Creating Finite Element Models of Facial Soft Tissue

Mark Warburton
Department of Computer Science
The University of Sheffield
211 Portobello
Sheffield, S1 4DP, United Kingdom
M.Warburton@dcs.shef.ac.uk

Steve Maddock
Department of Computer Science
The University of Sheffield
211 Portobello
Sheffield, S1 4DP, United Kingdom
S.Maddock@sheffield.ac.uk

ABSTRACT

Physically-based animation techniques enable more realistic and accurate animation to be created. Such approaches require the creation of a complex simulation model that, for computer graphics applications, can efficiently produce realistic-looking animations. We present a process to automatically create animatable non-conforming hexahedral finite element (FE) simulation models of facial soft tissue, including automatic computation of skin layers and element material properties, muscle properties and boundary conditions, making them immediately ready for simulation. Using the GPU, the detailed models can simulate complex gross and fine-scale behaviour, such as wrinkling. Our process can also be used to create a multi-layered FE model of any object (not just soft tissue).

Keywords

physically-based modelling, soft-tissue modelling, facial modelling, physically-based animation, finite element method

1 INTRODUCTION

Facial modelling and animation is one of the most challenging areas of computer graphics. Currently, most facial animation requires performance-capture data, or models to be manipulated by artists. However, using a physically-based approach, the effects of muscle contractions can be propagated through the facial soft tissue to automatically deform the model in a more realistic and anatomical manner.

Physics-based soft-tissue simulation approaches often focus on either efficiently producing realistic-looking animations for computer graphics applications [TW90, KHYS02], or simulating models with high physical accuracy for studying soft-tissue behaviour [BJTM08, KSY08] or surgical simulation [KRG$^+$02, ZHD06]. The former normally simulate large areas, such as the face, using the efficient mass-spring (MS) method [TW90, KHYS02], or physics engines that focus on performance and stability [Fra12]. On the other hand, the latter tend to simulate more detailed models of smaller areas, like a block of skin, using the accurate but computationally complex finite element (FE) method [SNF05, FM08], or the FE-based but precomputation-heavy mass-tensor (MT) method [XLZH11]. Given increases in computational power, and the use of GPU computing architectures, complex FE simulations are now possible in real time [TCO08].

Physics-based simulations require an appropriate simulation model to be created. Such models can either conform to a surface mesh [MBTF03, BJTM08], or a non-conforming model, such as a voxel representation with a bound surface mesh, can be used [DGW11, WM12]. High-quality conforming models that can be efficiently simulated are often difficult and time-consuming to create, and require considerable manual work, although such models may be required for high-accuracy applications. In contrast, non-conforming models can enable more efficient production of stable, realistic-looking animations for computer graphics applications.

The aim of this work is to develop an automatic process to easily create animatable non-conforming hexahedral FE simulation models of facial soft tissue (the soft tissue between the skull and outer skin surface, as shown by Figure 1). In our previous work, basic models were generated that, for example, are unable to simulate anatomical muscle contraction or wrinkles [WM12]. Our current approach can generate much more detailed models that are able to simulate complex gross and fine-scale facial movement, including wrinkles. This approach includes automatic computation of skin layers and material properties, muscle properties, and boundary conditions (such as rigid nodes). The models are

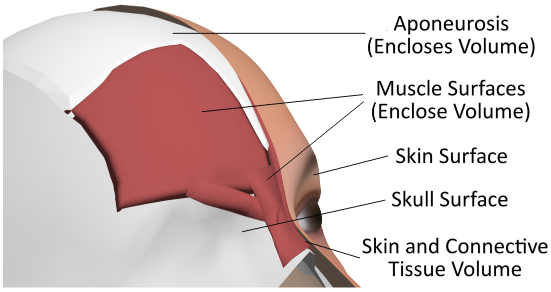

Figure 1: Surfaces and volumes of a facial soft-tissue model. The whole volume between the skull and skin surfaces (i.e. the skin and muscle volumes) is discretised to create an FE facial soft-tissue model.

optimised for GPU simulation, and can be used, for example, to efficiently produce realistic-looking facial animations for computer graphics applications. Our process can also create any multi-layered FE model from any surface mesh (not just soft-tissue models). The following sections detail relevant related work, followed by an overview of our physically-based animation approach, and a description and examples of the model creation process, including model simulation examples using our GPU-based FE system.

2 RELATED WORK

2.1 Physically-Based Facial and Soft-Tissue Models

Physically-based facial animation systems for computer graphics applications normally consist of muscle and skin models, sometimes along with a skull model and wrinkle models. For increased realism, a skull model can include a rotatable mandible [KHYS02]. Muscles have been modelled using vectors [Wat87], and more anatomically accurate geometric [KHYS02] and physics-based volumes [RP07]. Many muscle contraction models are based on a Hill-type model [RP07], some of which are biologically inspired [MHSH10], and the direction of contraction can be approximated as parallel to the central action curve [TZT09], or, more anatomically, by using a fibre field [SNF05].

