Computers & Graphics 51 (2015) 98–105

Contents lists available at ScienceDirect

Computers & Graphics

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

A region-based expression tracking algorithm for spacetime faces

Jing Chi [a,b,c,*], Shanshan Gao [a,b,c], Yunfeng Zhang [a,b,c], Caiming Zhang [a,b,c]

[a] Department of Computer Science and Technology, Shandong University of Finance and Economics, Ji'nan, China
[b] Shandong Provincial Key Laboratory of Digital Media Technology, Ji'nan, China
[c] Digital Media Technology Research Institute, Shandong University of Finance and Economics, Ji'nan, China

ARTICLE INFO

Article history:
Received 11 March 2015
Received in revised form
15 May 2015
Accepted 16 May 2015
Available online 29 May 2015

Keywords:
Expression tracking
Spacetime meshes
Region-wise transformation
Normal constraint

ABSTRACT

We propose a novel region-based algorithm to quickly track time-varying facial expressions. Our algorithm takes a sequence of spacetime face meshes captured at video rate as input, and uses a template mesh to register each input face mesh to track facial dynamic expressions. In each mesh registration, we first automatically segment the template mesh into three initial regions with a metric, then for each initial region, segment it into different subregions and optimize a non-rigid affine transformation for each subregion with a variation of non-rigid ICP framework so as to approximate closely the input mesh. Region-wise non-rigid transformation decreases the unknowns in optimization significantly, and initial region segmentation decreases unnecessary optimizations and the template vertices involved in optimization greatly, so our algorithm has high speed. In addition, we introduce a new normal constraint in the non-rigid ICP framework to estimate more reasonable region-wise non-rigid transformation. The normal constraint makes our algorithm not only maintain good tracking accuracy while improving tracking speed, but also achieve automatic tracking without manual intervention. Experimental results show the efficiency of our algorithm.

1. Introduction

Facial expression tracking based on spacetime meshes is a difficult and challenging problem in computer graphics. Recent progress in real-time 3D face acquisition techniques makes it simple and convenient to capture a sequence of spacetime non-rigidly deforming face meshes with video rate. These sapcetime meshes record the shape and dynamic expressions of a real person's face. Such a sequence of meshes is inherently unstructured since the captured mesh at each time frame has different topology (vertex number and connectivity). This limitation makes it unable to establish inter-frame vertex correspondences, and consequently, to track expressional dynamics accurately and reanimate the captured expressions. Thus, it is necessary to reconstruct these captured meshes to create new meshes that have the same topology, only with different vertex positions. The new mesh sequence reproduces expressional dynamics, and is easy to integrate into a single time-varying mesh model on which further processing can be effectively supported, e.g., hole filling, finite element analysis, and expression editing.

The complex anatomical structure of human face allows for a great quantity of subtle expressions, so a highly accurate tracking algorithm is needed to simulate expressions with all its subtleties. Additionally, many practical applications, e.g., teleconference and videophone, demand high tracking speed, so a fast tracking algorithm is needed in this case. In this paper, our goal is to propose an efficient algorithm that tracks expressional dynamics quickly, meanwhile, achieves good accuracy, so as to meet the requirement in the above applications.

Refs. [1–5] introduce high-resolution deformable template models for expression tracking. Amberg et al. [4] extend the ICP algorithm to non-rigid registration by incorporating adjustable stiffness weights. They assign an affine transformation for each vertex of the template model. Wang et al. [6] introduce region-wise deformations based on physical face model. Their segmented regions are fixed. In our proposed algorithm in this paper, the region segmentation is dynamically determined and can be adjusted flexibly.

Refs. [7–10] track subtle expressional details such as furrows, wrinkles and pores by using facial motion capture data. The data capture of subtle facial movements need to place or paint markers on the human face. Le et al. [11] investigate the optimization of facial mocap marker layouts to capture more accurate minute details. Chi et al. [12] track facial subtleties via Laplacian smooth and multi-scale mesh matching.

Refs. [13–15] use optical flow to achieve automatic tracking of spacetime faces. Optical flow is prone to errors during rapid

* Corresponding author at: Department of Computer Science and Technology, Shandong University of Finance and Economics, ji'nan, China.
E-mail address: peace_world_cj@hotmail.com (J. Chi).

http://dx.doi.org/10.1016/j.cag.2015.05.019
0097-8493/

deformation, and inaccurate in some cases, e.g., occlusions, insufficient image details, and appearance changes. Furthermore, optical flow may cause drift. Beeler et al. [16] propose an anchor-based reconstruction with image-space tracking that can bound temporal drift, correctly handle occlusion and motion blur.

Refs. [17–20] compute a coherent deforming mesh model directly from the captured sequence of spacetime face meshes in 4D space. The spatio-temporal coherent model can well fit through all the time frames in the captured sequence, but the model computation is generally complex since it is performed in 4D space.

