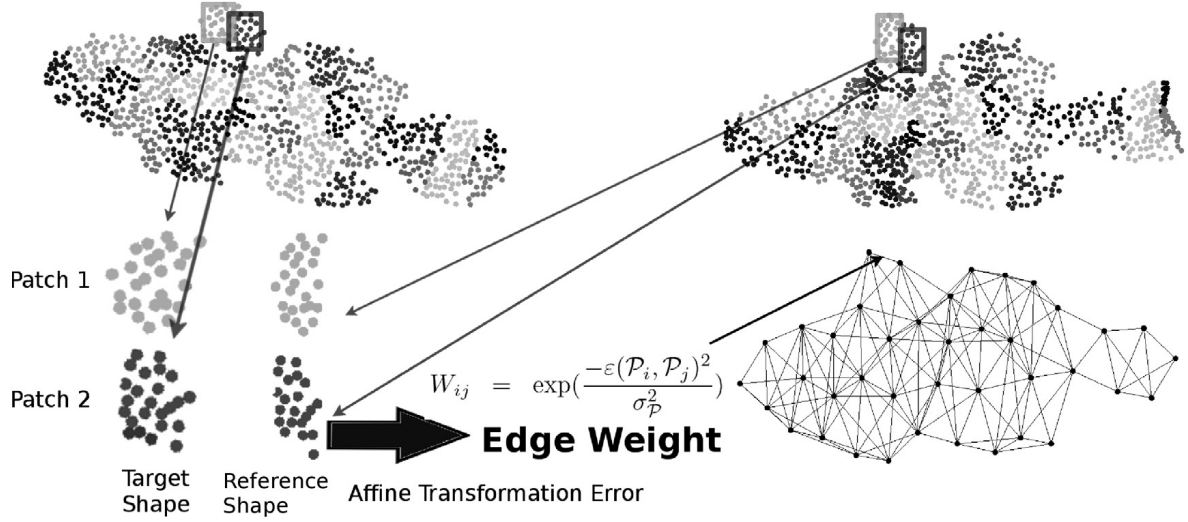


Fig. 3 The construction of the graph to be partitioned from the affine error between two patches and their correspondence in reference shapes.

deformation difference between them with considering the similarity of the corresponding patches throughout all reference images. We start with the estimation of affine transformation. The affine transformation matrix of a cluster of points \mathcal{P} to its counterpart in a reference image \mathcal{Q} is given as:

$$\mathbf{A}(\mathcal{P}, \mathcal{Q}) = (\mathbf{Q}^+ \mathbf{P})^T,$$

where \mathbf{P} and \mathbf{Q} are the homogeneous coordinates of patch \mathcal{P} (here, we view it as point cluster) and \mathcal{Q} , respectively, i.e.,

$$\mathbf{P} = \begin{pmatrix} 1 & p_{1x} & p_{1y} \\ \vdots & \vdots & \vdots \\ 1 & p_{n_p x} & p_{n_p y} \end{pmatrix}, \mathbf{Q} = \begin{pmatrix} 1 & q_{1x} & q_{1y} \\ \vdots & \vdots & \vdots \\ 1 & q_{n_q x} & q_{n_q y} \end{pmatrix},$$

\mathbf{Q}^+ is the pseudoinverse of \mathbf{Q} , that satisfy $\mathbf{Q}\mathbf{Q}^+\mathbf{Q} = \mathbf{Q}$ and n_p is the number of points in patch \mathcal{P} .

We now consider the affine error between patches \mathcal{P}_i and \mathcal{P}_j of target image. We denote \mathcal{P}_i 's corresponding patch in reference image m as \mathcal{P}_i^m . For every two patches \mathcal{P}_i and \mathcal{P}_j , we define their pair-wise affine error as:

$$\varepsilon(\mathcal{P}_i, \mathcal{P}_j) = \frac{1}{2M} \sum_{m=1}^M \left(\sum_{k=1}^{n_{p_i}} \|p_{ik}^m - \mathbf{A}(\mathcal{P}_i, \mathcal{P}_i^m) p_{ik}\| + \sum_{k=1}^{n_{p_j}} \|p_{jk}^m - \mathbf{A}(\mathcal{P}_j, \mathcal{P}_j^m) p_{jk}\| \right),$$

where M is the number of reference images, and p_{ik}^m ($k = 1, \dots, n_{p_i}$) and p_{ik} are column vectors with 1 and sampled points in patch \mathcal{P}_i^m and ones in \mathcal{P}_i , respectively.

