

## Multiresolution morphing for planar curves

S. Hahmann, G.-P. Bonneau, B. Caramiaux, and M. Cornillac, Grenoble

Received January 21, 2006; Revised April 10, 2006

Published online: March 7, 2007

© Springer-Verlag 2007

### Abstract

We present a multiresolution morphing algorithm using “as-rigid-as-possible” shape interpolation combined with an angle-length based multiresolution decomposition of simple 2D piecewise curves. This novel multiresolution representation is defined intrinsically and has the advantage that the details’ orientation follows any deformation naturally. The multiresolution morphing algorithm consists of transforming separately the coarse and detail coefficients of the multiresolution decomposition. Thus all LoD (level of detail) applications like LoD display, compression, LoD editing etc. can be applied directly to all morphs without any extra computation. Furthermore, the algorithm can robustly morph between very large size polygons with many local details as illustrated in numerous figures. The intermediate morphs behave natural and least-distorting due to the particular intrinsic multiresolution representation.

*AMS Subject Classifications:* 68U07, 68U05, 65D05, 65D08, 65D17.

*Keywords:* Geometric modeling, curves, multiresolution analysis, morphing, interpolation vertex correspondence.

### 1. Introduction

In computer graphics, morphing (or metamorphosis) is known as the smooth and progressive transformation of one shape into another. The shape can be an image or a planar curve in 2-D space, or it can be a surface or a volume in 3-D space. The problem is to create an aesthetic and intuitive transition between two shapes. The intermediate shapes should preserve the appearance and the properties of the input shapes. A trivial linear interpolation is often not appropriate, since the intermediate shapes tend to vary a lot in their volume or they loose the proportions of their shape features. Another negative effect is that the geometric details can disappear and re-appear later during the transition. The morphing process consists of solving two problems: the *vertex correspondence problem* (finding the correspondence between the geometric features of the source and target object) and the *vertex path problem* (finding the trajectory two corresponding elements follow during the morphing). Both problems still attract much attention in research, since no formal definition of a successful solution exists. In this work we assume that the correspondence is given and only the vertex path problem is to be solved for 2-D polygonal shapes.

Good results are generally achieved not by interpolating the positions of the boundary representation but by interpolating elements of alternative representations. In the case of 2-D polygonal shapes, Sederberg et al. [16] represent polygons by at set of

lengths and angles. Shapira and Rappoport [18] use a star-skeleton representation. Goldstein and Gotsman [10] use a multiresolution representation based on curve evolution. Morphing is also performed on elements describing the interior of a 2-D shape like triangles. Herein Floater and Gotsman [9] have used barycentric coordinates to morph compatible triangulations and Surazhsky and Gotsman [20] furthermore interpolate intrinsic components of these coordinates. Alexa et al. [3] morph compatible triangulations by locally least-distorting maps. Let us also mention one multiresolution mesh morphing technique by Lee et al. [12]. For the 3-D mesh morphing techniques a survey is given by Alexa [2]. The key to a successful method thus seems to be the use of a representation based on intrinsic properties of the object geometry such that interpolation of its elements achieves automatically pleasing morphs.

The morphing method we introduce in the present paper is based on a new multiresolution representation. When modeling complex shapes with many geometric details which are difficult to manipulate efficiently, a multiresolution analysis provides a powerful tool for efficiently representing and deforming objects at multiple levels of resolution. Herein, a complex shape is decomposed into a “coarser” low resolution part, together with a collection of detail coefficients, necessary to recover the original shape. Deformation of complex objects with a lot of detail can be tricky and expensive to compute. In a multiresolution setting however complex objects can be edited at a chosen scale with mainly two effects. First, modifying some low-resolution control points and adding back the details modifies the overall shape of the object. Second, modifying a set of fine detail coefficients modifies the character of the object without affecting its overall shape.

Multiresolution morphing thus decomposes the source and the target shape into a coarse approximation and a set of detail coefficients, it computes separately the sequence of coarse intermediate shapes and details, and then reconstructs all intermediate shapes. The choice of the multiresolution representation is crucial for the quality of the resulting shapes. For example, a wavelet based multiresolution analysis would not preserve the orientation of the details during deformation. In fact, the details here are encoded in a global coordinate system. A multiresolution representation that encodes the details using local frames similar to [8] solves this problem, but is not appropriate for morphing. In fact, while the coarse polygon can make profit from Sederberg’s intrinsic morphing, the detail coefficients cannot. Thus the morph of the details would suffer from the same lack of realism as linear vertex morphs do.

In the present paper, we introduce a curvature based multiresolution representation for 2-D polygonal curves. All detail coefficients will be represented intrinsically by lengths and angles. We show that similar to local frames, the multiresolution representation preserves the orientation of the details during deformation. Furthermore we investigate its use in morphing by combining it with the “as-rigid-as-possible” morphing technique of Alexa et al. [3] which has been proven to provide good quality morphs. We therefore present an as-rigid-as-possible intrinsic multiresolution morphing algorithm.