Various facial soft-tissue models have been proposed, ranging from simple but efficient physics-engine-based [Fra12] and MS models [TW90, KHYS02], to more anatomical and realistic FE models [SNF05, ZHD06]. Detailed models of blocks of skin and soft tissue have also been created [KSY08], along with complex soft-tissue constitutive models [Bis06]. Due to its efficiency, the total Lagrangian explicit dynamic (TLED) formulation has been used for various non-linear FE soft-tissue simulations [TCO08], resulting in large speed-ups. The FE-based MT method has also been used to produce such simulations and also for facial surgical applications [XLZH11], showing similar accuracy to the FE method when simulating small displacements.

2.2 Model Creation Approaches

The model creation process is normally difficult and time-consuming, and it is also dependent on the required model structure. To create an FE model, a suitable simulation mesh must be created, and FE simulation properties, such as boundary conditions, must be set. We focus on simulation meshes constructed using 3D elements to model complex soft-tissue volumes, and these can be conforming or non-conforming (e.g. a voxel representation) with respect to the polygonal surface meshes used for visual purposes. Regarding element types for simulation, we only consider linear elements with a single integration point for optimal computational performance with the high-resolution models.

Simple automatic model creation approaches have been used that just create a layered MS model and skull from a surface mesh [TW90]. On the other hand, CT and MRI scans, or anthropometric data can be used to manually or automatically create an anatomical reference head model [KHYS02]. Such data from the Visible Human Dataset[1] has previously been used for reference model creation [SNF05]. Various techniques have been proposed to deform reference skull, muscle or full physically-based head models using manually defined landmarks [KHYS02, AZ10], although these often rely on good landmark placement. Kähler et al. also developed an interactive editor to enable easy muscle creation by processing user-specified grid points [KHYS02].

Numerous algorithms exist for fast automatic generation of high-quality tetrahedral models that conform to surface meshes [MBTF03, SG05]; however, 4-node tetrahedra are susceptible to volume locking, particularly when simulating incompressible materials like soft tissue. In contrast, reduced-integration 8-node hexahedral elements (with hourglass control) have increased stability and accuracy [WJC^{+}10], particularly when modelling non-linear anisotropic materials [LLT11], and can be used to create meshes using fewer elements, normally outweighing the efficiency of tetrahedra. Hexahedra are therefore often preferred for FE simulations.

Although various algorithms for producing conforming hexahedral meshes have been proposed [SKO^{+}10, ZHB10, NRP11], hexahedral mesh generation is often difficult and time consuming, and, without considerable manual work, many such algorithms suffer problems regarding element quality and robustness, particularly with complex geometries like soft tissue. Techniques have been proposed to improve the quality of hexahe-

[1] `http://www.nlm.nih.gov/research/visible/visible_human.html`

dral meshes [ISS09, SZM12], although these can produce models with an increased number of elements.

Simple hexahedral meshes can also be merged to produce a complex mesh [SSLS10, Lo12], although these approaches would require a manual decomposition of the complex model such that high quality elements are able to be produced during the merging process. Similarly, conforming and non-conforming domain decomposition FE methods can be used [Lam09], which involve performing an FE analysis on a model decomposed into several independent subdomains. Some other techniques involve deforming a reference hexahedral mesh [CPL00, LLT11], although such approaches require a high-quality reference mesh, and often also require manual work or modifications to the final mesh.

Alternatively, non-conforming hexahedral meshes are easier to create, for example, using voxelisation techniques, and a surface mesh can also be bound to the volume mesh for visual purposes. Such meshes can be used to create models for more stable and computationally efficient FE simulations [DGW11], which can also be optimised for more efficient simulation on the GPU [WM12]. Kumar et al. performed linear elastic FE simulations using structured non-conforming hexahedral grids, and compared these with conforming hexahedral simulation meshes [KPB08], which produced similar stresses, although only relatively simple models were examined. Non-conforming tetrahedral facial and soft-tissue FE models have also been used for stability and performance reasons [SNF05], although linear tetrahedral elements can cause problems such as volume locking.

Once a simulation mesh has been created, model properties, such as element material properties, must be specified. Lee et al. approximated such properties for non-conforming tetrahedral FE soft-tissue models using a sampling procedure [LST09]. A process to generate non-conforming hexahedral FE soft-tissue models has also been proposed [WM12], although this generates very basic models, and neither of these model creation approaches model skin layers (necessary to simulate wrinkles [FM08]), or the sliding of soft tissue over tough deep layers. Techniques have also been proposed to infer muscle fibre directions from B-spline volumes [TSB$^+$05] and conforming volumetric muscle meshes [MHSH10].