3.3.2 Normalized Cut

Consider first two-way graph partition to introduce k -way graph partition. Let $\{A, B\}$ be a partition of vertices V in

graph \mathbf{G} , a cut in a graph is sum of the weights of the edges connecting two parts:

$$cut(A, B) = \sum_{u \in A, v \in B} \mathbf{W}_{uv}.$$

In [12], Shi and Malik proposed *normalized cut (Ncut)* as the graph partition to minimize the criterion:

$$Ncut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)},$$

where $assoc(A, V)$ is the total weights of edges connecting vertices in A to all vertices, written as

$$assoc(A, V) = \sum_{u \in A, t \in V} \mathbf{W}_{ut}.$$

Although finding normalized cut is NP-complete, it has efficient approximate discrete solution when it is relaxed to real value domain. Let \mathbf{W} be the weight matrix of \mathbf{W}_{ij} , \mathbf{D} be a diagonal matrix with $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$, and \mathbf{x} be the indicator vector of partition V :

$$\mathbf{x}_i = \begin{cases} 1, & v_i \in A \\ -1, & v_i \in B \end{cases}.$$

In [12], it is proved that finding real value solution for Ncut problem can be done by getting the second smallest eigenvector of the generalized eigenvalue system:

$$(\mathbf{D} - \mathbf{W})\mathbf{y} = \mathbf{D}\lambda\mathbf{y},$$

where

$$\mathbf{y} = (\mathbf{1} + \mathbf{x}) - \frac{\sum_{x_i > 0} \mathbf{D}_{ii}}{\sum_{x_i < 0} \mathbf{D}_{ii}} (\mathbf{1} - \mathbf{x}),$$

and $\mathbf{1}$ is a vector that all elements are one. It is solved by transforming it to a standard eigenvalue system:

Algorithm 1 *k*-way Normalized-cut algorithm

Input: the number *k* of clusters, the similarity graph $G = \langle V, E, W \rangle$, constructed based on *k*-NN or ϵ -neighborhood.

Compute the normalized graph Laplacian of graph *G*: $L_{\text{norm}} = D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}$.

Solve for the eigenvectors corresponding to the *k* smallest eigenvalues of L_{norm} , $u_1, u_2 \dots u_k$.

Let $Y \in \mathbb{R}^{K \times k}$ be the matrix composed of $u_1, u_2 \dots u_k$.

Let $\{y_i \in \mathbb{R}^k | i = 1 \dots n\}$ be *i*-th row vector of matrix *Y*.

Cluster $\{y_i | i = 1 \dots n\}$ into *k* clusters $C_1, C_2, \dots C_k$ using *k*-means algorithm.

Output: Clusters $C_1, C_2 \dots C_k$

$$D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}z = \lambda z,$$

where $z = D^{\frac{1}{2}}y$, and $D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}$ is called the normalized graph Laplacian of graph *G*.

k-way normalized cut is based on the same idea, which takes the eigenvectors corresponding to the *k* smallest eigenvalues of the normalized graph Laplacian for some *k*, and then use another (e.g. *k*-means) algorithm to cluster them by their respective *k* components. The trick behind the algorithm is to change the representation of original data points in \mathbb{R}^n into points in \mathbb{R}^k that enhances the cluster properties. The complete algorithm is given as Algorithm 1.

For each $i = 1, \dots, K$, $\mathcal{P}_i \in C_j$ indicates patch \mathcal{P}_i belongs to *j*-th component. By graph partition, the patches are grouped into the desired number of components. As post-processing, we group every pixel in the internal of the target image into the component that its closest sample point belongs to.