## 2. Angle-length based multiresolution analysis

Multiresolution (MR) analysis is a process of splitting a vector of coefficients  $c^n$  into a set of a low-resolution part  $c^{n-1}$  and a detail part  $d^{n-1}$ , which can be applied recursively to the new vector  $c^{n-1}$ . Thus, the original vector of coefficients can be decomposed into a hierarchy of lower-resolution vectors  $c^{n-1}, \dots, c^0$  and detail coefficients  $d^{n-1}, \dots, d^0$ . This process is called *analysis or decomposition*. The original vector of coefficients can be recovered from any sequence  $c^j, d^j, d^{j+1}, \dots, d^n$ ,  $j = 0, \dots, n - 1$ . This process is called *synthesis or reconstruction*. More details and mathematical fundamentals on multiresolution analysis can be found in the original work of Mallat [14]. An overview of applications in computer graphics is given in [19].

Most existing MR representations for curves in CAGD or computer graphics are based on B-splines [8], [6], [7], [4], [11] and are designed for particular applications. In the case of deformation, and morphing belongs to this application, the orientation of the details is an important issue. As we have explained in the introduction, deformations of wavelet-based MR representations generally suffer from the lack of realism, since the orientation of the details keeps fixed during deformation.

Therefore we introduce here another MR analysis for piecewise linear planar curves (2-D polygons), which has the following two important properties regarding its use for morphing: the orientation of details follows the deformation naturally and the MR representation is defined intrinsically.

From differential geometry it is known that the curvature function of a parametric curve determines the curve uniquely up to an isometry [5]. The curvature is an intrinsic property of a curve. In the case of piecewise linear planar curves, angles and lengths define a curve intrinsically up to an isometry [1], [16]. We will use these intrinsic quantities to define the coarse coefficients and details of a MR analysis now.

Let denote the vertices of a polygon to be MR-analyzed by  $P_i = (x_i, y_i)$ ,  $i = 0, \dots, N - 1$ . The initial polygon needs to be transformed from  $(x, y)$ -coordinates into so-called  $(\theta, l)$ -coordinates, where  $\theta_i = \angle(\overline{P_{i-1}P_i}, \overline{P_iP_{i+1}})$  is the counterclockwise angle of the two consecutive polygon segments at  $P_i$  and  $l_i = \|\overline{P_iP_{i+1}}\|$ ,  $i = 0, \dots, N - 2$ , see Fig. 1. The  $(x, y)$ -coordinates of the control points  $P_i$  can be recovered directly using for example  $P_0$  as anchor point and  $\overline{P_0P_1}$  as anchor line (determining the translation and rotation).

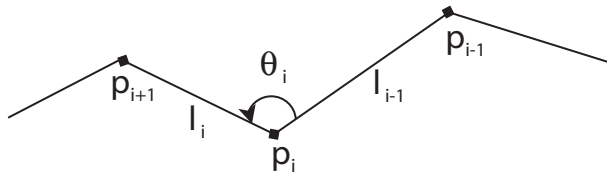


Fig. 1. From  $(x, y)$ -coordinates to  $(\theta, l)$ -coordinates ( $l_i, \theta_i$  function of  $P_{i-1}, P_i, P_{i+1}$ )

The *angle-length MR representation* we introduce now, exclusively works with  $(\theta, l)$ -coordinates. Similar to lazy wavelets [21] we choose to coarsen the polygons by subsampling half of the control points at each analysis step.

### 2.1. Analysis

Analysis (decomposition) is the recursive procedure of splitting the vector of coefficients of a polygon  $(\theta^{n+1}, l^{n+1})$  into a vector of coarse coefficients of a lower-resolution polygon  $(\theta^n, l^n)$  and a vector of detail coefficients. Let the upper index  $n$  denote the level of resolution. Both, coarse and detail coefficients of level  $n$  must be computed directly from  $(\theta^{n+1}, l^{n+1})$  and vice-versa.

Using basic trigonometric rules for triangles, we get the following analysis formulas, illustrated in Fig. 2:

#### Coarse Coefficients

$$\theta_i^n = \theta_{2i}^{n+1} + \theta_{2i-1}^{n+1} + \alpha_i^n - \alpha_{i-1}^n - \pi, \quad \text{for } i = 0, \dots, 2^n - 1, \tag{1}$$

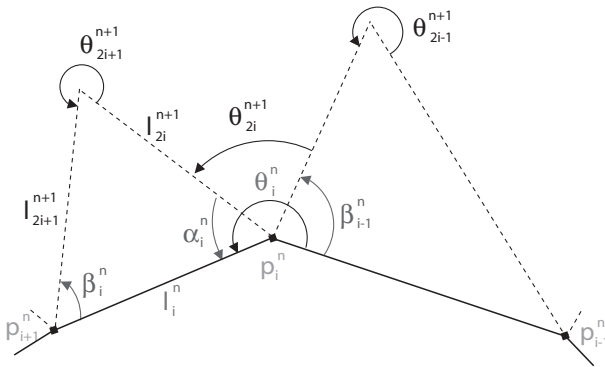
$$l_i^n = \sqrt{\left(l_{2i+1}^{n+1}\right)^2 + \left(l_{2i}^{n+1}\right)^2 - 2 \cdot l_{2i+1}^{n+1} \cdot l_{2i}^{n+1} \cos\left(\theta_{2i+1}^{n+1}\right)}.$$

