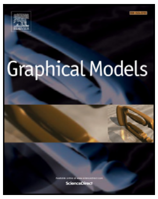
Graphical Models 85 (2016) 37–45

Contents lists available at ScienceDirect

Graphical Models

journal homepage: www.elsevier.com/locate/gmod



Skeleton-guided 3D shape distance field metamorphosis

Bo Wu[a], Kai Xu[a,*], Yang Zhou[b], Yueshan Xiong[c], Hui Huang[d]

[a] HPCL, National University of Defense Technology, China
[b] Shenzhen VisuCA Key Lab / SIAT, China
[c] HPCL, National University of Defense Technology, China
[d] Shenzhen University, Shenzhen VisuCA Key Lab / SIAT, China

ARTICLE INFO

Article history:
Received 1 December 2015
Revised 21 February 2016
Accepted 15 March 2016
Available online 26 March 2016

Keywords:
Metamorphosis
Skeleton
Distance field
Correspondence

ABSTRACT

We introduce an automatic 3D shape morphing method without the need of manually placing anchor correspondence points. Given a source and a target shape, we extract their skeletons and automatically estimate the meaningful anchor points based on their skeleton node correspondences. Based on the anchors, dense correspondences between the interior of source and target shape can be established using earth mover's distance (EMD) optimization. Skeleton node correspondence, estimated with a voting-based method, leads to part correspondence which can be used to confine the dense correspondence within matched part pairs. This produces smooth and plausible morphing sequence based on distance field interpolation (DFI). We demonstrate the effectiveness of our algorithm over shapes with large geometric variation and structural difference.

1. Introduction

Shape morphing (shape interpolation) [1,11,15,27,31–33] is a widely studied problem in both fields of computer vision and computer graphics. Smooth transition between 2D objects or even 3D shapes is very useful in many applications, such as special effects in film industry. In general, shape morphing encompasses both pose transition and shape variation, making it difficult for input objects with different shapes with arbitrary poses.

In computer graphics, shape morphing approaches can be classified according to the representation of input objects. Surface-based methods [1,33] are used for interpolating between meshes sharing same topology. On the other hand, volumetric (e.g. distance field method) methods [11,32] represent object as a grid of distance field with respect to the surface. By interpolating the distance values between source and target, it can generate grids with in-between distance values. At the cost of expensive computation, distance field interpolation (DFI) method can handle objects with different topology, as an advantage over explicit surface methods. Implicit surfaces [14,20,22,26], defined by continuous functions are also well suited for morphing shapes of arbitrary topologies, but the components from the two input shapes need to be paired.

The core issue of shape morphing is to establish meaningful correspondences between the two input objects. Surface-based approach usually requires dense correspondence between source and target meshes. DFI methods do not require dense correspondences at the beginning, however, it usually needs user to specify anchor points (usually more than a dozen on each object) manually [11,32]. To generate satisfactory in-betweens, these anchor points should be carefully placed in semantically meaningful positions on both source and target objects, which is a tedious task for human demanding many rounds of trial-and-error.

In this paper, we propose an automatic DFI morphing method in part wise with the assistance of shape skeleton. The correspondences of skeletal feature nodes (i.e. junction and terminal nodes) are first established using a method similar to [5]. Based on the matching of these feature nodes, the correspondences between skeleton branches can be determined. By parameterizing and sampling the matched skeleton branches, more corresponding point pairs can be extracted as anchors. In our approach, skeleton is exploited in two ways. First, it helps automatically determine where to place anchor points according to corresponding nodes on source and target skeletons. Second, during skeleton construction [38], shapes can be segmented into meaningful parts associated with skeleton branches. This association facilitates the construction of dense correspondence at part level based on earth mover's distance (EMD) optimization [19]. In consequence, our algorithm produces visually pleasing morphing results without tedious user interaction.

---

* Corresponding author. Tel.: +8618673358686.
E-mail addresses: wubo.gfkd@gmail.com (B. Wu), kevin.kai.xu@gmail.com (K. Xu), zhouyang@siat.ac.cn (Y. Zhou), ysxiong@nudt.edu.cn (Y. Xiong), hhzhiyan@gmail.com (H. Huang).

http://dx.doi.org/10.1016/j.gmod.2016.03.003

Our key observation is that the skeleton of an object, encoding both geometric and structural information, offers meaningful positioning of anchor points. Besides, part-based dense correspondence, in contrast to whole-shape correspondence [32], alleviates artifacts caused by correspondence drifts in EMD optimization. Although based on part-level correspondence, our method generates smooth and plausible transitions between the whole shape of source and target objects. This distinguishes from the work of Alhashim et al. [2] which also uses skeleton but generates in-betweens based on part-wise blending.

2. Related work

Shape morphing is closely related to shape correspondence problem. Extensive literature exists on both shape morphing and shape correspondence topics. In this section, we briefly review related shape morphing methods and shape matching on skeleton.

