Implementing Non-Isometric Curvature Flows

15-300, Fall 2019

Alex Havrilla

November 8, 2019

1 Description

For my project I will be formulating non-isometric curvature flows for curves and surfaces and implementing them in C^{++}. I will be working with Professor Keenan Crane, a faculty member in the Computer Science Department concentrating in the application of Discrete Differential Geometry to a number of areas.

Currently most curvature shortening flows are isometries, ie. fix edge lengths. However this is not always ideal as it may lead to deformation and "unnatural shaping". Ideally we should have more control over how flow is performed, for example in curve drawing and edge detecting. Implementing non-isometric flows would take us a step in this direction, allowing us to produce "smoother" and more "natural" curves. DDG, Discrete Differential Geometry, appeals to me in particular because of the close interplay being the development of analytical theory and the resulting implementation of algorithms exploiting this theory.

This should be a fairly challenging problem. Since we aim to minimize both edge lengths and curvature, we will need to impose additional constraints to ensure well-posedness of the problem.

In the isometric setting to ensure the closedness of a curve we are flowing, we must periodically check conditions on the tangents and endpoints of the curve, and make updates accordingly. In particular it seems likely additional constraints will need to be formulated in the non-isometric setting in order to ensure closedness.

More information can be found at https://dahoas.github.io/DDGFlows/

2 Goals

1. 75%: Develop a robust(with respect to time step and discretization quality) curvature flow algorithm which seeks to globally minimize both angle and edge length.

2. 100%: Develop a robust curvature flow algorithm which seeks to minimize both angle and edge length and allows for fixing of subregion to perform flow(eg: pick curve between two points).

3. 125%: Develop a robust curvature flow algorithm which seeks to minimize both angle and edge length and allows for fixing of subregion to perform flow(eg: pick curve between two points) and allows user control over speed of flow, duration etc. More interactive.

3 Milestones

1st technical milestone: By the end of the semester I aim to have completed all recommended background reading and implemented contemporary algorithms for comparison: For example implementing the isometric conformal wilmore flow described by Crane, Pinkall, and Schroder et al. [1].

Bi-weekly Milestones: January 27th: Finish formulating constraints for edges.

February 10th: Finish formulating constraints for edges, curvature and other factors to begin implementing an algorithm.

February 24th: Complete rough prototype of algorithm.

March 16th: Improve robustness with respect to time step, discretization quality, other factors.

March 30th: Finish global algorithm and start implementing local control conditions, ie. picking subregions bounded between two points.

April 13th: Have rough prototype allowing for subregion control.

April 27th: Present finished algorithm with developed constraints and control.

4 Literature

Robust Fairing via Conformal Curvature Flow by Crane, Pinkert, and Schroder develops

Discrete Wilmore Flow by Bobenko and Schroder introduces

As a good introduction/reminder of how flows are commonl implemented and used in DDG, Keenan Crane has a nice homework as part of his course on DDG: http://brickisland.net/cs177fa12/?p=320. As I continue reading inevitably I'll come into contact with more papers, some of which may become useful and some of much may not.

5 Resources

Currently I plan to implement future algorithms in C++. However when doing any work related to geometry it is always nice to have visuals/physical models representing the application your algorithm to a surface. Should I find this useful I believe the Geometry Collective has access to printers allowing for this kind of modeling.

References

[1] Keenan Crane, Ulrich Pinkall, and Peter Schröder. 2013. Robust fairing via conformal curvature flow. ACM Trans. Graph. 32, 4, Article 61 (July 2013), 10 pages. DOI: https://doi.org/10.1145/2461912.2461986

[2] Alexander I. Bobenko and Peter Schröder. 2005. Discrete Willmore flow. In Proceedings of the third Eurographics symposium on Geometry processing (SGP '05). Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, , Article 101

[3] Keenan Crane. 15-458 Introduction to Discrete Differential Geometry. http://brickisland.net/cs177fa12/?p=320