Extending existing work [WM12], our model creation process generates more detailed and accurate models, which include skin layers, accurate approximations of element material and muscle properties, and advanced boundary conditions, for example, that can model sliding effects. The models are able to simulate complex gross and fine-scale behaviour, including wrinkling. As well as computer graphics applications, due to the detail

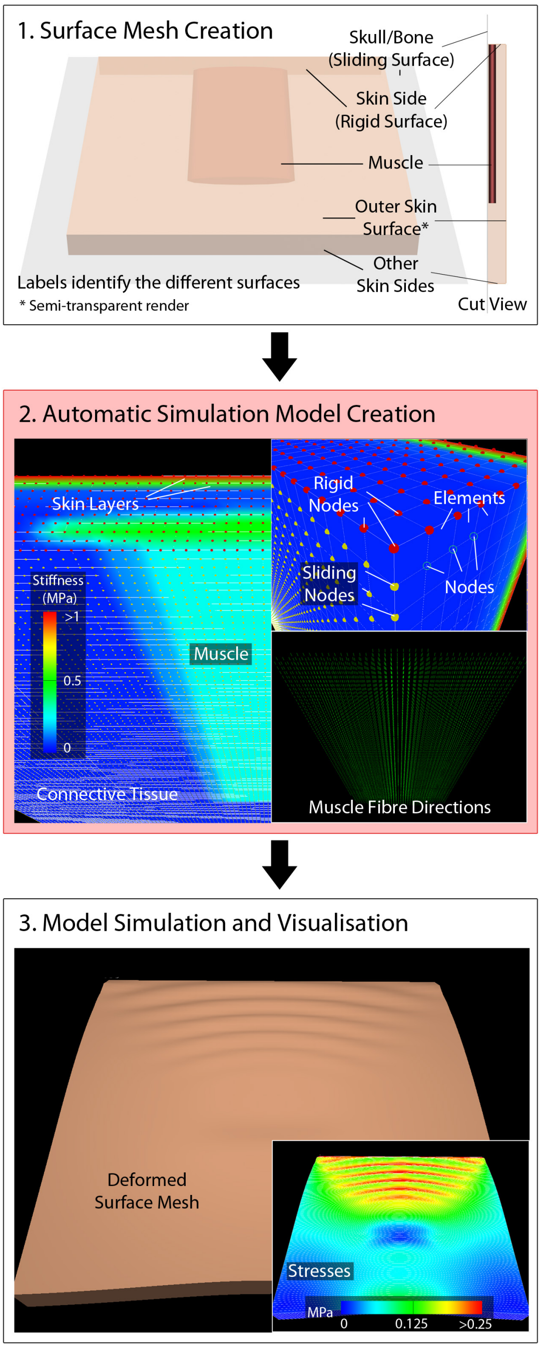

Figure 2: An overview of our physically-based animation approach. Simulation model creation is the focus of this paper.

and accuracy of the models, our creation process could also be useful in other fields, such as biomechanics and surgery.

3 OVERVIEW OF OUR PHYSICALLY-BASED ANIMATION APPROACH

Figure 2 shows an overview of our entire animation approach, which involves three major stages:

1. Creating the surface mesh for an object

2. Creating a suitable simulation model
3. Simulating and visualising the model over time

The surface mesh can be created using any 3D modelling software. The next stage (the focus of this paper) involves using our model creation system to automatically discretise the volumes enclosed by this mesh into a collection of nodes that are connected to form volumetric elements, creating a simulation mesh. FE model parameters are then computed to produce a complete simulation model. We use non-conforming hexahedral models due to model creation, performance and stability advantages [WM12], and surface meshes are bound to these for visual purposes. The models can then be simulated and visualised using a GPU-based FE simulation and visualisation system [WM13].

4 MODEL CREATION

Our model creation process involves five main stages:

1. Voxelising the multi-surface mesh
2. Computing skin layers and element material properties
3. Computing element muscle properties
4. Computing boundary conditions, such as rigid or constrained nodes
5. Binding the surface mesh to the simulation model

The surface mesh can contain various surfaces, including internal surfaces, and volumes are user defined by organising these into closed collections of surfaces. For example, with a facial mesh, there may be a volume for skin and connective tissue (between the skin and skull surfaces), and a volume for each muscle, as shown by Figures 1 and 2. Internal volumes, such as muscles inside the skin volume, overlap the volume they are contained within (i.e. the skin volume doesn't contain holes for the muscles), simplifying surface mesh creation. Volumes can also overlap, for example, to represent the blend of fibres between connecting muscles.

All mesh volumes are grouped into a number of user-defined *levels*, where level 0 is the highest level. Semantically, volumes in a lower level are contained within, and bound by, volumes in the level immediately above. For example, level 0 might consist solely of a skin and connective tissue volume, whereas level 1 might consist of the muscle volumes, which are contained within the skin volume. Other input consists of properties (such as material and muscle properties) associated with each volume, and model properties (such as voxel size). It should be noted that, for a standard soft-tissue model, the volumes and levels are known and can be automatically defined from appropriately labelled surfaces.

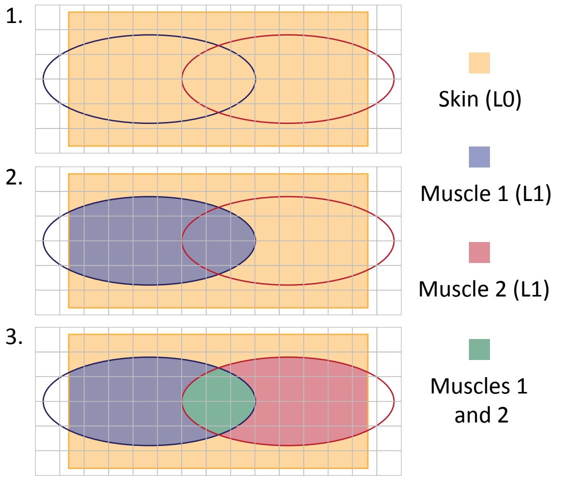

Figure 3: An example of the level-based voxelisation process for a skin block containing two muscles, showing the model state after each mesh volume has been considered in turn. Note this is a 2D illustration of a 3D process.

