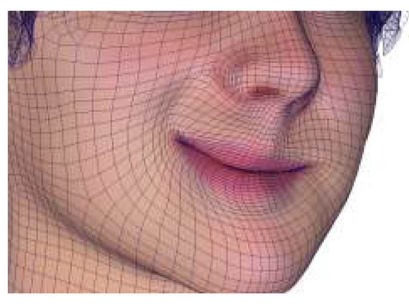
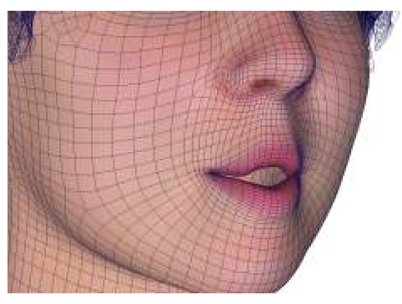
TECHNISCHE UNIVERSITÄT CHEMNITZ

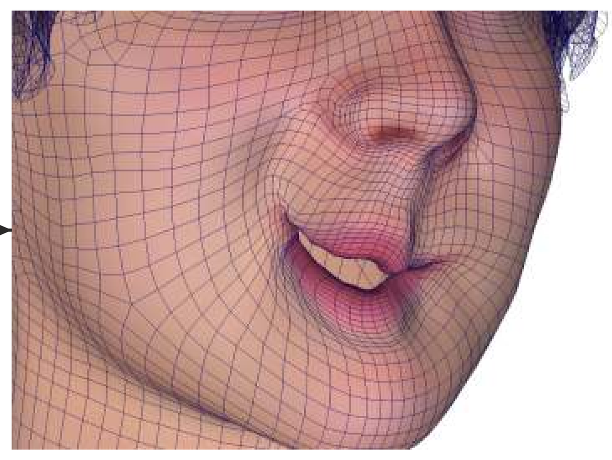
**Faculty of Computer Science**
Professorship of Computergraphics and Visualization
**Towards Point-based Facial Movement Simulation**
Tom Uhlmann and Guido Brunnett

# Towards Point-based Facial Movement Simulation

## Problem Statement


blendshape "right lip up"


blendshape "Ooh"

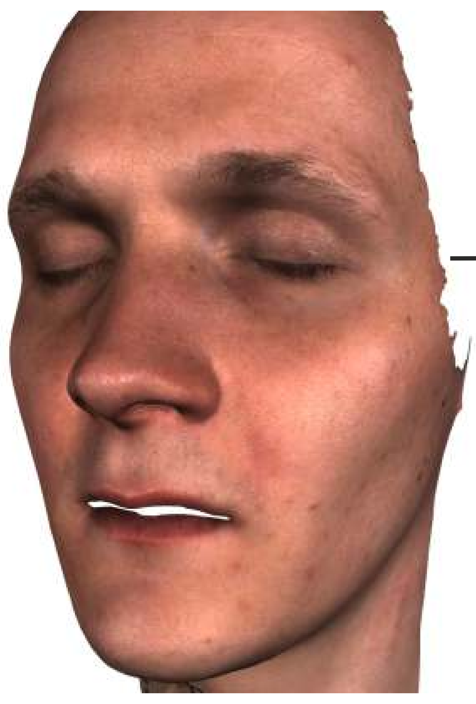

Combination of both blendshapes causes implausible deformations and self - intersections on a professional mesh.

The blendshape technique is almost ubiquitous in facial animation. A weighted sum of vector deformation fields is added to the base mesh in order to create a deformation. Hence, the deformations are applied independently onto the surface. This can lead to implausible deformations and self-intersections of the mesh if several blendshapes are activated simultaneously. This is depicted in the figures on the left where two simultaneously active blendshapes cause problems. Due to the simple computational model, error control is very limited or has to be performed on the mesh itself, which is computational demanding and therefore not suited for real time applications.

## Our Approach

We argue, that point-based simulation techniques are better suited to simulate and animate the facial tissue movement during facial movements. By using such an approach, the deformation of the surface is driven by the displacement of a sparse set of vectors. Hence, the individual facial movements are not applied directly to the surface, but to the control nodes. This results in an easier to control rig and simpler error control. Furthermore, these technique can simulate physical effects such as surface strain, which is not possible with blendshapes.