#### Detail Coefficients

$$\alpha_i^n = \arccos\left(\frac{l_{2i}^{n+1} - l_{2i+1}^{n+1} \cos\left(\theta_{2i+1}^{n+1}\right)}{l_i^n}\right), \tag{2}$$

$$\beta_i^n = \theta_{2i+1}^{n+1} - \alpha_i^n - \pi, \quad \text{for } i = 0, \dots, 2^n - 1.$$

From a polygon with  $2^{n+1}$  segments, one analysis step creates a polygon with  $2^n$  segments and  $2^n$  detail coefficients, which are represented by two-dimensional vectors of the form:



**Fig. 2.** The analysis. The dotted polygon belongs to resolution level  $n + 1$ , the fat polygon belongs to level  $n$

$$\begin{aligned}
 (\theta^{n+1}, \mathbf{l}^{n+1}) &\rightarrow (\theta^n, \mathbf{l}^n) \\
 &\searrow \\
 &(\alpha^n, \beta^n),
 \end{aligned}$$

where  $\theta^n = (\theta_0^n, \dots, \theta_{2^n-1}^n)$  and analogous for  $\mathbf{l}^n, \alpha^n, \beta^n$ .

### 2.2. Synthesis

The analysis provides a coarse polygon  $(\theta^0, \mathbf{l}^0)$  (simple geometric shape) and a set of detail coefficients  $(\alpha^0, \beta^0), \dots, (\alpha^n, \beta^n)$ . The synthesis will reconstruct the original polygon. We are still working with the  $(\theta, l)$ -coordinates. Therefore, the algorithm consists in calculating the lengths  $l_{2i}^{n+1}, l_{2i+1}^{n+1}$  and the angles  $\theta_{2i}^{n+1}, \theta_{2i+1}^{n+1}$  of level  $n + 1$  from the coarse coefficients and the details of level  $n$ . The process corresponds to one synthesis step. We can then iterate the algorithm on the reconstructed shape with details.

#### Reconstruction of Lengths

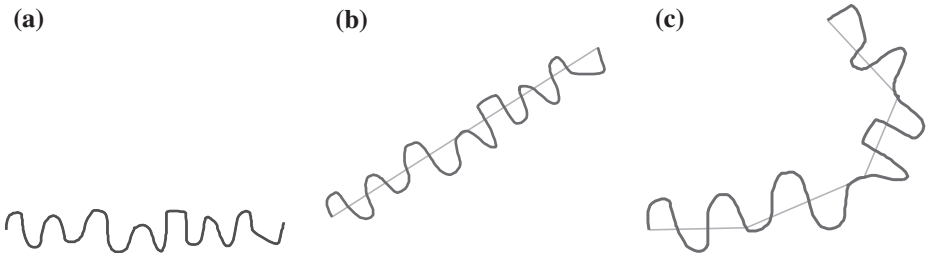
$$\begin{aligned}
 l_{2i}^{n+1} &= l_i^n \cdot \frac{\sin(\beta_i)}{\sin(\alpha_i + \beta_i)}, \\
 l_{2i+1}^{n+1} &= l_i^n \cdot \frac{\sin(\alpha_i)}{\sin(\alpha_i + \beta_i)}.
 \end{aligned} \tag{3}$$

#### Reconstruction of Angles

$$\begin{aligned}
 \theta_{2i}^{n+1} &= \theta_i^n - \alpha_i^n - \beta_{i-1}^n, \\
 \theta_{2i+1}^{n+1} &= \pi + \alpha_i^n + \beta_i^n.
 \end{aligned} \tag{4}$$

### 2.3. Special cases

Three special cases occur which need an extra treatment. They correspond to the cases where the detail coefficients  $(\alpha_i, \beta_i)$  admit the values  $(0, 0), (0, \pi)$  and  $(\pi, 0)$ .



**Fig. 3a.** Initial curve with 256 control points. **(b)** Deformation of coarse polygon at level 6. **(c)** Deformation of coarse polygon at level 8. The orientation of the details follows naturally the orientation of the coarse polygon during deformation

Geometrically these cases correspond to degenerated triangles, i.e., where  $p_{2i}^{n+1}$ ,  $p_{2i+1}^{n+1}$  and  $p_{2i+2}^{n+1}$  are collinear. When storing these detail values as usual during analysis, one would not be able to recover correctly the lengths  $l_{2i}^{n+1}$ ,  $l_{2i+1}^{n+1}$  during synthesis since these values reported in the previous formulas (3) involve divisions by 0. Thus  $l_{2i}^{n+1} = \infty$  and  $l_{2i+1}^{n+1} = \infty$ . When these cases occur we proceed as follows:  $\beta_i$  will be set to a specific value corresponding to one of the three cases. This value will allow us to know if a special treatment is needed during the synthesis, and  $\alpha_i$  will be set to the ratio of lengths, so that the reconstruction can be performed without the numerical aberrations mentioned above.