4.1 Voxelising the Surface Mesh

Starting with a grid of regularly- (cubes) or irregularly-shaped voxels (cuboids) that overlaps the entire surface mesh, voxel properties are calculated based on the proportions of overlap between the voxels and mesh volumes. Enclosed voxels with more than a user-defined proportion of overall overlap (with the union of all level 0 volumes) are used as hexahedral elements.

As illustrated by Figure 3, starting at level 0, sections of voxels that overlap a mesh volume in this level are assigned the properties of that volume. By iteratively considering the next level down, the properties of sections that overlap a mesh volume in both the current and previous levels are overwritten; hence, the properties of sections overlapping the skin volume that also overlap a muscle volume would be overwritten. When a voxel section overlaps multiple mesh volumes in the same level, the properties of these volumes are combined to model the the blend between materials.

From Figure 3, it can be seen that only lenient requirements are imposed on the creation of the surface mesh; for example, as muscles should be contained within the skin volume, parts of muscle surfaces that cross the bounds of this volume are appropriately ignored. Muscle surfaces can therefore simply penetrate the skull, rather than having to attach and conform to the skull surface. This simplifies the modelling of surface meshes by enabling less accurate surfaces to be used without affecting the simulation model.

Similar to some existing approaches for approximating proportions of overlap between element and mesh volumes [LST09], we use a sampling procedure to sample voxels (see Figure 4). Our level-based voxelisation process is performed by assigning the point samples prop-

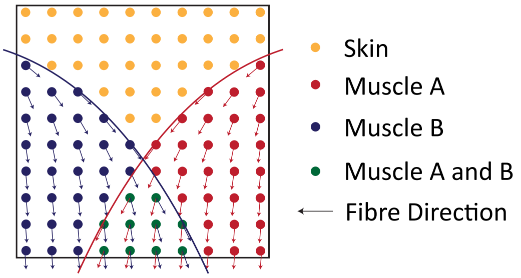

Figure 4: Element samples assigned material and muscle properties, which are used to calculate the overall element properties. Note this is a 2D illustration of a 3D process.

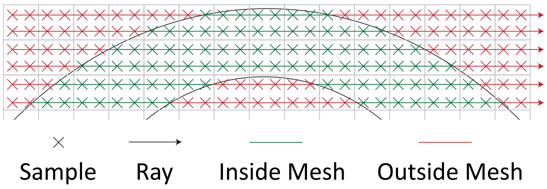

Figure 5: Efficient computation of whether samples are inside a mesh volume using few ray-surface intersections. Note this is a 2D illustration of a 3D process.

erties based on a weighted combination of those associated with level mesh volumes they are contained within:

$$w^{(s,v)} = \begin{cases} \frac{1}{n^{(s)}} & v \text{ encloses } s \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where $w^{(s,v)}$ is the weight of mesh volume v with sample s, and $n^{(s)}$ is the number of mesh volumes overlapping the sample. Ray-surface intersection tests determine whether a sample is inside a mesh volume. As shown by Figure 5, by uniformly sampling voxels (rather than, for example, randomly scattered samples [LST09]), a single ray and its surface intersection points can be used to efficently test each sample along an entire line.

4.2 Computing Element Material Properties

Using the voxel element samples, material properties associated with mesh volumes are weighted to calculate element material properties:

$$w^{(e,v)} = \frac{1}{n^{(e)}} \sum_{s=1}^{n^{(e)}} w^{(s,v)} \tag{2}$$

where $w^{(e,v)}$ is the weight of mesh volume v with element e, and $n^{(e)}$ is the number of samples in the element that are enclosed by at least one mesh volume. Within the skin volume, constant thickness layers with different material properties can be generated. Modelling skin layers is necessary to simulate fine details like wrinkles [FM08]. By defining a start depth and thickness for each layer, these can also contain overlapping regions, for example, to help capture the non-uniform blend between real skin layers. To determine which layers each sample inside the skin volume is contained within, the distances between the samples and the outer skin surface are tested, and the properties of samples are modified accordingly.