Shape morphing. Since shape morphing is a useful tool in many applications, researches on both 2D and 3D morphing start quite early [7,11,21,22]. Morphing methods can be classified into several kinds, e.g. surface mesh method, implicit surface method and DFI etc., according to how objects are represented. However, such methods rely heavily on the quality of correspondences to produce plausible morphing sequences. Surface mesh morphing approach [1,33] needs dense correspondences for each vertex on source and target objects, while DFI [11,32] needs correspondences of voxels on grids. Without bothering with voxels, implicit surface methods [20,27] defined by scalar functions, can also morph objects with different topologies. In order to save effort on vertex correspondence in surface based morphing, Blanz and Vetter [9] proposed a morphable model with full correspondence to synthesize new faces from given examples. Similar work [3] has also been done for human poses. But such model is not applicable to DFI method, as it is implicit. In practice, given only sparse correspondences, some strategies can be used to generate fuzzy dense correspondence to make DFI work [11,32].

Many works have been done to find a reasonable trajectory for morphing. The well known one is done by Alexa et al. [1], in which they proposed a new way to decompose the deformation gradient matrix into a rigid part and a stretching part. By interpolating each part separately to get a new deformation matrix, the blending path is visually reasonable. This method, however, was designed for surface or volumetric mesh, and is not straightforward for DFI. Weng [32] introduced such method into DFI by essentially constructing a volume mesh from grid, and their results are promising. Besides, Xu et al. [33] introduced a method to obtain shape blending sequences by solving Poisson equation. But the deformation matrix decomposition is similar to [1]. There are also some works [12,16,17,28] embedded physical model constraint into the morphing process, in order to produce physically plausible transformation sequences. Another recent work [15] proposed a data-driven method to yield reasonable morphing results. More recently, Von-Tycowicz et al. [31] exploited a set of shapes to generate real-time non-linear shape interpolation. Obviously, such methods depend on a dataset, in which all models are fully corresponding to each other. On the contrary, this also limits the method for arbitrary shapes.

Skeleton matching. Shape blending results greatly depend on the quality of correspondence. Establishing meaningful shape correspondence is a difficult problem, especially semantically similar objects may vary significantly in both geometry and topology. Shape matching is extensively studied and comprehensive review of this topic is beyond the scope of this paper. Please refer to survey papers like [30] for more details. In this paper, we focus on the correspondence of shape skeleton.

Curve skeleton contains both geometry and structure information about the shape. But how to efficiently extract skeleton from arbitrary shape is nontrivial. Several methods [4,29] have been proposed to automatically extract high quality shape skeleton. Even recently, Zhou et al. [38] proposed a method to decompose shape into approximate generalized cylinders and one of its applications is to construct skeleton. In this paper, we employ this method to construct skeleton as it also gives us part information. Skeleton is also very useful in many aspects. Zheng et al. [37] made use of consensus skeleton to register point cloud with noise and occlusion. Jiang et al. [18] exploited skeleton to detect intrinsic symmetry of point clouds.

Establishing skeleton correspondences is a challenging task. Bai and Latecki [6] proposed a geodesic distance based skeleton matching algorithm, in which they only considered the match of skeleton endpoints. To avoid extensive computation, Au et al.[5] introduced a voting strategy which used several geometric metrics to search skeleton node correspondences. However, few works have been done to exploit skeleton for morphing. Blanding et al. [8] simply regarded the medial axis as the intermediate shapes. Lian and Xiao [23] made use of skeleton, strokes and key points to interpolate the same Chinese character in different fonts. Alhashim et al. [2] also exploited skeleton correspondence to generate creative shapes. We prefer using skeleton correspondence not only because it is computationally more efficient than vertex correspondence [36], but skeleton nodes often locate in the meaningful positions of a shape.

3. DFI morphing

Given a source and a target surface meshes, their occupied spaces are first uniformly voxelized as volume objects V_0 and V_1 respectively. For i^{th} voxel $v_i^0 \in V_0 (v_i^1 \in V_1)$, it has a unique coordinate in source (target) domain and $D_0(v_i)$ $(D_1(v_i))$ is distance to its surface, in which positive value means outside, and negative means inside. Suppose there exists a third uniformly voxelized space domain V_t at time t $(t \in [0, 1])$, DFI method use warping and interpolation steps to reconstruct intermediate shapes by assigning distance values to voxels in V_t [11].

The warping step can be divided into *forward* warping and *backward* warping. Given a bunch of voxel correspondences, a forward warping function can be computed to transform source domain V_0 or target domain V_1 to the intermediate voxel domain V_t. The backward warping function, on the contrary, deforms intermediate voxel domain to source domain or target domain. Specifically, the forward warping function from source domain to target domain (or from target to source) is first computed according to the given voxel correspondences between V_0 and V_1. Applying the forward warping function to all source (target) domain, each voxel in source (target) will have a corresponding positions in target (source):

$$F_1^0(V_0) \approx V_1, \qquad F_0^1(V_1) \approx V_0 \tag{1}$$

Note that F_1^0 is unnecessary same as the inverse of F_0^1, and vice versa. With F_1^0 and F_0^1 in hand, different interpolation schemes could be exploited to obtain intermediate voxels V_t^0 and V_t^1 separately. Then forward warping functions F_t^0 and F_t^1 can be applied to update their correspondences in intermediate voxel domain V_t, and its backward warping functions B_0^t and B_1^t can also be obtained in a similar way. Finally applying the backward warping functions, correspondences of all V_t voxels in source domain and target domain are achieved:

$$B_0^t(V_t) \approx V_0, \qquad B_1^t(V_t) \approx V_1 \tag{2}$$

Here as-rigid-as-possible (ARAP) interpolation method in [32] is used to calculate in-between voxel positions corresponding to