Eurographics/ ACM SIGGRAPH Symposium on Computer Animation (2007), pp. 1–2
D. Metaxas and J. Popovic (Editors)

Towards Efficient 3D Facial Appearance Models

K. A. Sidorov[1,2], A. D. Marshall[1], P. L. Rosin[1], S. Richmond[2]

[1]School of Computer Science, Cardiff University, UK
[2]School of Dentistry, Cardiff University, UK

Abstract

We present a novel approach to building realistic 3D facial appearance models. We seek to efficiently model temporal dynamics of a single human face by encoding raw 3D geometrical data and texture from scanners in a model suitable for statistical treatment. Preparation of our models requires minimal human interaction and, once built, our models can be used for real-time facial animation intuitively controlled by a small number of parameters, ready for rendering on consumer-class graphics hardware. Our method robustly handles poor-quality or deficient sample meshes and allows for processing of shape data in 2D mode using established image processing algorithms.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism — Animation

1. Introduction

The statistical appearance models suitable for photorealistic 2D facial animation is an established computer vision technique [CE01]. Due to the recent advances in 3D scanning technology the need for compact and easily controllable 3D appearance models is becoming apparent. While other approaches to 3D facial modelling exist [BV99], they require laborious manual preparation of the source data. We seek to build appearance models with minimal user interaction. Incorporating 3D geometrical data in such models is, however, a non-trivial task: existing scanners that rely on non-invasive markerless acquisition process produce a point cloud in $\mathbb{R}^3$ as output, which is further triangulated to produce a polygonal mesh; when scanning of a subject is performed sequentially to capture animation, the resulting point clouds inevitably have varying configurations and, consequently, their triangulations have different topologies, as the existing approaches to sequential scanning treat triangulation of each frame's point cloud as a separate problem. In addition, resulting meshes are often deficient — they contain missing triangles or suffer from nonuniform density of point clouds where the scanner fails: inner mouth, eyes and hair are typical problematic areas.

Real-time applications of 3D appearance models require them to have little computational overhead over a comparable 2D model. Here, we introduce an efficient method to entirely encode 3D shape and texture data in a combined eigenmodel. We achieve computation efficiency by separating the shape data into a fixed-topology low-resolution mesh and the difference between the original mesh and the low-resolution mesh (represented by heightmaps), while solving a number of technical problems along the way. Benefits of our approach include:

- Deficient meshes can be handled easily (missing triangles can readily be interpolated).
- Geometrical data can be edited in 2D as pictures or be subjected to existing image processing algorithms.
- Model can be easily augmented with some extra information not present in the scanner data: for example, problematic appearance of the inner-mouth and eyes can be excluded from the model and modelled differently.
- Model preparation can be partially delegated to graphics hardware (texture warping and heightmap computations).
- Rendering with dynamic level of detail is straightforward (by regeneration of meshes with desired resolution from heightmaps).

2. Representation of Shape and Texture

We record and combine textural information from multiple cameras into a single rectangular texture map (cylindrical projection) per frame. We then place a small number of landmarks (control points) on the 2D texture maps,