4. Experiments

We used for statistic shape segmentation experiment shapes from MPEG-7 Shape Silhouette Database CE-Shape-1 part B [13], which provides 20 samples per category. For a shape to be segmented, we prepared 9 images in a category as reference and one target image, forming a test set, as shown in Fig. 4. Next, the method proposed in the previous section was applied. In the experiment, we used similarity graphs based on *k* nearest neighbor criterion. Experiment with 15 categories of images was carried out. Some results are shown in Fig. 5.

In the experiment, the number of nearest neighbor was $k = 10$ and the initial number of clusters was $K = 60$, which were determined experimentally. The parameters need not to be exact and usually a large *K* and *k* produce better result since this will allow the method to generate a more subtle similarity graph. However, since *K* is the number of nodes in the similarity graph, computation speed should also be considered when choosing *K*. Solving eigenvectors for the Laplacian of the similarity graph will become expensive if a large value of *K* is used. The value of σ for Gaussian similarity function was set to 10% of the total range of distance function $\mathcal{E}(\mathcal{P}_i, \mathcal{P}_j)$, the same as it was recommended in [12].

While some segmentation results may contain faults in the sense of human perception, they do not disprove the validity of the proposed method. The general problem of unsupervised shape segmentation is not trivial and it is highly possible that there exist multiple grouping from different

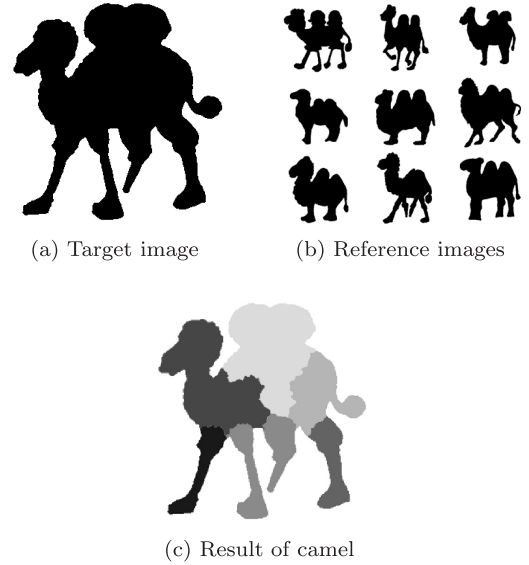


Fig. 4 A target image, reference images and segmentation result. Number of segments is seven.

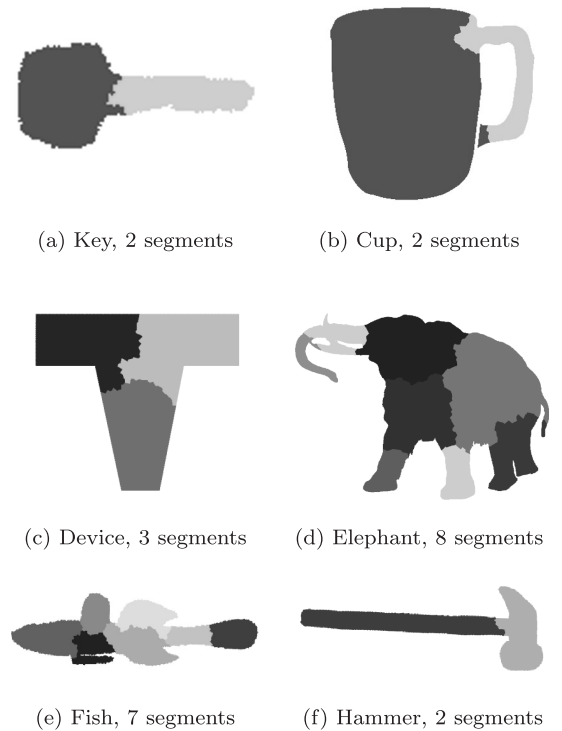


Fig. 5 Segmentation output of various images from MPEG7 CE Part B.

