Feature Detection Using Curvature Maps and the Min-Cut/Max-Flow Algorithm

Timothy Gatzke[1] and Cindy Grimm[1]

Washington University in St. Louis, St. Louis MO 63130, USA

Abstract. Automatic detection of features in three-dimensional objects is a critical part of shape matching tasks such as object registration and recognition. Previous approaches often required some type of user interaction to select features. Manual selection of corresponding features and subjective determination of the difference between objects are time consuming processes requiring a high level of expertise. The *Curvature Map* represents shape information for a point and its surrounding region and is robust with respect to grid resolution and mesh regularity. It can be used as a measure of local surface similarity. We use these curvature map properties to extract feature regions of an object. To make the selection of the feature region less subjective, we employ a min-cut/max-flow graph cut algorithm with vertex weights derived from the curvature map property. A multi-scale approach is used to minimize the dependence on user defined parameters. We show that by combining curvature maps and graph cuts in a multi-scale framework, we can extract meaningful features in a robust way.

1 Introduction

Advances in three-dimensional (3-D) scanning capability are providing ready access to 3-D data. Automatic detection of features in 3-D objects is critical for tasks such as object registration and recognition. For example, identifying corresponding regions between two similar surfaces is a necessary first step toward alignment and registration of those surfaces. A fundamental question is: What constitutes a feature? Man-made objects often have well-defined features such as edges, but features on natural shapes, such as the wrist bones shown in Figure 1, are more subjective. Furthermore, such shapes can have subtle variations, the importance of which may not be obvious.

We aim to detect subtle shape features in a robust way with a fully automated process. The types of features we expect to be useful are peaks, pits, ridges, and valleys. Important features may be of various sizes within one object. We need not (in fact, cannot) detect every feature, and the features we do detect may or may not be unique. We just need to identify enough features to resolve any ambiguities during shape matching. It is desirable for feature detection to be consistent, robust, independent of the mesh resolution, and relatively insensitive to noise.

Previous approaches often required some type of user interaction to select features. Manual selection of corresponding features and subjective determination of the difference between objects are time consuming processes requiring a high level of expertise. In contrast, our approach is entirely automatic.

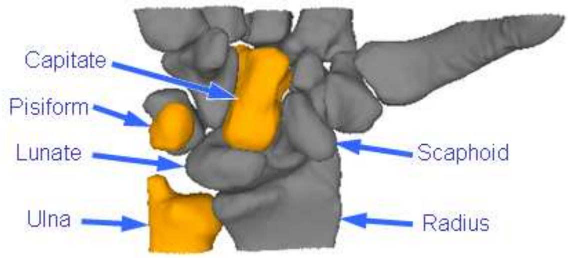

Fig. 1. Bones making up the human wrist. Natural objects have subtle shape variations that are challenging to characterize.

1.1 Approach

In this paper we present a feature detection algorithm based on the *Curvature Map* [1], which at a point represents shape information for the point and its surrounding region. A min-cut/max-flow graph cut algorithm, popular for image segmentation tasks, is employed to identify features at various scales. Results from multiple graph cuts are combined in a novel manner to produce a final feature set. A two-step multi-scale approach eliminates the need for user interaction, and for tuning parameters based on a particular application. This algorithm can extract meaningful features in a robust way.

Section 2 focuses on related work in object recognition, feature detection, and segmentation. In Section 3 we give an overview of the algorithm. Details omitted due to space constraints can be found in [2]. Results for various shapes, and conclusions and possible areas for future work, are presented in Sections 4 and 5 respectively.

2 Related Work

The two main areas of research related to this work are shape representations or signatures, and feature segmentation. Object recognition, correspondence, and registration often rely on similarity measures to quantify the similarity or dissimilarity between objects by computing distances between shape representations, such as sets of points, feature vectors, histograms, signatures, or graph representations. Methods that are more applicable to 2D images rather than 3D object representations will not be discussed here. See [3] for a survey of methods applied to medical images.

Graph representations, such as skeletons [4, 5] and multi-resolution Reeb graphs [6], like algorithms based on point sets [7, 8], can be useful for computing similarity and registration. But these methods are primarily global rather than local and often can be sensitive to the distribution of the mesh points.

Signatures may be global or local, and provide a compact representation that results in more efficient comparison at the expense of their ability to discriminate shape. Methods used for shape retrieval, such as shape distributions [9], spin images [10], and spherical spin images [11], tend to be global measures, and generally provide limited discrimination between similar shapes.

Signatures of a more local nature include statistical signatures [12] and shape contexts [13], but the use of local point-to-point distances and angles, and sampling of points respectively, limits the suitability of these methods for detailed shape comparison. The point fingerprint [14], which defines an irregularity measure for geodesic circles around a point, and the surface curvature signature [15] rely on high curvature feature points. Unlike these approaches, we are looking for subtle shape differences that require more than signatures just at 'interesting' points.