$$- (\alpha_i, \beta_i) = (0, 0)$$

the analysis gives us the ratio  $r = \frac{l_{2i}^{n+1}}{l_{2i+1}^{n+1} + l_{2i}^{n+1}}$  and  $l_i^n = l_{2i+1}^{n+1} + l_{2i}^{n+1}$

the synthesis gives:  $\implies l_{2i}^{n+1} = l_i^n \times r$  and  $l_{2i+1}^{n+1} = l_i^n \cdot (1 - r)$

$$- (\alpha_i, \beta_i) = (0, \pi)$$

the analysis gives us the ratio  $r = \frac{l_{2i}^{n+1}}{l_{2i}^{n+1} - l_{2i+1}^{n+1}}$  and  $l_i^n = l_{2i}^{n+1} - l_{2i+1}^{n+1}$

the synthesis gives:  $\implies l_{2i}^{n+1} = l_i^n \times r$  and  $l_{2i+1}^{n+1} = l_i^n \cdot (r - 1)$

$$- (\alpha_i, \beta_i) = (\pi, 0)$$

the analysis gives us the ratio  $r = \frac{l_{2i}^{n+1}}{l_{2i+1}^{n+1} - l_{2i}^{n+1}}$  and  $l_i^n = l_{2i+1}^{n+1} - l_{2i}^{n+1}$

the synthesis gives:  $\implies l_{2i}^{n+1} = l_i^n \times r$  and  $l_{2i+1}^{n+1} = l_i^n \cdot (1 + r)$ .

### 3. Multiresolution morphing

Given two polygons with the same number of corresponding control points  $P_S$  and  $P_T$ , called source and target polygons. We aim to construct in-between polygons  $P_t$  which gradually change  $P_S$  into  $P_T$  for  $t \in [0, 1]$ , where  $P_S = P_0$  and  $P_T = P_1$ . There exist many possible interpolants for  $P_t$  and it is particularly difficult to characterize mathematically “good” morphing algorithms, since the judgment depends on both, the application area and the human perception of geometry.

The most trivial method consists in interpolating along a straight line between corresponding vertex positions. This vertex interpolation is known to change inconsistently many geometric properties, such as lengths, angles and area during the morph. Sederberg et al. [16] presented an improved interpolation method. Instead of interpolating between vertex positions, the algorithm interpolates between entities of an intrinsic polygon definition: angles and edge lengths. Such a polygon representation is rigid-motion invariant. The advantage is that the morphing of two characters behave naturally. It turns out that the proportions of all geometric features are preserved.

The work of Lipman et al. [13] can be seen as a kind of generalization of Sederberg’s 2-D interpolation technique to 3-D mesh interpolation. They introduce a

rigid-motion invariant representation of surfaces based on intrinsic discrete fundamental forms, and propose to interpolate the coefficients of these forms.

The approach of interpolating between intrinsic entities has further been employed by Surazshky and Gotsman [20]. They first compute compatible triangulations of the source and target polygons and then interpolate the mean value barycentric coordinates of the triangles by using a lengths and angles description. The advantage of morphing compatible triangulations is that self-intersecting morphs can generally be avoided. However, the computation of compatible triangulations and other representations of the interior of polygons [18] are generally difficult and very time consuming.

The algorithm we present in this section belongs to the class of algorithms based on intrinsic boundary representations. The input polygons are represented by the intrinsic multiresolution representation we introduced in Sect. 2. The multiresolution nature of this representation has the advantage to generate the in-between polygons directly at many levels of detail without computing a MR-analysis for them. The following features characterize our intrinsic MR algorithm and make it particularly appropriate for morphing:

- The orientation of details follows the deformation naturally.
- The MR representation is defined intrinsically.
- The intermediate morphs interpolate the source and target curves “as-rigid-as-possible” similar to [3].
- The intermediate morphs can be visualized at any level of resolution due to the MR morphing which applies the interpolation separately to the coarse and detail coefficients.
- In the case of closed curves, the non-closure problem of Sederberg’s method [16] does not occur.

### 3.1. Algorithm

Once angle-length MR representations of  $P_S$  and  $P_T$  have been computed one disposes of two sets of coarse coefficients  $(\theta_S^0, \mathbf{l}_S^0)$ ,  $(\theta_T^0, \mathbf{l}_T^0)$  and of two sets of detail coefficients  $(\alpha_S^0, \beta_S^0), \dots, (\alpha_S^n, \beta_S^n)$ , and  $(\alpha_T^0, \beta_T^0), \dots, (\alpha_T^n, \beta_T^n)$ . In principle, the in-between morphs are now generated by interpolating the coefficients of this intrinsic representation. But special attention has to be paid to the coefficients of the coarse polygon. When interpolating the  $(\theta, l)$ -coordinates of the coarse polygon, this would mean that we apply Serderberg’s algorithm [16] to the coarse polygon. But in this case we would inheritate the problem of non-closing intermediate polygons in cases where  $P_S$  and  $P_T$  are closed. And thus some extraneous optimization would be necessary. Instead, we propose a solution where this problem does not occur.