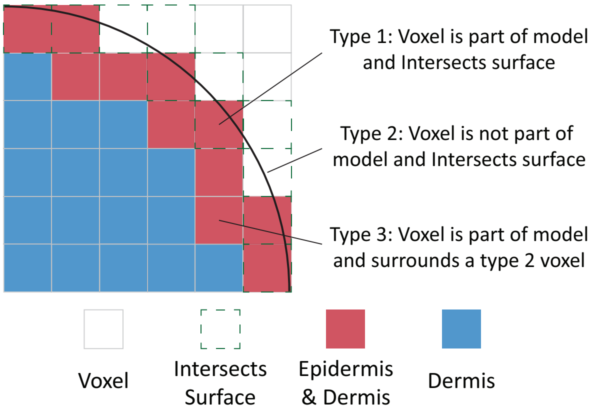
Type 1: Voxel is part of model and Intersects surface
Type 2: Voxel is not part of model and Intersects surface
Type 3: Voxel is part of model and surrounds a type 2 voxel
Voxel
Intersects Surface
Epidermis & Dermis
Dermis

Figure 6: Identification of elements that approximate the outer skin surface and therefore contain epidermal properties. Note this is a 2D illustration of a 3D process.

However, without using an extremely high element and sampling resolution, the thin outer epidermal layers are unlikely to be captured using such an approach, and these are therefore treated differently than the other thicker layers. For a single outer epidermal layer, the elements that approximate the outer skin surface are assigned a weight for the epidermal layer of t/e_{avg} (where t is the layer thickness, and e_{avg} is the average element dimension). As shown by Figure 6, such elements are identified as those that are intersected by the outer skin surface, or neighbour such a voxel that isn't included as part of the model (due to insufficient mesh volume overlap).

4.3 Computing Element Muscle Properties

As with material properties, element samples are used to weight muscle stresses for overlapping muscles (see equation 2). As shown by Figure 4, at each element sample, s, a fibre direction, $\mathbf{d}^{(\mathbf{s},\mathbf{m})}$ is also calculated for each muscle, m, that overlaps the sample, and these are averaged to produce a fibre direction, $\mathbf{d}^{(\mathbf{e},\mathbf{m})}$, for each muscle that overlaps the element:

$$\mathbf{d}^{(\mathbf{e},\mathbf{m})} = \frac{1}{n^{(e,m)}} \sum_{s=1}^{n^{(e,m)}} \frac{\mathbf{d}^{(\mathbf{s},\mathbf{m})}}{\|\mathbf{d}^{(\mathbf{s},\mathbf{m})}\|} \tag{3}$$

where $n^{(e,m)}$ is the number of element samples inside the muscle.

Sample fibre directions are calculated using NURBS volume approximations of the muscles, which are

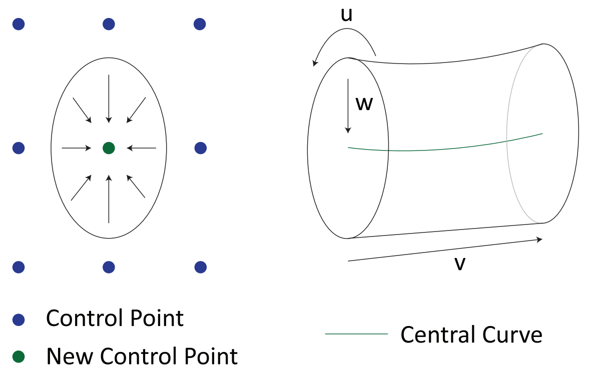

Figure 7: Creation of a NURBS volume by shrinking a NURBS surface, creating a 3rd dimension running from the NURBS surface to a central curve.

created by shrinking NURBS surfaces to their central curve [MLC01], as shown by Figure 7. Such surfaces must be closed along one dimension, and, for each row of control points along this dimension, a new control point is created at the centre of these points to create a central curve. A NURBS volume can then be created, the 3rd parametric dimension of which runs from the NURBS surface to this central curve.

The gradient of the NURBS volume, $V(\mathbf{u})$, with respect to the parametric coordinate along the length of the muscle, a, is used as an implicit fibre field, $d(\mathbf{x})$ [TSB+05]:

$$d(\mathbf{x}) = \frac{\dfrac{\partial V(V^{-1}(\mathbf{x}))}{\partial a}}{\left\| \dfrac{\partial V(V^{-1}(\mathbf{x}))}{\partial a} \right\|} \tag{4}$$

$$V(\mathbf{u}) = \sum_{i}^{p}\sum_{j}^{q}\sum_{k}^{r} R_{i,j,k}(\mathbf{u})\mathbf{c_{i,j,k}} \tag{5}$$

where $\mathbf{u}$ and $\mathbf{x}$ are the parametric and spatial coordinates respectively, $R_{i,j,k}(\mathbf{u})$ are the NURBS volume rational basis functions, $\mathbf{c_{i,j,k}}$ are the control points, p, q and r are the number of control points along u_1, u_2 and u_3 respectively, and $a \in \{u_1, u_2, u_3\}$. As the NURBS volume function requires parametric rather than spatial coordinates, the parametric coordinates of sample points, $V^{-1}(\mathbf{x})$, are estimated using the Newton-Raphson root finding method.

4.4 Computing Boundary Conditions

To define model boundary conditions, using our simulation system, it is possible to restrict the movement of particular nodes by setting them as rigid or sliding (bound by a surface). Rigid nodes remain fixed throughout a simulation, whereas sliding nodes remain a fixed distance from the non-conforming surface they are restricted by, and can be used to model, for example, the sliding effect between the superficial stiff deep facial soft-tissue layers [WMSH10] (a phenomenon often neglected with current physics-based facial animations [SNF05, BJTM08]).