Feature regions can be extracted based on critical points (peaks, pits, and passes) and associated ridge and valley lines. In [16], smoothing was required as a preprocessing step. Peak (pit) areas surrounded by valley (ridge) cycles then provide the candidate feature areas to be selected interactively. The uncertainty as to an appropriate amount of smoothing and the narrow definition of a feature are drawbacks to this approach.

Volume decomposition based on topology [17] or morphological tools [18] provides volumetric features rather than surface features. Surface segmentation methods, which identify local regions of an object, have been based on the sign of the curvature [19], isosurfaces and extreme curvatures [20], and watersheds of a curvature function [21, 22]. Methods that identify salient features [23, 24] have also been developed. However, these methods do not yield the types of features we are interested in for shape matching.

Graph cut algorithms have been used to segment images [25] and medical datasets [26]. They are effective at assigning the vertices of a graph to either a feature (foreground) or background set, based on graph properties such as the gradient of the image intensity. Some of these methods employ an interactive step, where the user identifies feature and background seed points, to guide the algorithm to the objects that are to be separated. By treating our mesh as a graph, we can apply the graph cut algorithm and identify features based on the resulting segmentation.

3 Feature Detection Method

The basic feature shapes we are looking for include the peak, pit, ridge, and valley. The common link between these features is the dependence on the magnitude of the mean curvature. The curvature map [1] provides a context for each point that can be used to define a local shape property to help identify these features.

3.1 Local Shape Property

For a vertex p, the 1-D curvature map, $Kmap(p)$, is defined by two curves representing the average mean and Gaussian curvature as functions of distance from the vertex. We will refer to these curves as $Mean(Kmap(p))$ and $Gauss(Kmap(p))$ respectively. We define our local shape property S as

$$S(p) = \int_0^R Mean(Kmap(p))(r)dr$$

where R represents the radius corresponding to the maximum feature size.

Algorithm 1 Multi-Scale Feature Detection

Read Curvature Map (K_{map}) for Mesh M
for K_{map} radius R from R_{min} to R_{max} **do**
  Compute S as the integral of the K_{map} mean curvature component from 0 to R
  for a range of weight factor α **do**
    Create graph cuts $C_{abs}, C_{pos}, C_{neg}$ on the absolute, positive, and negative values of S
    Identify the features in $C_{abs}, C_{pos}, C_{neg}$
    for each vertex v in Mesh M **do**
      Count feature occurrences $N_{abs}, N_{pos}, N_{neg}$ in $C_{abs}, C_{pos}, C_{neg}$
    end for
    for each edge **do**
      count how many times both endpoints occur in the same region
      Note: Used to generate edge weights for the later max-flow/min-cut runs
    end for
  end for
end for
for a range of weight factor α **do**
  Create graph cuts $C_{abs}, C_{pos}, C_{neg}$ from normalized counts $N_{abs}, N_{pos}, N_{neg}$
  Identify and merge features from $C_{abs}, C_{pos}, C_{neg}$ into composite feature sets $G_{abs}, G_{pos}, G_{neg}$
end for
Merge G_{neg} and G_{pos} into G_{abs} to create the Master Feature Set G

We also considered functions based on the Gaussian curvature component of the curvature map, but given a suitable threshold, the mean curvature function gave the most consistent identification of the features in our test cases.

Although the local shape property often highlights the expected features, finding an appropriate threshold requires manual adjustment, and the results still depend on the curvature map radius R. In addition, no single threshold could extract both the positive curvature features (peak and ridge) and the negative curvature features (pit and valley). These factors motivated our search for an improved feature detection approach.

3.2 Multi-Scale Algorithm

Combining our local shape property with the min-cut/max-flow graph cutting technique [25] creates a multi-scale approach for feature detection as presented in Algorithm 1. Varying the curvature map radius R detects features at different scales, while increasing the weights by a scale factor α detects less prominent features.

Ranges for these parameters are discussed in [2]. For our examples, we use 8 K_{map} radii cross 10 scale factors, resulting in 80 graph cuts each for the absolute value, positive, and negative of the shape property, plus 30 in the second step, for a total of 270 graph cuts. Fortunately, the graph cut algorithm is very efficient, with the 270 graph cuts on a 10,000 vertex mesh taking less than 40 seconds on a 2.8GHz Pentium 4 processor.