*Interpolation of coarse coefficients “as-rigid-as-possible”:* In the case of closed polygons with  $3 \cdot 2^n$  vertices, the coarse polygons after  $n$  analysis steps  $P_S^0$  and  $P_T^0$  consist of a triangle. In this case we interpolate between these two triangles with Alexa’s “as-rigid-as-possible” interpolation [3]. The unique affine transformation  $A$  between  $P_S^0$

and  $P_T^0$  is decomposed into a rotation matrix  $R_\gamma$  and a symmetric matrix  $S$  using SVD:  $A = R_\gamma S$ . The intermediate morphs of both triangles are then obtained by interpolating the components of this decomposition:  $A_\gamma(t) = R_{t\gamma}((1-t)I + tS)$ . This interpolation between two triangles is said to be “as-rigid-as-possible” since the angles vary close to linear path. In any other case, we can resample the original curve in order to have  $D \cdot 2^n$  points, where  $D$  is the small number of coarse control points. A coarse polygon with only a few vertices can easily be triangulated and interpolated by Alexa’s locally least distortion interpolation method.

*Interpolation of detail coefficients:* The detail coefficients intrinsically represent triangles which are added to the coarse polygon in order to bring it to a higher resolution level. Intermediate detail coefficients of level  $j$  are obtained by interpolating

$$\begin{aligned} \alpha_t^j &= (1-t)\alpha_S^j + t\alpha_T^j, \\ \beta_t^j &= (1-t)\beta_S^j + t\beta_T^j, \quad \text{for } 0 \leq j \leq n \quad \text{and } t \in [0, 1]. \end{aligned} \quad (5)$$

The intermediate shapes are the “as-rigid-as-possible” deformations of the source and target shape. In fact, the reconstruction of all morphs works through adding triangles to the lower-resolution polygons at each resolution level. Furthermore, these triangles are obtained by linearly interpolating the angles, see Eq. 5. Our method can thus be regarded as an “as-rigid-as-possible” morphing of multiresolution curves, following Alexa’s definition.

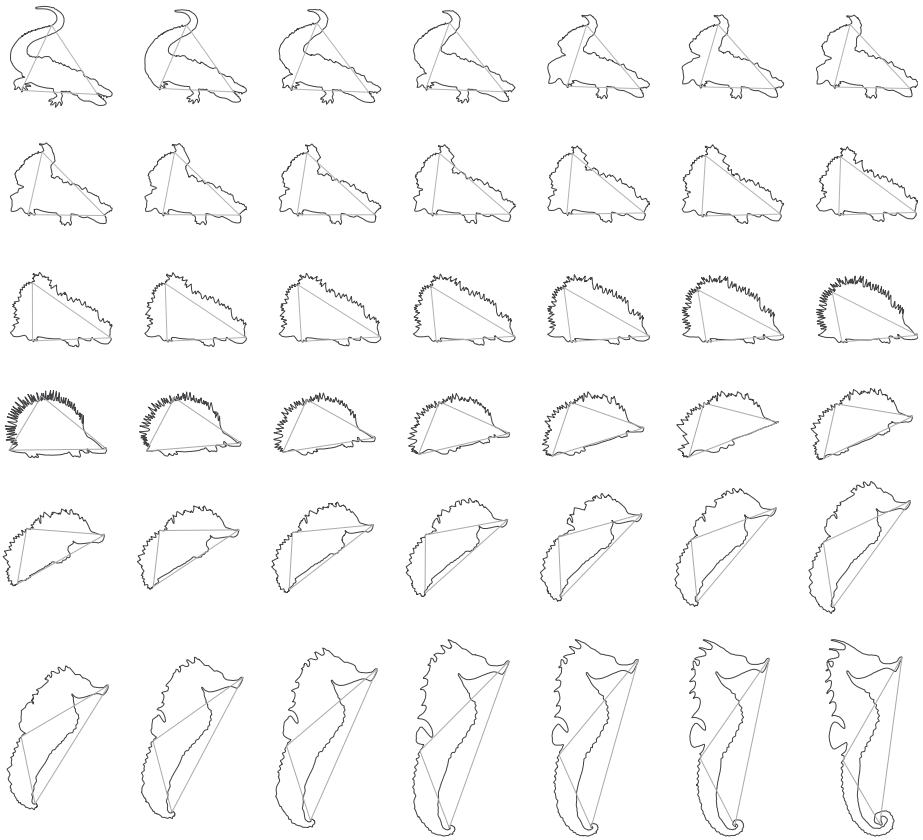
Another advantage of our method is, that we have an entire MR representation of all intermediate curves and can visualize them at different resolution levels.

### 3.2. Results and discussion

In contrast to most existing morphing algorithms, our method is designed for very large size polygons with a lot of local details. However, it can handle all other simple and smooth polygons as well. Therefore we only selected examples with a lot of fine details, which are difficult to morph. Most of them are generated by extracting a contour of an image.