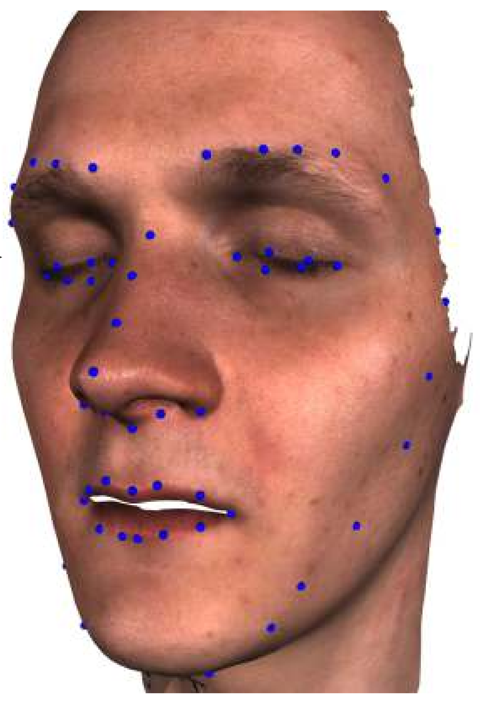
## Automatic Feature Point Detection


3D surface mesh

cylindrical projection


Feature point detection on 2d image

reprojection of landmarks onto the 3d model


3D Facial feature points

We use an automatic facial feature detection approach for our facial models. First we create a cylindric projection by sampling the surface using raycast and mesh intersection. Then we use OpenFace [ALS16] to detect facial landmarks on the flat projection. By reprojecting the detected points on the surface we retrieve exact 3D facial feature points.

## Point-based Simulation Method

We use the meshless finite element technique proposed by Adams et al. [AWO10] to create the deformations of our facial mesh. The technique uses a sparse set of control nodes strategically placed on the mesh's surface to control the deformation.

We use the sparse set of automatically detected facial feature points directly as control points. Thus, creating a facial expression or a viseme is a matter of displacing the control points.

## Facial Shape Alteration

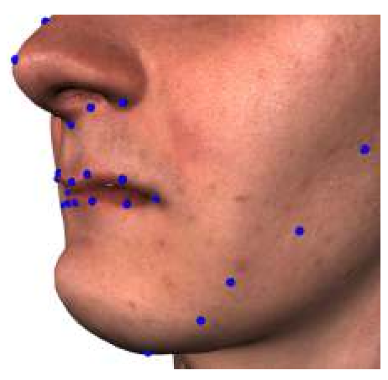

Only one control node has to be displaced to create this new nose shape.

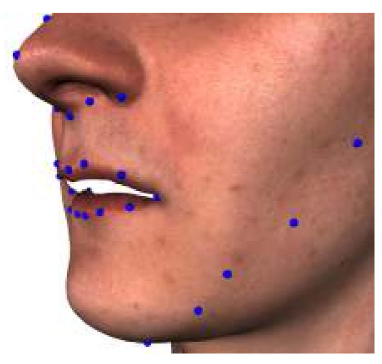
Our approach can also be used to easily alter the shape of the face. Here we change the shape of the nose simply moving the control node on the tip of the nose closer or further away from the face. The deformation field is stable even under large changes.


Large displacements of the control nodes don't disrupt the mesh.

## Facial Movement Simulation

The mesh's vertices are deformed by moving the control nodes. The strength of deformation decreases with increasing geodesic distance regarding the control node. This creates a smooth deformation suited for facial animation.




To close the mouth the surrounding control nodes simply have to be moved. The deformation extends to chin and cheeks creating a smooth deformation of affected facial tissue.

## Future Research

We are aiming for a facial movement simulation model with as little user input as possible. To that end, and to overcome current limitations, we want to extend our approach with the following features in the future:

1. Weights are currently based on a uniform geodesic distance. We want to learn correct weights from real data (scans, motion capturing) employing an analysis-by-synthesis approach in order to create a model which can reproduce particular facial movements and expressions by specifying a small set of parameters.

2. The current control node setting causes instability in the deformation field in regions with low coverage. We want to investigate this behavior further to create a sampling which creates a more stable deformation field.

## References

[ALS16] B. Amos, B. Ludwiczuk, M. Satyanarayanan, "Openface: A general-purpose face recognition library with mobile applications," CMU-CS-16-118, CMU School of Computer Science, Tech. Rep., 2016.

[AWO10] ADAMS B., WICKE M., OVSJANIKOV M., WAND M., SEIDEL H.-P., GUIBAS L. J.: Meshless shape and motion design for multiple deformable objects. Computer Graphics Forum 29, 1 (2010), 43–59. doi:10.1111/j.1467-8659.2009.01536.x. 1, 2