Segmentation of Point Sets

Ichitaro Yamazaki*, Vijay Natarajan†, Zhaojun Bai‡, and Bernd Hamann§
Institute for Data Analysis and Visualization (IDAV)
Department of Computer Science
University of California
Davis, California 95616
{yamazaki*, bai‡, hamann§}@cs.ucdavis.edu
vijayn@ucdavis.edu†

ABSTRACT

We introduce a new technique to segment a point set, where the connectivity between points is not known. Our technique is based on multi-level segmentation of the point set and is computationally efficient. We apply our technique to point sets obtained from scans of 3D models and demonstrate that it results in good segmentations.

1. INTRODUCTION

A point set is a collection of points, where each point is defined by a fixed-length coordinate vector, but the connectivity between points is not known. One example of a point set is a 2D image, which is defined by the two pixel (i, j). Other examples include sets of protein molecules and handwritten characters represented as points in high dimensional spaces.

The problem of point set segmentation is to segment the points within a point set such that the points in one segment are strongly related while the points in different segments are weakly related. The basic framework of our multi-level segmentation approach follows these steps:

1. Compute the distance between points along an approximate surface (geodesic distance approximation).
2. Collapse the original point set to construct a coarse point set.
3. Define a similarity measure between points in the coarse point set.
4. Segment the coarse point set, using the normalized cut (NC) technique.
5. "Uncoarsen" the segments of the coarse graph to obtain the segmentation of the original point set.

2. DISTANCE COMPUTATION

The distance between points can be computed as the shortest distance between points along an approximated surface, which is sometimes referred as the geodesic distance of the points [2]. The geodesic distance of points is approximated as follows:

1. Compute the Euclidean distance between points.
2. Construct a graph $G(V, E)$ by connecting each given point to its k-nearest "neighbor points."
3. Compute the shortest path between all pairs of points in G and store the distance in an $n \times n$ matrix GD, where $n = |V|$ and $GD(i, j)$ is the approximated geodesic distance between v_i and v_j.

For our experiments, seven nearest neighbors are connected to construct the graph G.

3. COARSENING

The purpose of coarsening is to reduce the number of vertices in the graph by collapsing multiple vertices into a single "supernode." At the same time, the structure of the original graph needs to be preserved in the coarse graph so that a segmentation of the coarse graph reflects that of the original graph. With a good coarsening of the original graph, the computational cost to segment the point set can be reduced significantly while the quality of the segmentation is maintained.

Many of the previous coarsening techniques are based on the matching and collapsing of vertices such as "strong edge matching," "weak edge matching," or "random edge matching" where matched points are collapsed into a supernode. We propose a coarsening approach based on the notion of local features defined by local maxima of a function defined over the point set [3]. We perform these steps:

1. Construct a diagonal matrix $D = diag(d_i)$, where $d_i = \sum_{j \neq i} GD(i, j)$. A vertex v_i is a local maximum if $d_i > d_j$ for all neighbors v_j of v_i.
2. Sort V according to D in descending order, i.e., $d_i \geq d_j$ if $v_i, v_j \in V$ and $i < j$.
3. Create a supernode containing v_1. Notice that v_1 is a local maximum.
4. Iterate over all $v_i \in V$ in descending order.
   (a) Create a supernode containing v_i if v_i is a local maximum.
   (b) Collapse all neighbors v_j of v_i, that have not been collapsed yet, to the supernode that contains v_i.

5. Construct a coarse graph $\hat{G}(\hat{V}, \hat{E})$, where $\hat{V}$ consists of supernodes. Edge weights between the supernodes consist of four weighted components: the distance between the local maximum that represents the supernodes F_1; the relative size of removed feature (maximum difference of D within the supernodes) F_2; the cut between the two supernodes (sum of all edge distances between supernode) F_3; and measure of common boundary between the supernodes (number of crossing edges) F_4. The similarity measure matrix $\hat{W} = (\hat{w}(v_i, v_j))$ is computed as

$$\hat{W} = \sum_{i=1}^{4} e^{-\frac{\alpha_i \times F_i}{max\{|F_i|\}}}.$$

4. HIERARCHICAL NORMALIZED CUT

Given a graph $\hat{G} = \hat{G}(\hat{V}, \hat{E})$ and a corresponding similarity matrix $\hat{W}$, the normalized cut step constructs a 2-way disjoint segmentation of $\hat{V} = \hat{V}_1 \cup \hat{V}_2$ and $\hat{V}_1 \cap \hat{V}_2 \neq 0$ such that the total normalized cut

$$\frac{cut(\hat{V}_1, \hat{V}_2)}{assoc(\hat{V}_1, \hat{V})} + \frac{cut(\hat{V}_1, \hat{V}_2)}{assoc(\hat{V}_2, \hat{V})}, \tag{1}$$

