Image Segmentation in Shape Synthesis, Shape Optimization, and Reverse Engineering

Milan Ćurković[1], Andrijana Ćurković[2],
Damir Vučina[1] and Domagoj Samardžić[1]

[1]Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split, Croatia
[2]Faculty of science Split, University of Split, Croatia

Abstract

Image segmentation and segmentation of geometry are one of the basic requirements for reverse engineering, shape synthesis, and shape optimization. In terms of shape optimization and shape synthesis where the original geometry should be faithfully replaced with some mathematical parametric model (NURBS, hierarchical NURBS, T-Spline, ...) segmentation of geometry may be done directly on 3D geometry and its corresponding parametric values in the 2D parametric domain. In our approach, we are focused on segmentation of 2D parametric domain as an image instead of 3D geometry. The reason for this lies in our dynamic hierarchical parametric model, which controls the results of various operators from image processing applied to the parametric domain.

Keywords

Image Segmentation, Shape Optimization, Shape Synthesis, Reverse Engineering.

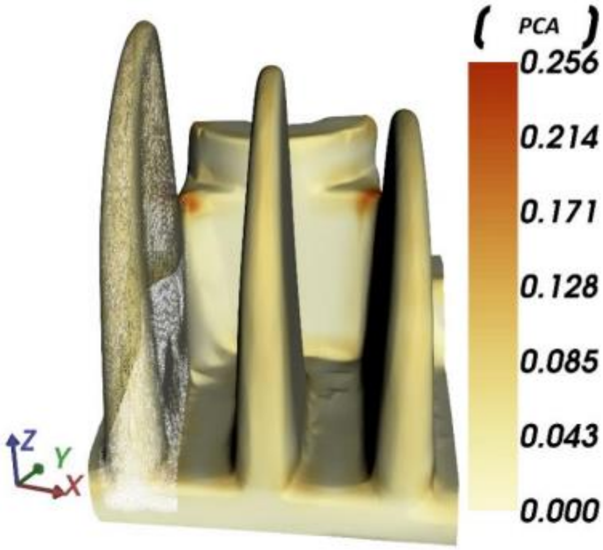
1. Introduction

Segmentation of geometry is one of the basic requirements for reverse engineering, shape optimization, and shape synthesis. There are successful algorithms for segmenting 3D geometry (3D point cloud / triangulated surface) [1,2,3], and successful algorithms for segmenting 2D images [4,5,6,7,8]. Algorithms for both dimensions, 2D and 3D, are based on the same ideas/approaches. The key difference is in simply topology in the case of 2D image (matrix representation). Those algorithms use information about geometric features: edges, peaks, gaps, and for that use PCA (principal component analysis), and Gaussian maps, etc. To determine the boundaries between regions, mainly the principle of 3D water shadow and its variations are used. In terms of shape optimization and shape synthesis, segmentation of geometry may be done directly on 3D geometry and on its corresponding parametric values in the projected 2D parametric domain. In our approach, we are more focused on the 2D parametric domain instead on 3D geometry. The reason for that lies in our dynamical hierarchical parametric model where new regions may appear and old ones may disappear. Moreover, some parts of the projected geometry cannot be assigned to any regions, and in this case, we use more layers of subparametrizations. The parametric model successfully deals with both situations: connected regions and more layers of sub-parametrizations, but due to faster convergence in optimizations of engineering samples and more simply CAD reversing, the clear boundaries between nature regions are more preferred.

David C. Wyld et al. (Eds): AI, AIMLNET, BIOS, BINLP, CSTY, MaVaS, SIGI - 2022
pp. 279-284, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.121824

2. PARAMETRIC DOMAIN

In the processes of shape optimization and shape synthesis we have changing geometry, changing topology, changing partitions, and the way of connection between partitions. As we mentioned, the partition creation may be done by segmentation of geometry on 3D triangulation shape. In the beginning, before the initial solution, the segmentation may be applied to the initial 3D geometry with the original triangulated mesh mainly obtained from a 3D scanner (Figure 1 a). After the initial solution in the optimization process, the segmentation can also be applied to the 3D geometry of the parametric model which is smooth geometry whose sharpness depends on how well the parametric model fits the referent geometry (initial geometry, geometric primitives, some mathematical functional, …). Regardless of that possibility, we do segmentation in the 2D parametric model (Figure 1 b). The benefits of this choice are the usage of fast algorithms from the area of image processing, the matrix topology of parametric values, the easy possibility to make connections between regions (parametric patches) and making more levels in the hierarchical parametric model. Of course, there are some drawbacks. If we do segmentation on the image obtained from parametric projection (Figure 2), there is the possibility that more triangles (all three vertices) have the same pixel positions (discrete x and y coordinates). This drawback is visible as wide white regions between partitions in figure (Figure 4). This problem can be solved by subpixel approaches which are part of future work.