#### *Vertex correspondence*

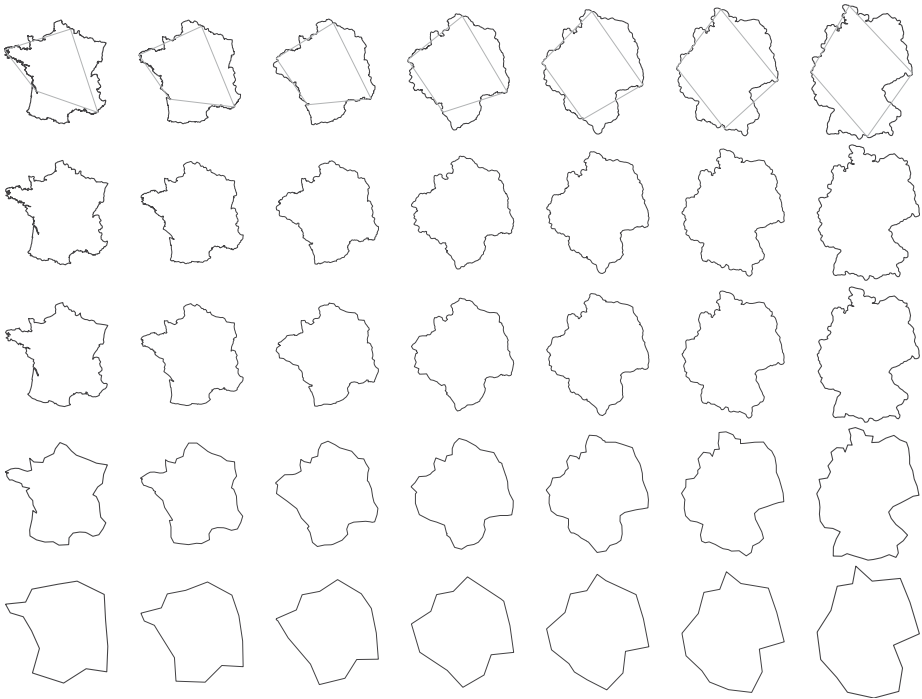
It is well known that the correspondence of several vertices can significantly increase the quality of the morphing [17]. Most morphing algorithms thus enforce manually the correspondence of a few selected points [3], [16], [10] or simply suppose that the problem is solved [20], [18]. In [15], the vertex correspondance problem is solved using wavelets. For the correspondence of vertices, we define manually a first starting point for the source and target polygon. All remaining vertices correspond then automatically by their indices. In order to avoid self-intersections of the morphs, the vertices should be ordered identically, in counterclockwise order for example. Furthermore the start vertices should either correspond to a particular feature of



**Fig. 4.** 2 successive morphs: crocodile–hedgehog–sea horse (1536 points)

the curves, or they should correspond geographically. For example select the vertices with minimal x-coordinate. In Fig. 4, we select the nose of the animals as starting vertex.

The selection of more than one corresponding vertex is possible with our method by selecting the vertices of the coarse polygon in the following way. Before morphing between source and target polygon, a multiresolution analysis is performed following the algorithms described in Sect. 3.1. Since the MR analysis is based on subsampling, the vertices of the coarse polygon belong to the fine polygon and  $2^n - 1$  vertices lie in between. These particular relations between the fine and coarse polygons can now be used to fix manually all the vertices of the coarse source and target curves as corresponding points for the morphing. Once the user has manually selected three or four vertex indices destined to construct the coarse polygon, the fine polygons are resampled so that a power of two number of vertices are lying between them. This is the way we proceed for Figs. 4 and 8. For the other Figs. 5, 6, and 7 we simply fixed the vertex with minimal x-coordinate as starting point.



**Fig. 5.** France – Germany (8192 points) at 5 resolution levels  $n=13$  (finest level), 10, 8, 6, 4

### *Multiresolution representation of all curves*

The morphing algorithm interpolates between intrinsic representations of the coarse polygon and of the detail coefficients. As result an entire multiresolution representation of each in between curve is provided. Thus all LoD (level of detail) applications like LoD display, compression, LoD editing etc. can be applied directly on all morphs without any extra computation. Figure 5 shows in the first row the MR morphing between the maps of France and Germany together with the morphs of the coarse polygons. The full resolution curves (8192 vertices) are obtained after a complete synthesis (Sect. 2.2). The other four rows correspond to lower resolution morphs obtained by partial synthesis.

