Morphing Planar Triangulations and Polygons using Convex Representations

Vitaly Surazhsky*
vitus@cs.technion.ac.il

Craig Gotsman †
gotsman@cs.technion.ac.il

Introduction

Morphing, also known as metamorphosis, is the gradual transformation of one shape into another. Morphing has wide practical use in areas such as computer graphics, animation and modeling. Until now, to achieve more spectacular, impressive and accurate results, the morphing process requires a lot of the work to be done manually. A major research challenge is to develop techniques that will automate this process as much as possible.

The morphing problem has been investigated in many contexts, e.g., morphing of two-dimensional images [3, 7], polygons and polylines [11, 10, 12, 8, 4], freeform curves [9] and even voxel-based volumetric representations [5]. The morphing process always consists of solving two main problems. The first one is to find a correspondence between elements (features) of the two shape representations. The second problem is to find trajectories that corresponding elements traverse during the morphing process. Regrettably, a formal definition of a successful correspondence does not exist, as well as a definition of a successful solution of the trajectory problem.

The naive approach to solve the trajectory problem is to choose the trajectories to be straight lines, where every feature of the first shape traverses with a constant velocity towards the corresponding feature of the second shape. Unfortunately, this simple approach can lead to some undesirable results. The intermediate shapes can vanish, i.e. degenerate into a single point, or contain self-intersections even though the two initial shapes are simple, namely, self-intersection free. Even if the linear morph is free of self-intersections and degenerate regions, its intermediate shapes may have areas or distances between the shape elements far from those of the given shapes, resulting in a 'misbehaved' looking morph. Most of the research on solving the trajectory problem for morphing concentrates on the elimination of self-intersections and on the preservation of geometrical properties of the intermediate shapes. Many existing methods achieve good results for many inputs, however, none guarantees that the resulting morph is always self-intersection free.

Morphing Planar Triangulations

Two triangulations with a predefined correspondence are said to be *compatible* (*isomorphic*), if they are topologically equivalent. In [6], an innovative approach for morphing compatible planar

*Computer Science Department, Technion City, Haifa 32000, Israel
†Computer Science Department, Technion City, Haifa 32000, Israel

triangulations was introduced. The approach is based on convex representation of triangulations. It is shown that the method, applied to compatible triangulations with identical convex boundaries, always yields a continuous sequence of valid compatible triangulations, namely, the morph is always self-intersection free.

In this work we investigate properties and capabilities of the basic approach. Having many degrees of freedom, the approach may be tuned to obtain morphs with various prescribed characteristics, hence we show how to generate a smooth morph (vertex trajectories are C^1 continuous) through a predefined intermediate triangulation using this technique. A drawback of this method, however, is that the resulting morph may have non-linear vertex trajectories even when linear trajectories are legal and have a natural look. To rectify this, we present a linear-reducible scheme extending the basic technique to one which generates the linear morph if it is valid, or a valid morph with trajectories as close to linear as possible otherwise. Another extension of the basic approach allows control of the geometric features of the triangulation in a local manner, yet still preserving global validity. We demonstrate how to obtain a morph with a natural evolution of triangle areas using this extension. See Figure 1 for an example of a morph between two triangulations.

These techniques always yield a valid morph and are a robust method for morphing images represented by textured triangulations. Regrettably, they are appropriate only for planar triangulations with identical boundaries. On the other hand, all existing techniques for morphing polygons and polylines do not guarantee a valid self-intersection morph. Therefore, the next challenge is to apply our robust techniques to polygons and polylines.

Morphing Planar Polygons

A sequence of contiguous edges within a triangulation, namely, a path in the graph of triangulation, may be viewed as a piecewise linear curve. A simple closed path forms a polygon. Hence, a morph of two corresponding polylines or polygons embedded into two compatible triangulations may be generated by morphing these triangulations. Clearly, any valid morph of these two triangulations will define a valid morph of the embedded polylines. However, this requires us to find embeddings that allow this.

The problem of finding embeddings of polygons within triangulations may be formulated as follows: Given two polygons with a correspondence between their vertices, find two compatible triangulations with identical boundaries such that the two given polygons are embedded into the corresponding triangulations and these embeddings correspond. See the first and the last triangulations on Figure 1 for an example of how two polygons may be embedded into compatible triangulations. We have tried to solve the problem in such a way that the solution might be generalized for embeddings of piecewise linear 3D shapes within compatible tetrahedralizations as well. In [1], two algorithms that construct corresponding compatible triangulations of simple polygons with at most $O(n^2)$ Steiner vertices are presented. A subsequent work [2] constructs compatible triangulations of polygons with holes. Both these works show the feasibility of the solution to the embedding problem, but may cause numerical problems and very unfair triangulations as a result of producing skinny (close to degenerate) triangles.