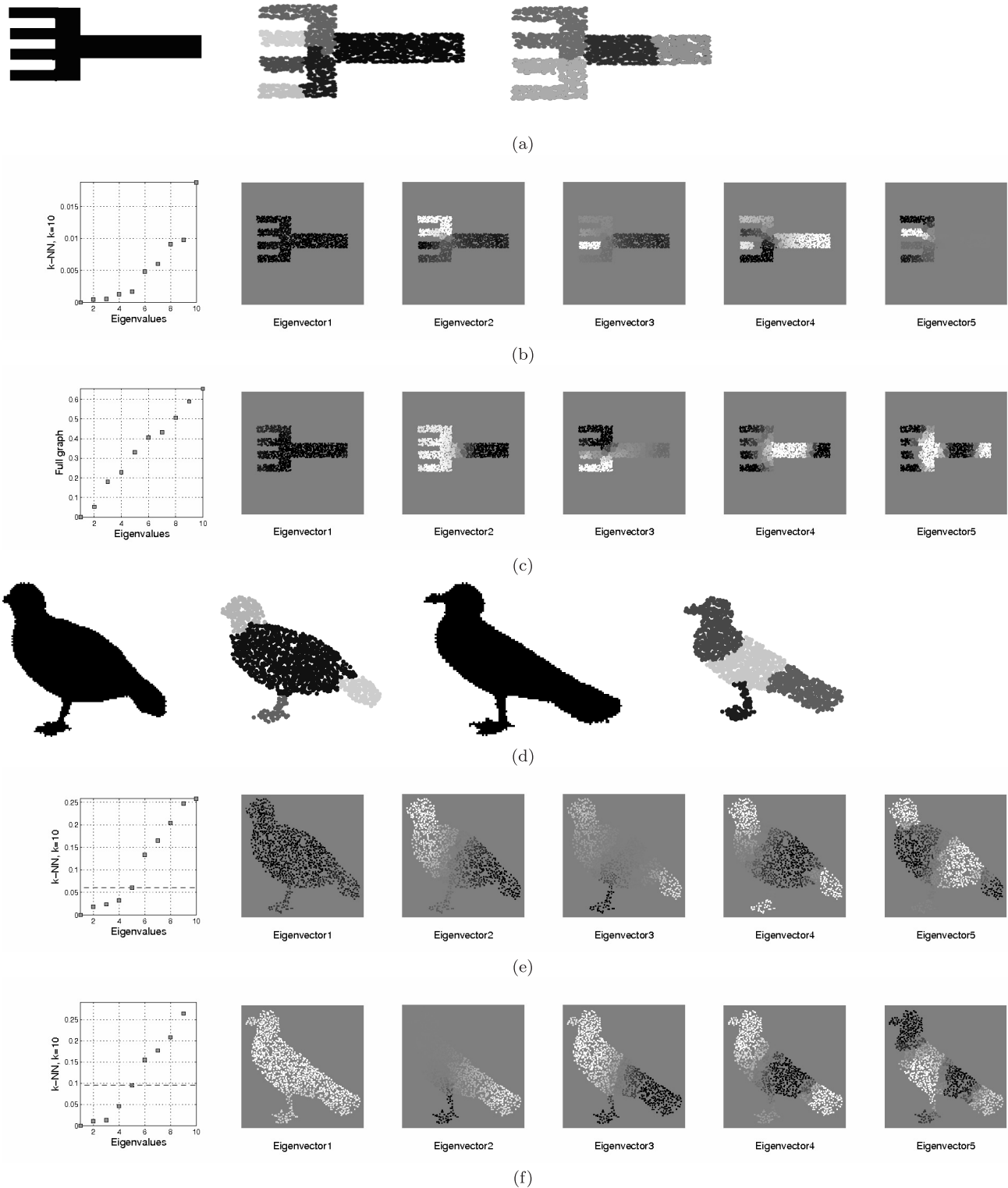


Fig. 6 (a) The target image “fork” and its segmentation based on k -NN and fully-connected graph, respectively; (b) and (c): Its eigenvalues and eigenvectors of the corresponding k -NN graph with $k = 10$ and that of the fully connected graph; (d) The segmentation of two target images from the same category “bird”; (e) and (f): Their eigenvalues and eigenvectors of under k -NN similarity graph.

semantic hierarchical level that the segmentation can take cue from. In the experiment, except simple shapes such as “key,” “cup” and “hammer” in Fig. 5, in which segmenta-

tion is straight-forward, many complex objects can be segmented in different ways. It is possible that two parts which should be grouped together in geometric point of view may

be separated since a shape share dissimilar deformational patterns. It should be noted that error on shape segmentation can somewhat be eliminated by tweaking parameters K , k and σ . However, we used uniform parameters to show that the proposed method does not heavily depend on parameter settings.