### *As-rigid-as-possible MR morphing*

It can be observed in all tests we did, that the morphs behave natural and that the transformations are least-distorting. In fact, the way we interpolate the coarse polygons and detail coefficients makes our shape interpolation method to be “as-rigid-as-possible”. For the interpolation of the coarse polygons we use Alexa’s method [3] which approximates optimal least-distorting interpolation by some optimization. This method generally requires computing compatible triangulations between the source and target objects. This sometimes difficult and time consuming step is omitted in our case, since the coarse polygons are quite trivial (consisting of 1–3 triangle

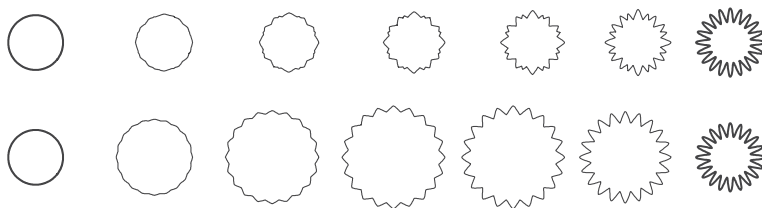


Fig. 6. Comparison MR morphing (upper row) – intrinsic morphing without MR (513 points) (lower row)

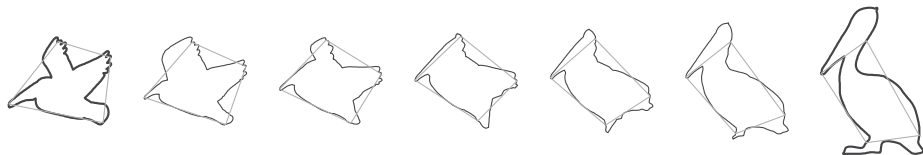


Fig. 7. Pelican (2048 points)

maximum). Furthermore, the details are transformed by linearly interpolating their intrinsic representations. Here also, the interpolation is least-distorting, since the angles vary linearly. In summary, our MR morphing method behaves “as-rigid-as-possible”. All figures testify this important property. In particular, a comparison between MR morphing and classical intrinsic morphing [16] shown in Fig. 6 clearly supports our claim.

### *A tricky example*

Although the method does not prevent explicitly self-intersections, it is nevertheless rare to observe self-intersections. In fact, the way we interpolate the coarse polygons guarantees self-intersection free transformations, but only of the coarse polygons. However, to test the robustness of our method with respect to this aspect, we use

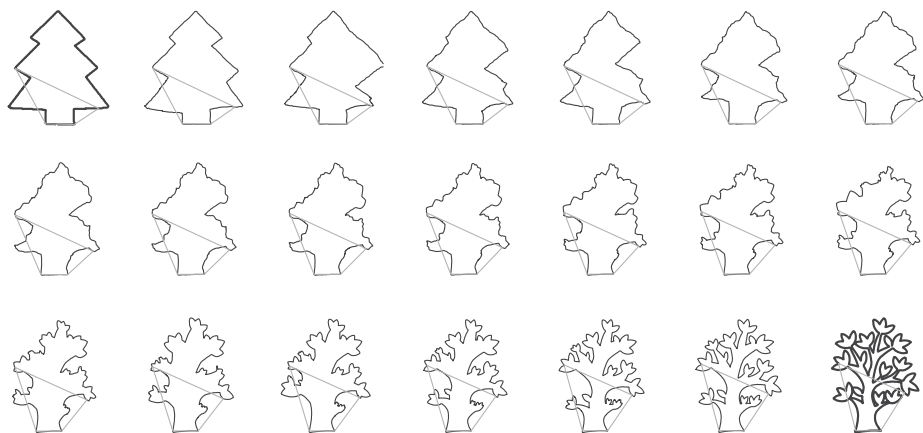
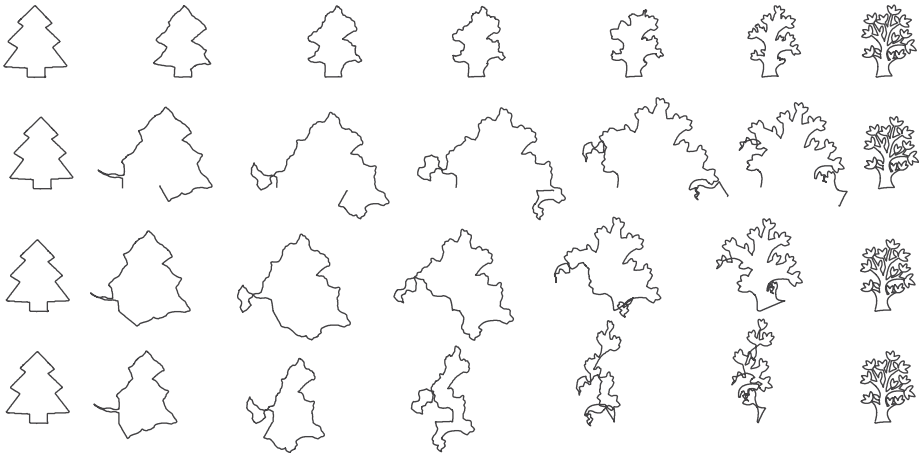


Fig. 8. Christmas tree – flower tree (2048 points)



**Fig. 9.** Comparison of 4 intrinsic morphing methods: First row: MR morphing. Second row: intrinsic morphing of  $(\theta, l)$ -coordinates of the fine polygons without MR decomposition. The polygons of the last two rows are obtained by closing the polygons of row two with two different methods. Third row: affine closing. Fourth row: edge-tweaking following [16]

a very tricky example shown in Fig. 8. No self-intersections occur in contrast to intrinsic morphing without multiresolution, see Fig. 9. Another problem of intrinsic morphing is the non-closure of the intermediate polygons. In [16], it has been stated, that the intermediate polygons always come near together. This might be true for morphs between similar shapes as illustrated in [16]. But in cases where the shapes are very different and complicate with a lot of details, the non-closure is a real problem since large parts of the morphs need to be corrected, see Fig. 9. Our method does not present this problem at all, since the polygons at the coarsest level are interpolated using a decomposition into triangles following [3].