The method we present is based on one of the algorithms described in [1] called a "universal triangulation". This "universal triangulation" has a form of a "spiderweb" of $n-2$ concentric layers of n-vertex polygons and one central vertex, where all corresponding vertices on adjacent layers are connected by radial edges. First, we add a n-vertex convex polygon around each of the

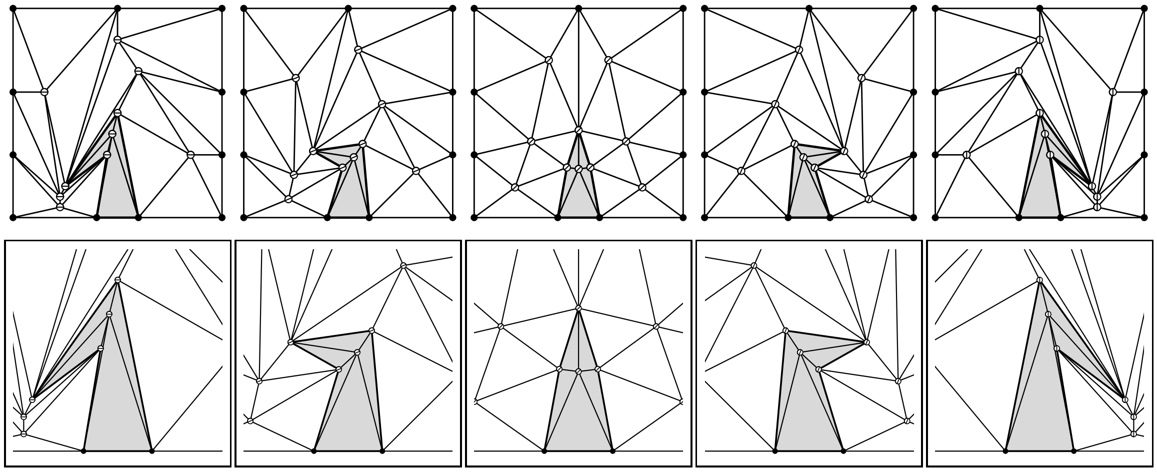

Figure 1: An example of the morphing sequences between two polygons embedded into two compatible triangulations. The lower row is a zoom in on the polygon. Note that the linear morph between these polygons has self-intersection.

two input polygons, which is necessary in order to apply the convex representation approach. The next step is to triangulate the interiors and exteriors (the space between each polygon and the added outer polygon) of the polygons in a compatible manner. We construct a striped version of the "spiderweb" triangulation that may have incomplete concentric layers and, thus, different radial path lengths from the polygon vertices to the central vertex. This reduced "spiderweb" is the minimal triangulation that allows compatibility of the two input polygons and has the "spiderweb" connectivity. Since the reduced triangulation has less Steiner vertices than the regular "spiderweb", it has less skinny and more uniform triangles. The exteriors of the polygons are filled in a similar manner by a reduced "spiderweb"-form triangulation, which has no central vertex due to the topology of the exteriors—polygons with a single hole.

References

[1] B. Aronov, R. Seidel, and D. L. Souvaine, *On compatible triangulations of simple polygons*, Computational Geometry: Theory and Applications, 3 (1993), pp. 27–35.

[2] M. Babikov, D. L. Souvaine, and R. Wenger, *Constructing piecewise linear homeomorphisms of polygons with holes*, Proceedings of 9th Canadian Conference on Computational Geometry, (1997).

[3] T. Beier and S. Neely, *Feature-based image metamorphosis*, Computer Graphics (SIGGRAPH '92), 26 (1992), pp. 35–42.

[4] E. Carmel and D. Cohen-Or, *Warp-guided object-space morphing*, The Visual Computer, 13 (1997), pp. 465–478.

[5] D. Cohen-Or, D. Levin, and A. Solomovici, *Three-dimensional distance field metamorphosis*, ACM Transaction on Graphics, 17 (1998).

[6] M. S. Floater and C. Gotsman, *How to morph tilings injectively*, (1999). To appear in Journal of Computational and Applied Mathematics.

[7] K. Fujimura and M. Makarov, *Foldover-free image warping*, Graphical

Models and Image Processing, 60 (1998), pp. 100–111.

[8] E. Goldstein and C. Gotsman, *Polygon morphing using a multiresolution representation*, Proceeding of Graphics Interface, (1995).

[9] T. Samoilov and G. Elber, *Self-intersection elimination in metamorphosis of two-dimensional curves*, The Visual Computer, 14 (1998), pp. 415–428.

[10] T. W. Sederberg, P. Gao, G. Wang, and H. Mu, *2D shape blending: an intrinsic solution to the vertex path problem*, Computer Graphics (SIGGRAPH '93), 27 (1993), pp. 15–18.

[11] T. W. Sederberg and E. Greenwood, *A physically based approach to 2D shape blending*, Computer Graphics (SIGGRAPH '92), 26 (1992), pp. 25–34.

[12] M. Shapira and A. Rappoport, *Shape blending using the star-skeleton representation*, IEEE Trans. on Computer Graphics and Application, 15 (1995), pp. 44–51.

[13] V. Surazhsky, *Morphing planar triangulations*, Master's thesis, Technion—Israel Institute of Technology, 1999.

[14] W. T. Tutte, *How to draw a graph*, Proc. London Math. Soc., 13 (1963), pp. 743–768.