In the experiment with MPEG-7 database, we visualized the eigenvalues and eigenvectors of the Laplacian of the similarity graph, as illustrated. To visualize an eigenvector u_i , we assigned every patch a grayscale value, based on its value in the eigenvector, that is, the color of \mathcal{P}_j in u_i is assigned based on $u_i(j)$, with negative values represented by brighter colors and positive values represented by darker colors. Figure 6 (b) and (c) show the effect of similarity graphs of k -NN and fully connected.

Segmentation of different target images in the same category of shapes are similar in nature in that the eigenvalues and corresponding eigenvectors of their similarity graphs are alike. In Fig. 6 (e) and (f), two target images from the same category, although different in shape, share similar eigenvectors, which indicates the robustness of the proposed method to variances. However, we observed that in cases where reference images contain such drastic difference (e.g. reference images are taken from other categories of shapes) that its TPS deformation model is virtually invalid and meaningless, the algorithm fails to produce informative segmentation, and the result become resemble to what directly applying k -means to data points gives.

Several important practical details affects the performance of Ncut clustering algorithm, one of which is choosing the appropriate number S of components. For the proposed method, the number of components S can either be assigned manually or automatically detected using eigengap heuristic. This method observes the magnitude of eigenvalues of graph Laplacian, and chooses the number of clusters to be k , such that $k + 1$ -th smallest eigenvalue λ_{k+1} is significantly larger than $\lambda_1, \dots, \lambda_k$, or, the gap between k -th and $k + 1$ -th smallest eigenvalue is relatively large, such as that plotted in Fig. 6 (e) and (f). Each eigenvector carries information about the clusters. We observed that the eigenvectors starting from that corresponds the smallest eigenvalue, namely u_2, \dots, u_k , each eigenvector roughly serves to separate two cluster from each other. For example, in Fig. 6 (e), eigenvector 2 discriminate the bird's head while eigenvector 3 discriminate the bird's tail from the rest of the body. However, as the eigenvalue grows larger, its eigenvector becomes less informative and therefore should not be used in segmentation.

The eigengap approach finds its root in perturbation theory [14]: the number of 0 eigenvalue in a graph with k disconnected components is k , and its λ_{k+1} is greater than 0. For the above reason, we can almost always find meaningful and stable segmentation if eigengap heuristic is followed. The segmentation number S in the proposed method is found by setting the least number of clusters and then detect the eigengap where $|\lambda_{k+1} - \lambda_k| > \gamma|\lambda_k - \lambda_{k-1}|$ occurs, and γ is a parameter usually set to 2. However, we find in

practice that the number of clusters for the most stable segmentation usually falls below k when k is large since there are cases where one eigenvector separates more than two clusters. In the experiment, we used the number of clusters $k - 1$ for k greater than 5.

5. Conclusion and Outlook

We have described a novel framework for shape segmentation, which is based on the learning of deformation models from multiple images instead of pure geometric analysis on one single shape. Represented by patches, the transformation model of the target shape to reference images is estimated. Using the affine error between patches, the deformational patterns of patches are contained into a similarity graph of patches and the segmentation is achieved by partitioning the similarity graph by normalized-cut algorithm. Experimental results and analysis on shapes from MPEG-7 database show that the proposed method is robust under variance and gives good performance in segmentation. The proposed segmentation method represents a shape as clusters of sample points in the interior of the shape. This sampling representation greatly enhances the robustness since we do not process small shape flaws and captures a general image of how the shape is like.