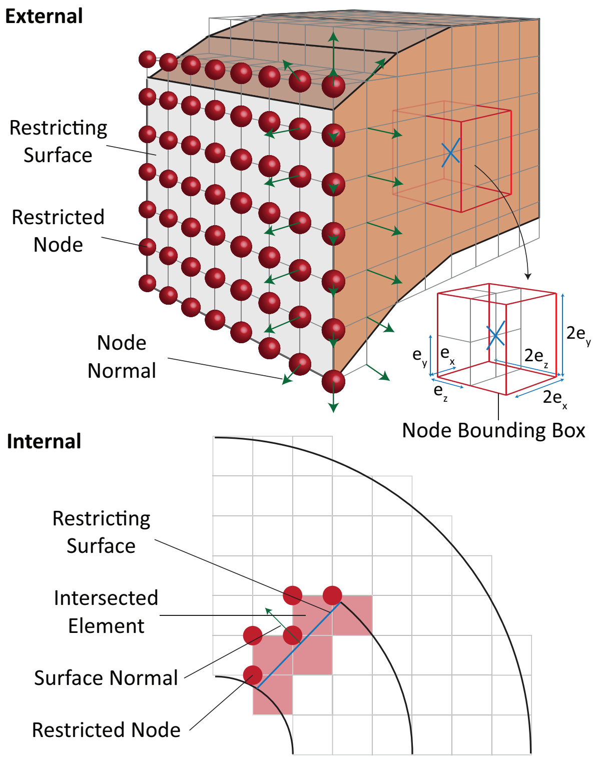

Figure 8: Identification of restricting nodes from external (top) and internal restricting surfaces (bottom, in 2D). In this case, the angle between the external surface and outer node normal must be less than $\pi/2$ for the node to be set as restricted.

As shown by Figure 8, restricted nodes are identified using a collection of non-conforming internal and external rigid and sliding surfaces. An internal surface might represent an attachment area inside the main volume (a level 0 volume), whereas the skull is an obvious external surface that defines part of the boundary of such a volume. With internal surfaces, elements that are intersected by the surfaces are firstly determined. For each node of each such element, if it is located in front of the surface (tested using the closest point on the surface and its normal), it is set as restricted (rigid or sliding).

With external surfaces, only the outermost nodes are considered. To handle cases where a surface lies outside elements without intersecting them, a bounding box with dimensions $2 \times e_i$ (where e_i is the voxel element dimension) is used around each such node centre, and the nodes are recorded if the surface intersects their bounding box. For each such node, if the angle between the node normal and the normal of the closest surface point is less than a specified size (i.e. they are pointing in a similar direction), it is set as restricted. This test is necessary to prevent false positives being detected, for example, if a node lies within range of the surface but is part of an adjoining surface approximation (see Fig-

Layer	Young's Modulus (MPa)	Poisson Ratio	Depth (mm)
Stratum Corneum	48	0.49	0.02
Dermis	0.0814	0.49	1.8
Hypodermis	0.034	0.49	Remains
Muscle	0.5	0.49	Variable
Tendon	24	0.49	Variable

Table 1: The neo-Hookean material properties used for the soft-tissue models.

ure 8). For each sliding node, the signed distance to the surface mesh is also calculated.

4.5 Binding the Surface Mesh, and Model Simulation

For the final stage of the model creation process, as with Dick et al.'s simulation approach [DGW11], the vertices of the surface meshes are bound to and animated with elements of the simulation mesh using trilinear interpolation and extrapolation. A position, $\mathbf{p}$, is bound to the closest element, e (determined by distance tests from the element centres), using three weights, $w_i^{(\mathbf{p})}$ - one along each local axis, i, from the first node of the element, $\mathbf{x}^{(\mathbf{e})}$. For an undeformed voxel element aligned with the global axes, these can be easily calculated:

$$w_i^{(\mathbf{p})} = 1 - \frac{p_i - x_i^{(e)}}{e_i} \qquad (6)$$

where e_i is the voxel dimension.

Models generated using our creation process can be simulated using an optimised GPU-based non-linear TLED FE solver [WM13], the full details of which are beyond the scope of this paper. For efficient simulation with this simulation system, the generated models and solver have been optimised to exploit the computational advantages of using such non-conforming hexahedral models on the GPU [WM12]. These optimisations enable both performance and memory advantages resulting from efficient element and node data organisation, facilitating GPU memory coalescing and global memory cache hits, and the reduced amount of element data that is stored, as all elements are initially the same size and shape. This has led to performance increases of almost 2x compared with using a conforming hexahedral simulation model.

5 RESULTS

Figures 9 and 10 show a forehead simulation model that has been generated using our creation process, and the surfaces that were used to generate this. It includes the frontalis, procerus and corrugator supercilli muscles. Figure 2 contains an example of a simpler soft-tissue-block model with a single muscle. As only part of the facial model has been created, nodes along the boundaries to the remainder of the face have been set as rigid to provide the anchoring effect that connecting elements would provide with a full model. While this could cause some artefacts at the model boundaries depending on the size of the external or propagated internal forces at these regions, none are visible in our examples.