In this paper, we propose a novel region-based algorithm for tracking time-varying facial expressions quickly and accurately. Our algorithm inputs a face mesh sequence captured at video rate, and uses a deformable template mesh model to register each input mesh so as to output a new sequence of high-resolution meshes that have the same topology and accurately reflect facial shape and dynamic expressions. We introduce region-based idea into mesh registration that represents inter-mesh deformations as region-wise non-rigid deformations. We automatically segment the template mesh into different regions while determining a non-rigid affine transformation for each region so as to deform the template mesh towards the input mesh closely. Additionally, we introduce a new normal constraint in the non-rigid ICP framework for estimating region-wise non-rigid transformation. *Contributions*: (1) It just computes one affine transformation for each region of the template mesh, which significantly reduces the unknowns to be determined, so our algorithm has high speed and low space complexity. (2) Introduction of a normal constraint for optimizing region-wise non-rigid transformations can effectively improve the registration accuracy, and achieve automatic tracking. (3) Our algorithm can adjust region segmentation of the template mesh automatically and flexibly to register arbitrary deformations on the input mesh with desired speed and accuracy.

2. Algorithm overview

Supposing that the input sequence captured at video rate is made up of M frames, and at frame i, the mesh with N_i vertices, for convenience, we call it target mesh, is given as $T_i = \{q_j\}$, $j = 1, 2, \ldots, N_i$. Our goal is to fit through the input sequence using a template mesh with N vertices given as $S = \{v_k\}$, $k = 1, 2, \ldots, N$, where v_k is the template vertex.

At each frame, firstly, we automatically divide the template mesh into three initial regions with a distance metric. Then, for each initial region, we further automatically segment it into several subregions and compute one affine transformation for each subregion. We assume that, to register the target mesh T_i, the template mesh S is finally segmented into L subregions, each is represented as S_l, $l = 1, 2, \ldots, L$, and the affine transformation computed for each subregion S_l is represented as X_l. Applying these region-wise affine transformations to S, we obtain a new mesh S' as follows:

$$S' = \{X_k v_k\},\ k = 1, 2, \ldots, N, \quad X_k \in \{X_1, \ldots, X_L\}$$

The mesh S' closely approximates T_i and maintains the topology of S. Thus, using the template mesh S to register each target mesh, we can obtain a sequence of new meshes that have the same topology and reflects facial expression dynamics.

3. Region segmentation and region-wise non-rigid transformations computation

3.1. The region segmentation idea

When registering the template mesh to the target mesh, if we divide the template mesh into different regions and each individually deforms to approximate the target mesh, then it equals to overlay many patches (regions) on the target mesh. Obviously, the more the patches, the more closely they overlay the target mesh. So we can recursively divide the patches into more smaller patches to improve the closeness, in extreme case, each patch consists of only one template vertex and has one affine transformation to displace it onto the target mesh. Therefore, within permissible error range we can segment the template mesh into many regions and determine one affine transformation for each region to approximately match the target mesh, i.e., represent the non-rigid deformations between meshes as group of region-wise non-rigid deformations. The basic segmentation progress is that: firstly, we compute a non-rigid affine transformation on the template mesh, then, reject from the region any template vertices that transform to the target mesh beyond a given error threshold. Once a region is computed we iterate this process until all vertices on the template mesh have been segmented.

Region segmentation significantly reduces the unknowns to be determined because that it computes an unknown affine transformation for each region rather than for each template vertex. This not only improves the computational speed, but also saves the storage space for unknown affine transformations. Therefore, our algorithm has low time and space complexity. It is noted that, in region segmentation, the first affine transformation computation conducts on the whole template mesh. This computation takes time since it involves in all the template vertices. Moreover, this computation is in vain since it is nearly impossible to apply one affine transformation on the whole template mesh to approximate the target mesh closely. Therefore, to avoid unnecessary computation and further increase the speed of algorithm, we introduce initial region segmentation before estimating region-wise non-rigid transformations.

3.2. Initial region segmentation

The input mesh sequence of our algorithm is captured at video rate, so arbitrary two adjacent meshes are very close in space and time, and the deformations between them are very small. We can assume that there are no deformations between two adjacent meshes in extreme case. In such a case, we can consider that the two mesh surfaces are coincident, i.e., for each vertex on one mesh, the distance between it and its closest point on the other mesh is zero. Therefore, we use the distance between corresponding points on two adjacent meshes to measure deformations between them. For a vertex on one mesh, if the distance between it and its closest point on the other mesh, for convenience, we call it closest distance, is small, then the deformation on the vertex that transforms it onto the other mesh is small, and vice versa. We call this metric the closest distance metric. Thus, when we register the template mesh S to the target mesh T_i, we can classify the vertices on S according to their closest distances to generate three initial regions as follows:

$$\begin{aligned} R_1 &= \{v_k \mid d_{Min} \le d_k < d_{Min} + d/6\}, \\ R_2 &= \{v_k \mid d_{Min} + d/6 \le d_k < d_{Min} + d/3\}, \\ R_3 &= \{v_k \mid d_{Min} + d/3 \le d_k < d_{Max}\}. \end{aligned} \tag{1}$$

where v_k is the vertex on S, d_k is the closest distance for v_k. d_{Min} and d_{Max} are respectively the minimum and maximum among all the closest distances. $d = d_{Max} - d_{Min}$. Because the adjacent target meshes are very close spatially, if the template mesh S has fitted to the target mesh T_i closely, then it will be very near to the next target mesh T_{i+1} and suitable to the closest distance metric. Therefore, the initial region segmentation can adapt to the inter-mesh deformations over time.

In Eqs. (1), we do not divide the interval $[d_{Min}, d_{Max}]$ into equal parts to generate initial regions because that, in general, the