For further extensions and applications of the method described in this paper, the generalization of the proposed method in 3D shapes is required. Moreover, applications other than shape segmentation can be developed, such as segmentation based recognition algorithm (an example of such algorithms can be found in [6]). Finally, we consider that developing a good quantitative measure of shape segmentation is also an important future work.

References

- [1] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.24, no.4, pp.509–522, April 2002.
- [2] J. Duchon, "Splines minimizing rotation invariant seminorms in sobolev spaces," *Constructive Theory of Functions of Several Variables*, vol.1, pp.85–100, 1977.
- [3] K. Siddiqi, A. Shokoufandeh, S.J. Dickinson, and S.W. Zucker, "Shock graphs and shape matching," *Proc. Int. Conf. on Computer Vision*, pp.222–229, 1998.
- [4] K. Wu and M.D. Levine, "3D part segmentation using simulated electrical charge distributions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.19, no.11, pp.1223–1235, Nov. 1997.
- [5] S. Loncaric, "A survey of shape analysis techniques," *Pattern Recognit.*, vol.31, pp.983–1001, 1998.
- [6] T.K. Dey, J. Giesen, and S. Goswami, "Shape segmentation and matching with flow discretization," *Workshop on Algorithms and Data Structures*, pp.25–36, 2003.
- [7] L. Gorelick, M. Galun, E. Sharon, R. Basri, and A. Brandt, "Shape representation and classification using the Poisson equation," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, vol.2, pp.61–67, 2004.
- [8] D. Reniers and A. Telea, "Skeleton-based hierarchical shape segmentation," *Proc. IEEE Int. Conf. on Shape Modeling and Applications*, pp.179–188, June 2007.

- [9] K. Kageyama, S. Omachi, and H. Aso, "Parts decomposition of silhouette shape and category classification algorithm," IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J91-D, no.5, pp.1305–1313, May 2008.
- [10] J.A. Hartigan, Clustering Algorithms, Wiley, 1975.
- [11] R. Jonker and A. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," Computing, vol.38, no.4, pp.325–340, March 1987.
- [12] J. Shi and J. Malik, "Normalized cuts and image segmentation," Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp.731–737, June 1997.
- [13] L.J. Latecki, R. Lakaemper, and U. Eckhardt, "Shape descriptors for non-rigid shapes with a single closed contour," Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp.424–429, 2000.
- [14] U. von Luxburg, "A tutorial on spectral clustering," Statistics and Computing, vol.17, no.4, pp.395–416, Dec. 2007.



Hirotomo Aso received his B.E., M.E., and Ph.D. degrees in Electrical Engineering from Tohoku University, Japan, in 1968, 1970, and 1974, respectively. He was with the Department of Information Engineering, Tohoku University in 1973, and later joined the Faculty of Engineering, Nagoya University from 1979 to 1986. He is now a professor at Tohoku University. He is presently engaged in research of character pattern recognition, cellular automata, concurrent program schema, and parallel processing. He

has received the Young Researchers' Award of IEICEJ in 1978 and the Achievement Award (Gyoseki-sho) of IEICEJ in 1992. Dr. Aso is a member of the Information Processing Society of Japan, the Japanese Society of Artificial Intelligence, IEEE, EATCS, and ACM.



Ruiqi Guo is an undergraduate in the Department of Computer Science, Zhejiang University, P.R.China. He is currently an exchange student at the School of Engineering, Tohoku University, Japan. His research interests include the field of image processing, computer vision, pattern recognition and machine learning.



Shinichiro Omachi received his B.E., M.E. and Doctor of Engineering degrees in Information Engineering from Tohoku University, Japan, in 1988, 1990 and 1993, respectively. He has worked as a research associate at the Education Center for Information Processing at Tohoku University from 1993 to 1996. He is now an associate professor at the Graduate School of Engineering, Tohoku University. From 2000 to 2001, he has been a visiting associate professor at Brown University. His research interests

include pattern recognition, computer vision, information retrieval, data mining and parallel processing. He has received MIRU Nagao Award and IAPR/ICDAR Best Paper Award in 2007. Dr. Omachi is a member of the IEEE, the Information Processing Society of Japan, and the Japanese Society of Artificial Intelligence.