Detail	Face	Skin Block	Armadillo
Nodes	629,178	146,410	19,698
Elements	503,530	129,600	15,107
Element Size (mm^3)	0.5^3	0.5^3	2.5^3
Voxel Samples	4^3	4^3	10^3
Model Generation (Single CPU Thread)			
Time (mins:secs)	6:00	0:23	0:41
Memory (MB)	3250	496	148
Simulation (GPU)			
Timestep (ms)	0.005	0.005	0.15
Timestep Computation Time (ms)	13.2	2.9	0.5

Table 2: Statistics of the examples, using an Intel i7-3930K CPU and an NVIDIA GTX 680 GPU.

Tables 1 and 2 show the material properties that were used, and some model and performance statistics. Such complex, high-resolution models are necessary to capture the thin structures, such as skin layers, and simulate fine wrinkling behaviour. As the deep layers are tough and fairly rigid, these are not modelled, and the superficial layers simply slide over the skull or bone surface.

The outer skin surface of the facial mesh was produced using FaceGen[2], while all other polygonal and NURBS surfaces were manually created based on anatomy using 3D modelling tools (see Section 6 for further discussion on surface mesh creation). Both models have constant soft-tissue thickness. As the frontalis has no skull attachment, the galea aponeurotica has been modelled on the forehead model to anchor this muscle and restrict soft-tissue movement towards the top of the head when it contracts. The region of overlap between these structures represents the smooth blend of fibres.

The examples demonstrate the complexity of models that can be created using our creation process, which include skin layers, anatomical fibre directions, and advanced boundary conditions. However, it can be seen that the epidermal layer (stratum corneum) has a higher stiffness and appears thicker than in reality. Due to the low thickness of this layer in relation to element size, the dermis dominates the outer layer of elements. When the material properties of these skin layers are combined for these elements, this results in a stiffness much

[2] `http://www.facegen.com/`

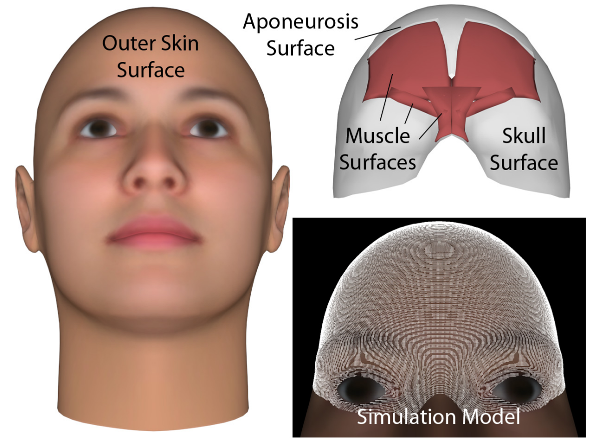

Figure 9: Surfaces and the simulation model for a forehead including the frontalis, procerus and corrugator supercilli muscles.

lower than that of the epidermis alone. The high stiffness is therefore necessary to produce a large enough difference between the stiffness of the two outer layers of elements, which is necessary to simulate wrinkles [FM08]. As a consequence, when using the same material models for the dermis and epidermis, the outer layer of elements acts like a thick epidermal layer. This is unavoidable without using different-shaped elements that can capture the thickness of the epidermal layer.

Regarding performance, the main bottleneck of the model generation was the computation of parametric coordinates of NURBS volumes when computing muscle fibre directions, which used an unoptimised Newton-Raphson root-finding algorithm, and contributed to roughly 38% of the computation time. Also, while memory requirements are quite high, small sections of larger models could be generated independently, as computations of properties for each element and node are independent.

Figures 2 and 11 show animation results using the described example models. More animation results with some parameter variations have been previously presented [WM13]. The elements are treated as 8-node reduced-integration hexahedral elements with hourglass control. Figure 12 shows results of a similar forehead animation using a model with the same number of nodes and elements, but created from a previous simpler version of the model generation process [WM12]. While this model took just 1 minute 20 seconds to generate, using 3.1 GB RAM, and each timestep 9.8ms to simulate (mainly due to no nodal sliding), no wrinkles were produced as only a single skin layer is modelled, and stretching effects can be seen along the lateral edges of the frontalis due to poor approximation of muscle fibre directions. Compression of muscles towards a single point also reduced simulation stability.