Re-running the graph cut algorithm on the occurrence count maintains focus on the strongest features. Once we have created the graph cut, we form features from contiguous groups of vertices in the feature set of the graph cut. For combining sets of features,

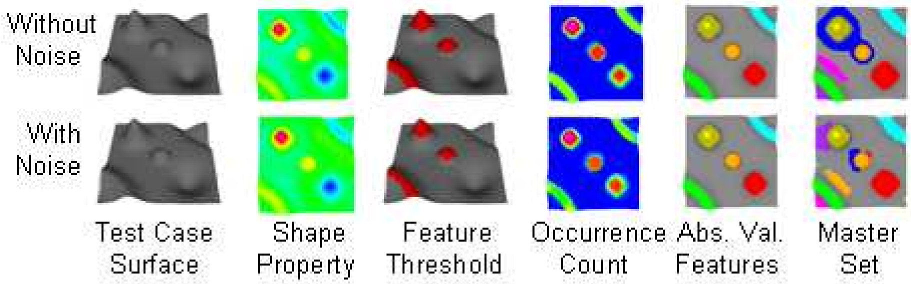

Fig. 2. Test case without and with Gaussian noise added. The function and final feature set are similar for the two cases, especially for the primary features.

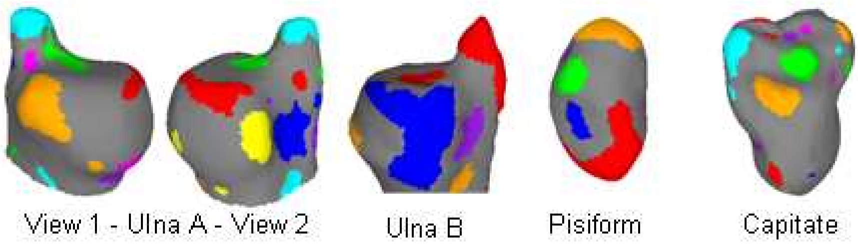

Fig. 3. Master Feature Sets for selected bone meshes. The Ulna is challenging due to the limited number of pronounced features and the significant difference between the scales of the features. Similar features were detected for Cases A and B even though the resolution of the meshes is very different. Reasonable features were also identified for the Pisiform and Capitate.

a simple greedy approach lets features grow, but without allowing neighboring features to merge. This ensures that all of the features do not get merged into a single feature, as might occur for a very large scale factor.

4 Results

Figure 2 shows the similar feature structures produced for a simple test surface with and without the addition of Gaussian noise.

The features for several bone meshes are shown in Figure 3. These bones have fairly subtle features. The feature layouts for Ulna A (View 2) and Ulna B are similar despite significant differences in mesh resolution and being from different subjects.

Although the face scans and bunny, presented in Figure 4, produced a number of very small features, the larger feature regions, such as the nose and eyes (face), and ears, feet, and tail (bunny), seem to be features that could be useful for shape matching. Also, features are ordered by strength so that the most significant features can be used first in operations such as shape matching, and the weaker features may not be needed.

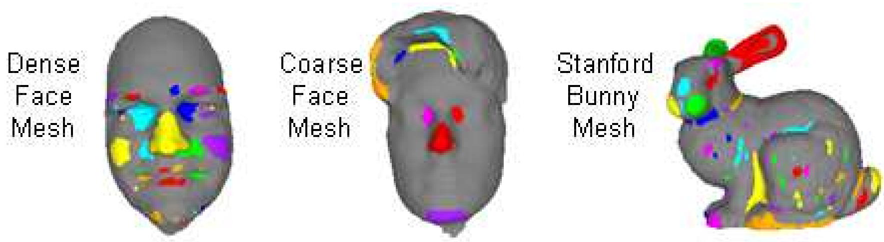

Fig. 4. Features detected for a dense face scan, coarse face scan, and the Stanford bunny. The larger features, which are also generally the strongest features, agree with the intuitive notion of features which may be useful for matching shapes.

5 Conclusions and Future Work

We have presented a two-step multi-scale feature detection approach that uses a local shape function based on the *Curvature Map*. It employs an efficient min-cut/max-flow graph cutting algorithm and greedy algorithm to merge feature sets. The method is robust with respect to noise, and consistently yields a reasonable set of features. Most importantly, there is no user interaction or parameter tuning required.

The method could benefit from alternate algorithms for merging feature sets. The greedy approach works fairly well, but may cause some over-segmentation, since it does not allow two features to coalesce into one, which might be desirable in some instances.

Acknowledgments This work was partially supported by NSF Grant 049856. The bone data was provided through NIH Grant AR44005, PI: J.J. Crisco, Brown Medical School /Rhode Island Hospital. The authors would also like to thank Vladimir Kolmogorov for the min-cut/max-flow code, Cyberware for the human head scans, and the Stanford Scanning Repository for the bunny data set.