### Acknowledgements

The work was partially supported by the European Community 6-th framework programm, with the Network of Excellence AIM@SHAPE (IST NoE No 506766) <http://www.aimatshape.net>. Thanks to Nicolas Szafran for creating examples by extracting contours of images.

### References

- [1] Adams, J. A.: The intrinsic method for curve definition. *Computer Aided Design* 7(4), 243–249 (1975).
- [2] Alexa, M.: Recent advances in mesh morphing. *Comput. Graph. Forum* 21(2), 173–196 (2002).
- [3] Alexa, M., Cohen-Or, D., Levin, D.: As-rigid-as-possible shape interpolation. In: *SIGGRAPH '00: Proc. 27th Annual Conf. on Computer Graphics and Interactive Techniques*. New York: ACM Press/Addison-Wesley 2000, pp. 157–164.
- [4] Bertram, M.: Single-knot wavelets for nonuniform b-splines. *Computer Aided Geometric* 22(9), 848–864 (2005)
- [5] Carmo, M. D.: *Differential geometry of curves and surfaces*. Prentice Hall 1976.
- [6] Chui, C., Quak, E.: Wavelets on a bounded interval. In: *Numerical Methods of Approximation Theory* (Braess, D., Schumaker, L., eds.). Basel: Birkhäuser 1992, pp. 1–24.

- [7] Elber, G., Gotsman, C.: Multiresolution control for nonuniform bspline curve editing. In: 3rd Pacific Graphics Conf. on Computer Graphics and Applications, Seoul, Korea, pp. 267–278, August 1995.
- [8] Finkelstein, A., Salesin, D. H.: Multiresolution curves. *Computer Graphics Proc. (SIGGRAPH 94)*, pp. 261–268 (1994).
- [9] Floater, M. S., Gotsman, C.: How to morph tilings injectively. *J. Comput. Appl. Math.* *101*(1–2), 117–129 (1999).
- [10] Goldstein, E., Gotsman, C.: Polygon morphing using a multiresolution representation. In: *Graphics Interface '95*. Canadian Inf. Process. Soc., pp. 247–254 (1995).
- [11] Hahmann, S., Bonneau, G.-P., Sauvage, B.: Area preserving deformation of multiresolution curves. *Computer Aided Geometric* *22*(4), 249–267 (2005).
- [12] Lee, A. W. F., Dobkin, D., Sweldens, W., Schröder, P.: Multiresolution mesh morphing. *Computer Graphics Proc. (SIGGRAPH 99)*, 343–350, (1999).
- [13] Lipman, Y., Sorkine, O., Levin, D., Cohen-Or, D.: Linear rotation-invariant coordinates for meshes. *ACM Trans. Graph.* *24*(3), 479–487 (2005).
- [14] Mallat, S. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence* *11*, 674–693 (1989).
- [15] Meyers, D. Multiresolution tiling. *Computer Graphics Forum* *13*(5), 325–340 (1994).
- [16] Sederberg, T., Gao, P., Wang, G., Mu, H.: 2-D shape blending: An intrinsic solution to the vertex path problem. *Computer Graphics (SIGGRAPH 93 Proc.)* *27*, 15–18 (1993).
- [17] Sederberg, T. W., Greenwood, E.: A physical based approach to 2-D shape bending. *Computer Graphics (SIGGRAPH '92 Proc.)* *26*(2), 25–34 (1992).
- [18] Shapira, M., Rappoport, A.: Shape blending using the star-skeleton representation. *IEEE Comput. Graph. Appl.* *15*(2), 44–50 (1995).
- [19] Stollnitz, E., DeRose, T., Salesin, D.: *Wavelets for computer graphics: Theory and applications*. Morgan Kaufmann 1996.
- [20] Surazhsky, V., Gotsman, C.: Intrinsic morphing of compatible triangulations. *Int. J. Shape Model* *9*(2), 191–201 (2003).
- [21] Sweldens, W.: The lifting scheme, A construction of second generation wavelets. *SIAM J. Math. Anal.* *29*(2), 511–546 (1997).

S. Hahmann, B. Caramiaux and M. Cornillac  
Laboratoire Jean Kuntzmann  
Institut National Polytechnique de Grenoble  
B.P. 53, 38041 Grenoble Cedex 9  
France  
e-mails: stefanie.hahmann@imag.fr  
baptiste.caramiaux@imag.fr  
melanie.cornillac@imag.fr

G.-P. Bonneau  
Laboratoire Jean Kuntzmann  
Univeristé Joseph Fourier  
INRIA Rhône-Alpes  
655 avenue de l'Europe  
Montbonnot  
38334 Saint Ismier Cedex, France  
e-mail: georges-pierre.bonneau@ujf-grenoble.fr