a)

b)

Figure 1. The reference point in the optimization process. a) Initial geometry with included geometric features; b) projection of geometry of a) into the 2D parametric domain. From our point of view, classic segmentation algorithms applied to 3D geometry give more stable and enough sense results. But, in our case where we have the parametric model as a supervisor in making decisions, segmentation on 2D images gives solutions that are completely incorporated into our parametric model. The figure below (Figure. 2) presents the surface of the parametric projection domain (Figure 1 b). In this image, we apply the water shadow algorithm [8] after the iterative distance algorithm.

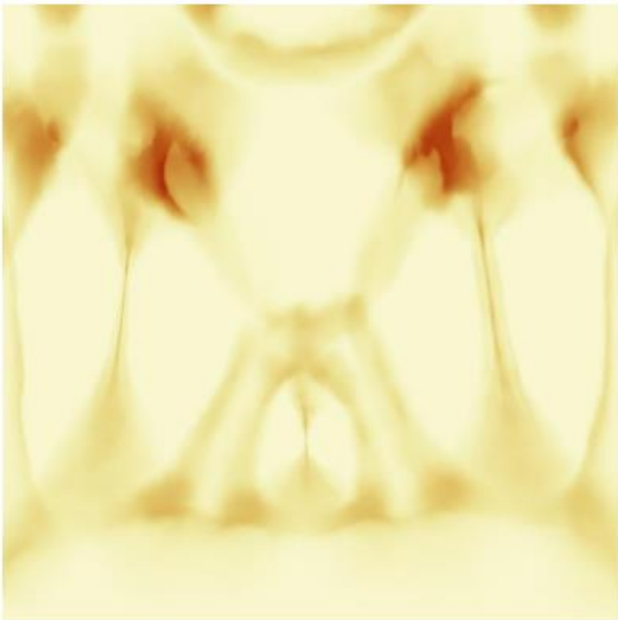
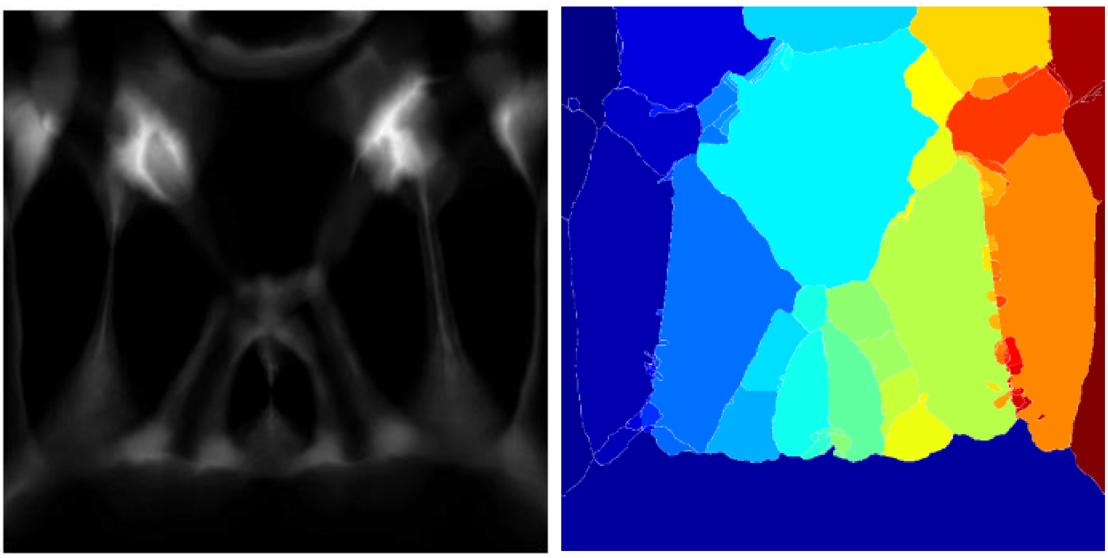


Figure 2. The surface of the parametric projection domain (Figure 1 b).