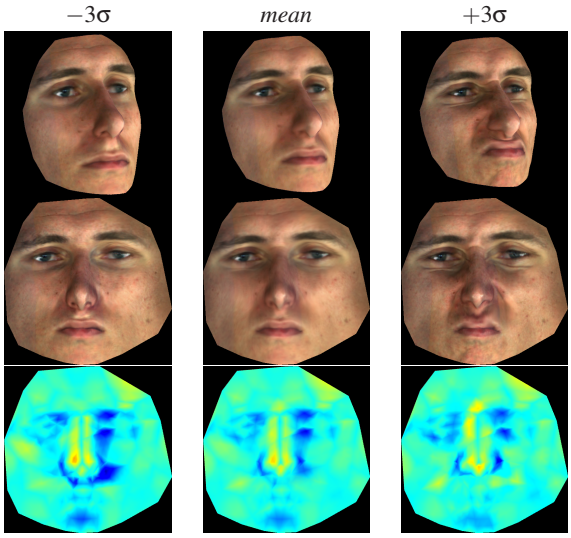

Figure 1: *First mode of variation. Complete model, texture and heightmaps.*

same in each frame. The tracking of a subset of landmarks, that can be precisely positioned by a human around fast-changing features, is achieved semi-automatically using the Downhill Simplex tracker described in [LT97]. An abundant number of candidate landmarks to cover the rest of the face are then seeded and subjected to automatic two-pass forward-backward tracking process (through frames $1 \ldots N \ldots 1$) using dynamic registration by means of normalised cross-correlation. The Euclidean distance between the ground truth position in frame 1 and the position after tracking is used as an error measure to optimise tracking parameters and reject unsuccessful candidates. Tracking of landmarks using optical flow estimation [BFBB92], [BV99] is also possible, but can be problematic. Triangulating the point cloud of landmarks gives us a very low-resolution mesh (a few tens of triangles). The textures are warped to the mean shape of the coarse meshes in order to force the same number of pixels per texture. This is done on the graphics hardware. Given a mapping from a 3D mesh to the 2D texture (as all vertices have texture coordinates also) we reconstruct the true 3D positions of the landmarks by computing their barycentric coordinates (which are the same in 3D and 2D) in the encompassing triangle on the texture.

We proceed to compute heightmaps for every triangle in the low-resolution mesh from landmarks. To do so, for every sampling point of the heightmap we cast a ray originating in a corresponding point of the triangle in the interpolated normal direction, and find the intersection of that ray with the original mesh, thus obtaining (signed) elevation of the original mesh above (or below) the low-resolution mesh. This process can be hardware-accelerated: for each triangle in the low-resolution mesh we place the camera at its barycenter, facing normally, and render the original mesh, then retrieve the contents of the depth-buffer. If we later need to

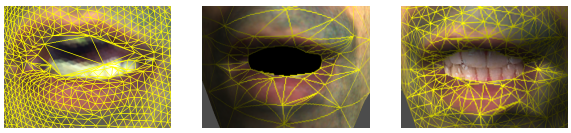

Figure 2: *Left: deficient source mesh. Middle: our model at low LOD, depth and texture information removed from the inner mouth. Right: LOD dynamically changed to high, depth and texture information substituted.*

dynamically regenerate the meshes for rendering at different LODs (Fig. 2), we store the height information as a set of height-textures of arbitrary resolution, sampling the elevation at every texel; the heightmaps can then be subjected to image processing algorithms to filter out noises, or interpolate depth information for missing triangles (Fig. 2), or add fine detail. Alternatively, to save memory, we create a new high-resolution mesh by pre-subdividing the coarse mesh and only sample the elevations at the nodes of the new mesh.

3. Model of 3D Appearance

Frames of animation are now represented by state vectors with the same number of components for every frame, ready for statistical treatment. We rely on an older idea of AAM [CE01]. If $\vec{s}_i = \{x_1, y_1, z_1, x_2, y_2, z_2, \ldots, x_{N_s}, y_{N_s}, z_{N_s}\}$ is a shape vector, encoding the 3D coordinates of all vertices in the low-resolution mesh, and if $\vec{h}_i = \{h_1, h_2, \ldots, h_{N_h}\}$ is a vector encoding heightmap values and $\vec{t}_i = \{r_1, g_1, b_1, r_2, g_2, b_2, \ldots, r_{N_t}, g_{N_t}, b_{N_t}\}$ encodes pixel values of the texture, then each frame of the original animation can be described by a state vector of $3N_s + N_h + 3N_t$ components $\vec{f}_i = \{\vec{s}_i, \vec{h}_i, \vec{t}_i\}$ and the observation matrix, encoding the entire animation, takes the form $\mathcal{A} = (\vec{f}_1, \vec{f}_2, \ldots, \vec{f}_N)^T$ We perform PCA to find the set of orthogonal modes of variation and manipulate the model as in [CE01].

4. Results

Utilising high-quality data, our method provides new levels of realism and control in 3D facial appearance modelling. Flexibility and efficiency of our model suggests wide range of applications in the area of computer graphics and animation.

References

[BFBB92] BARRON J., FLEET D., BEAUCHEMIN S., BURKITT T.: Performance of optical flow techniques. *CVPR 92* (1992).

[BV99] BLANZ V., VETTER T.: A morphable model for the synthesis of 3d faces. In *SIGGRAPH '99* (1999).

[CE01] COOTES T., EDWARDS G.: Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence 23* (2001), 681–685.

[LT97] LUETTIN J., THACKER N. A.: Speechreading using probabilistic models. *Computer Vision and Image Understanding: CVIU 65*, 2 (1997).