where

$$cut(\hat{V}_1, \hat{V}_2) = \sum_{\hat{v}_1 \in \hat{V}_1, \hat{v}_2 \in \hat{V}_2} \hat{w}(\hat{v}_1, \hat{v}_2)$$

and

$$assoc(\hat{V}_i, \hat{V}) = \sum_{\hat{v}_i \in \hat{V}_i, \hat{v} \in \hat{V}} \hat{w}(\hat{v}_i, \hat{v}),$$

is minimized [1]. It can be shown that minimizing equation (1) also maximizes

$$\frac{assoc(\hat{V}_1, \hat{V}_2)}{assoc(\hat{V}_1, \hat{V})} + \frac{assoc(\hat{V}_1, \hat{V}_2)}{assoc(\hat{V}_2, \hat{V})}. \tag{2}$$

Thus, the cut between $\hat{V}_1$ and $\hat{V}_2$ is minimized while the associations within $\hat{V}_1$ and $\hat{V}_2$ are maximized. It can be shown that the above minimization problem is equivalent to finding the eigenvector corresponding to the second smallest eigenvalue (Fiedler vector) of the generalized eigenvalue system, i.e., finding n-vector y such that

$$(\hat{D} - \hat{W})y = \lambda \hat{D} y, \tag{3}$$

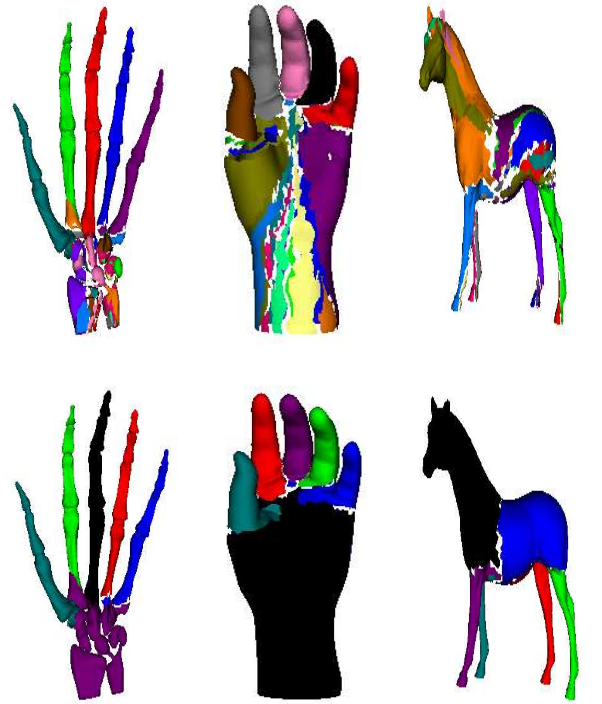
where $\hat{D}$ is a diagonal matrix such that $\hat{d}_i = \sum_{i \neq j} \hat{w}(\hat{v}_i, \hat{v}_j)$. The normalized cut of $\hat{V}$ is constructed such that $\hat{v}_i \in \hat{V}_1$ if $y(i) \leq \sigma$ and $\hat{v}_i \in \hat{V}_2$ otherwise for some threshold σ. We apply 2-way segmentation recursively until the normalized cut value between two segments is greater than a specified threshold.

5. REFINEMENT

To construct a segmentation of the original point set from that of the coarsened point set, we replace the supernode with the points that have been collapsed into the supernode. We will examine more sophisticated refinement techniques to generate a better segmentation in the future.

6. RESULTS

We presents results for the point sets describing 3D surface models. Figure 1 shows the segmentation of three sample point sets. For the similarity matrix used in these results, distances between local maxima $F1$ and the relative sizes of removed features $F2$ are considered, using weights 0.1 and 0.7, respectively, i.e., $W = e^{-\frac{0.1 \times F1}{max(|F1|)}} + e^{\frac{0.7 \times F2}{max(|F2|)}}$



	# of points	# of local max	# of segments
skeleton	6,000	35	6
hand	6,000	28	6
horse	4,000	32	6

Figure 1: Segmentation of point sets obtained from range scans of 3D surface models. Top row shows the result after coarsening step. Bottom row shows segments after normalized cut step.

7. CONCLUSION

We have described a new segmentation technique for point set data. Our results have shown that our technique produces good segmentations of point sets of 3D model surface. We will test our technique on higher-dimensional data sets such as point sets of molecules and hand-written characters.

8. REFERENCES

[1] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[2] J. B. Tenebaum and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[3] E. Zhang, K. Mischainkow, and G. Turk. Feature-based surface parameterization and texture mapping. *ACM Transaction on Graphics*, 24(1):1-27, 2005.