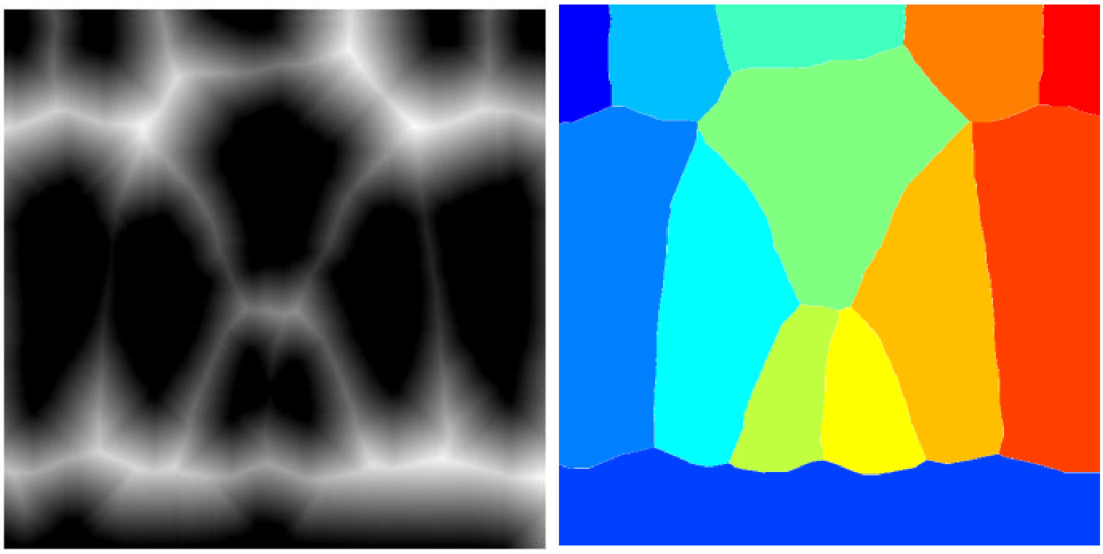
Despite the subpixel problem, we decided to show the capability of our approach on, in our opinion, not simply geometry. The key part of the algorithm is the usage of the iterative distance function as preparation for the water shadow algorithm. For simplicity, we have used a simple version of the distance function from Algorithm 1. The full versions can be found in [9,10,11]. As can be seen in Figure 3, we applied Algorithm 1 to the image as a projection of the geometry in the parametric 2D domain with two different values of the input parameter eps. As a result, in the first row in Figure 3, we get too many components that is the solution we want to avoid. The main goal is to get fewer clear (more convex look) components (second row in Figure 1). Of course, there is a possibility to overdo it with a small value of the eps parameter and in that case, we get too few components. The part of the future work is to find how to stop decreasing the parameter eps using the derivations of the distance algorithm. We concluded that under the control of the parametric model, i.e., control over what the components should look like and what conditions must be met, it is worth performing segmentation on 2D parameter values.

Algorithm 1. A proposed variation of the distance algorithm

```
Input:
-   Image I.
-   Real eps.
Output: result image R.
Begin
    Real eps2 = eps / √2
    for y = 0 to height
        for x = width – 1 to 0
            Real SW = eps + I(x - 1, y - 1)
            Real SE = eps + I (x + 1, y - 1)
            Real S = eps2 + I (x, y - 1)
            Real E= eps2 + I (x + 1, y)
            R (x, y) = min {I (x, y), SW, SE, S, E}
        end
    end
    for y = height to 0
        for x = 0 to width
            Real NW = eps + I (x - 1, y + 1)
            Real  NE = eps + I (x + 1, y + 1)
            Real  N = eps2 + I (x, y + 1)
            Real  W = eps2 + I (x - 1, y)
            R (x, y) = min {I (x, y), NW, NE, N, W}
        end
    end
    return R.
End
```



a) b)



c) d)

Figure 3. Segmentation of parametric domain of the parametric model. a) the results of the distance algorithm with parameter eps = 1e-2, b) the result of the water shadow algorithm applied on (a); c) the results of the distance algorithm with parameter eps = 1e-5, and (b); the result of water shadow algorithm applied on (c).

The figure above shows the final result of segmentation on the original 3D surface.