References

1. Gatzke, T., Zelinka, S., Grimm, C., Garland, M.: Curvature maps for local shape comparison. In: Shape Modeling International. (2005) 244–256
2. Gatzke, T.D., Grimm, C.M.: Feature detection using curvature maps and the min-cut/max-flow graph cut algorithm. Technical Report WUCSE-2006-22, Washington University, St. Louis Missouri (2006)
3. van den Elsen, P., Maintz, J., Pol, E., Viergever, M.: Medical image matching - a review with classification. In: IEEE Engineering in Medicine and Biology. (1993) 26–39
4. Bloomenthal, J., Lim, C.: Skeletal methods of shape manipulation. In: International Conference on Shape Modeling and Applications. (1999) 44
5. Klein, P.N., Sebastian, T.B., Kimia, B.B.: Shape matching using edit-distance: an implementation. In: Symposium on Discrete Algorithms. (2001) 781–790
6. Hilaga, M., Schinagawa, Y., Kohmura, T., Kuni, T.L.: Topology matching for fully automatic similarity estimation of 3d shapes. In: Computer Graphics (SIGGRAPH). (2001) 203–212

7. Besl, P.J., McKay, N.D.: A method for registration of 3-d shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence **14**(2) (1992) 239–256
8. Rusinkiewicz, S., Levoy, M.: Efficient variants of the icp algorithm. In: Third International Conference on 3D Digital Imaging and Modeling (3DIM). (2001) 145–152
9. Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Shape distributions. In: ACM Transactions on Graphics, 21(4). (2002) 807–832
10. Johnson, A., Hebert, M.: Recognizing objects by matching oriented points. In: CVPR '97. (1997) 684–689
11. Ruiz-Correa, S., Shapiro, L.G., Meila, M.: A new signature-based method for efficient 3-d object recognition. In: CVPR (1). (2001) 769–776
12. Planitz, B.M., Maeder, A.J., Williams, J.A.: Intrinsic correspondence using statistical signature-based matching for 3d surfaces. In: Australian Pattern Recognition Society (APRS) Workshop on Digital Image Computing (WDIC). (2003)
13. G., M., Belongie, S., Malik, H.: Shape contexts enable efficient retrieval of similar shapes. In: CVPR 1. (2001) 723–730
14. Sun, Y., Paik, J.K., Koschan, A., Page, D.L., Abidi, M.A.: Point fingerprint: A new 3-d object representation scheme. IEEE Trans. Systems, Man, and Cybernetics, Part B **33**(4) (2003) 712–717
15. Yamany, S.M., Farag, A.A.: Surface signatures: An orientation independent free-form surface representation scheme for the purpose of objects registration and matching. IEEE Trans. Pattern Anal. Mach. Intell. **24**(8) (2002) 1105–1120
16. Takahashi, S., Ohta, N., Nakamura, H., Takeshima, Y., Fujishiro, I.: Modeling surperspective projection of landscapes for geographical guide-map generation. Computer Graphics Forum **21**(3) (2002) 259–268
17. Mortara, M., Patanè, G., Spagnuolo, M., Falcidieno, B., Rossignac, J.: Blowing bubbles for multi-scale analysis and decomposition of triangle meshes. Algorithmica **38**(1) (2003) 227–248
18. Maintz, J.B.A., van den Elsen, P.A., Viergever, M.A.: Registration of 3d medical images using simple morphological tools. In: IPMI. (1997) 204–217
19. McIvor, A.M., Penman, D.W., Waltenberg, P.T.: Simple surface segmentation. In: DICTA/IVCNZ97, Massey University, New Zealand (1997) 141–146
20. Vivodtzev, F., Linsen, L., Bonneau, G.P., Hamann, B., Joy, K.I., Olshausen, B.A.: Hierarchical isosurface segmentation based on discrete curvature. In: Proceedings of VisSym'03-Data Visualization 2003, New York, New York, ACM Press (2003) 249–258
21. Mangan, A.P., Whitaker, R.T.: Partitioning 3d surface meshes using watershed segmentation. In: IEEE Transactions on Visualization and Computer Graphics 5(4). (1999) 308–321
22. Page, D.L., Koschan, A.F., Abidi, M.A.: Perception-based 3d triangle mesh segmentation using fast marching watersheds. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03) **2** (2003) 27–32
23. Gal, R., Cohen-Or, D.: Salient geometric features for partial shape matching and similarity. j-TOG **25**(1) (2006) 130–150
24. Lee, C.H., Varshney, A., Jacobs, D.W.: Mesh saliency. Trans. Graph. **24**(3) (2005) 659–666
25. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Trans. Pattern Anal. Mach. Intell. **26**(9) (2004) 1124–1137
26. Boykov, Y., Jolly, M.P.: Interactive organ segmentation using graph cuts. In: MICCAI '00: Proceedings of the Third International Conference on Medical Image Computing and Computer-Assisted Intervention, London, UK, Springer-Verlag (2000) 276–286