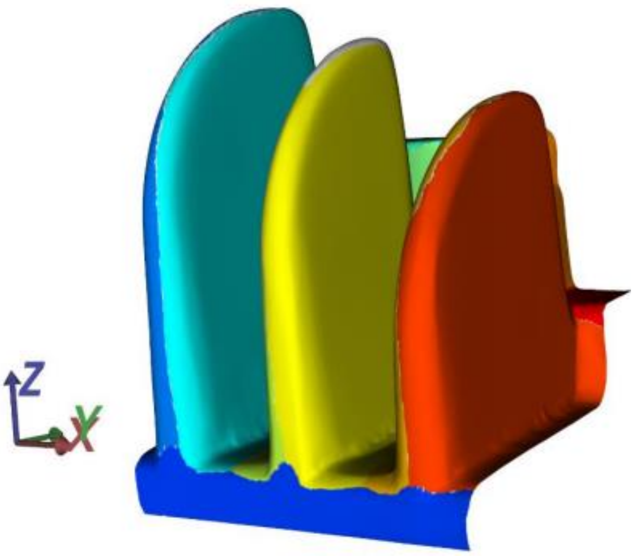


Figure 4. The segmentation result is shown on the initial 3D geometry.

3. CONCLUSIONS

Under parametric model supervision, it's worth performing the surface segmentation indirectly on its parametric domain as a 2D image. Without the control of the parametric model, i.e. the control of how the components should look like and what conditions must be satisfied (convexity), the segmentation presents many more challenges. Otherwise, the "referent" segmentation should be done directly on the 3D surface. If the segmentation result satisfies the parametric model, the shape optimization and synthesis processes can continue. Otherwise, a local extreme is created, and without the concept of the genetic algorithm, there is no point in continuing with the above processes.

ACKNOWLEDGEMENTS

This work was supported by the Croatian Science Foundation [grant number IP-2018-01-6774]

REFERENCES

[1] Daniel Mejia, Oscar Ruiz-Salguero, Jairo R. Sánchez, Jorge Posada, Aitor Moreno, Carlos A. Cadavid, Hybrid geometry / topology based mesh segmentation for reverse engineering, Computers & Graphics, Volume 73, 2018, pp. 47-58.
[2] Zhenyu Shu, Sipeng Yang, Haoyu Wu, Shiqing Xin, Chaoyi Pang, LadislavKavan and Ligang Liu, 3D Shape Segmentation Using Soft Density Peak Clustering and Semi-Supervised Learning, Computer-Aided Design, Vol. 145, 2022
[3] David George, XianghuaXie, Yukun Lai and Gary K.L. Tam, A Deep Learning Driven Active Framework for Segmentation of Large 3D Shape Collections, Computer-Aided Design, Vol. 144, 2022
[4] Shuai Luo, Yujie Li, Pengxiang Gao, Yichuan Wang and Seiichi Serikawa, Meta-seg: A survey of meta-learning for image segmentation, Pattern Recognition, Vol. 126, 2022
[5] Dong Wang and Xiao-Ping Wang, The iterative convolution–thresholding method (ICTM) for image segmentation, Pattern Recognition, Vol. 130, 2022
[6] JordãoBragantini, Alexandre X. Falcão and Laurent Najman, Rethinking interactive image segmentation: Feature space annotation, Pattern Recognition, Vol. 131, 2022
[7] Abdulateef, Salwa Khalid and Mohanad Dawood Salman. A Comprehensive Review of Image Segmentation Techniques. Iraqi Journal for Electrical and Electronic Engineering, 2021
[8] Wang, Bingshu& Chen, C., Local Water-Filling Algorithm for Shadow Detection and Removal of Document Images, 2020. Sensors. 20. 6929. 10.3390/s20236929.
[9] Maurer, Calvin, Rensheng Qi, and Vijay Raghavan, A Linear Time Algorithm for Computing Exact Euclidean Distance Transforms of Binary Images in Arbitrary Dimensions, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 2, February 2003, pp. 265-270.
[10] T. Schouten, E. van den Broek, Fast exact Euclidean distance (FEED) transformation, Proceedings of the 17th International Conference on Pattern Recognition, 2004., ICPR 2004, vol.3, IEEE pp. 594-597.
[11] LakshithaDantanarayana, Gamini Dissanayake, Ravindra Ranasinge, C-LOG: A Chamfer distance based algorithm for localisation in occupancy grid-maps, CAAI Transactions on Intelligence Technology, 2016., Volume 1, Issue 3