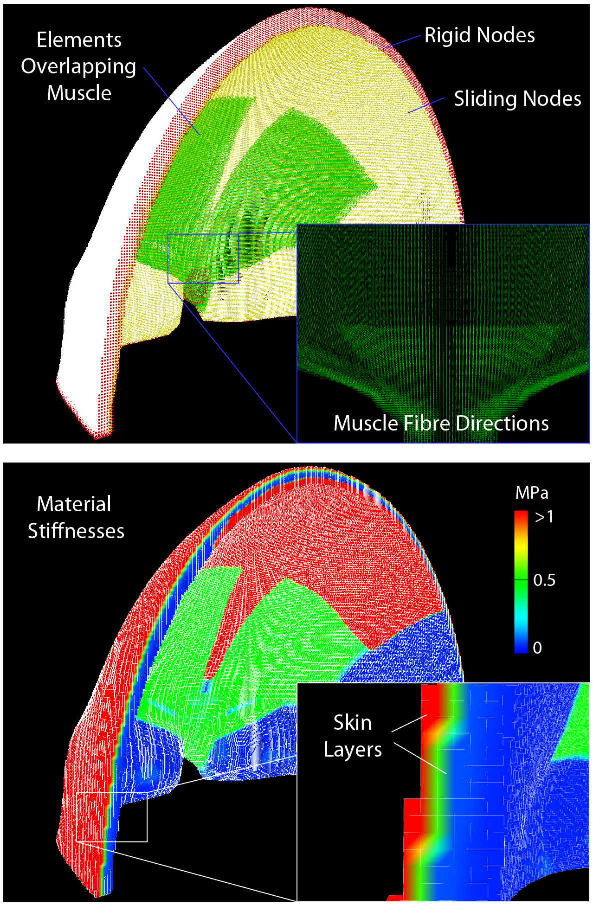

Figure 10: Rear views of the forehead simulation model.

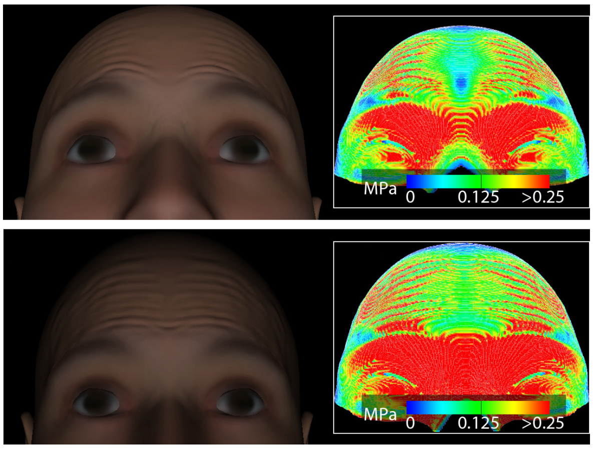

Figure 11: Animation results of forehead models under contraction of the frontalis. The top example uses the model in Figures 9 and 10, whereas the bottom example excludes the procerus and corrugator supercilli muscles, demonstrating the flexibility of the animations by using different models and parameters. Insets show the stresses.

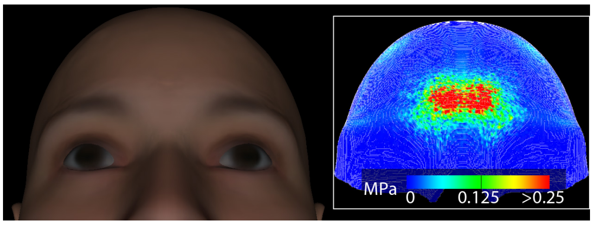

Figure 12: Animation results of a forehead model, created using a previous version of the model generation process [WM12], under contraction of the frontalis.

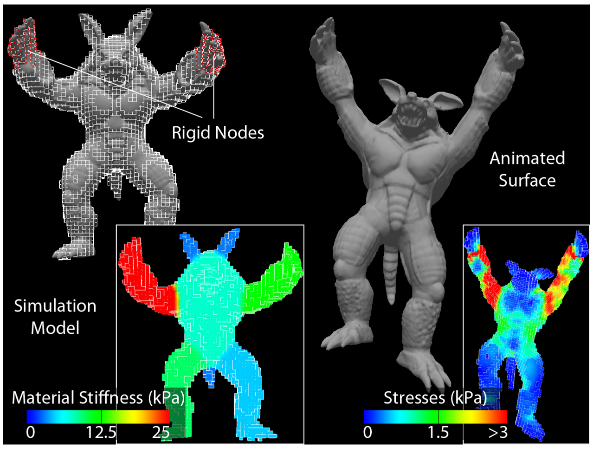

Figure 13: The simulation model and animation results under gravity of a multi-material Stanford Armadillo.

A comparison of using conforming and optimised non-conforming hexahedral models for GPU simulation has been previously discussed [WM12]. Finally, Figure 13 shows an example of a more generic multi-material object, demonstrating the flexibility of our model creation and animation approach. Fewer larger elements were used (but with a higher voxel sampling rate during creation) to animate the gross object movement.

6 DISCUSSION: SURFACE MESH CREATION

For a complex facial model, manually creating and tweaking all the surfaces can be a time-consuming task. This could be made easier, for example, by having an interactive editor for semi-automatic muscle surface creation to simplify muscle definition (like with previous approaches [KHYS02]), although this would probably impact on flexibility and user control. Alternatively, medical data (e.g. CT and MRI data) could be used to create more accurate surfaces. As the simulations are highly dependent on the simulation model and parameters, it is likely that much more realistic soft-tissue animations could be produced by using models generated from such surfaces.

To create a model using medical data, it would first be necessary to use such data to create the separate labelled polygonal surfaces for each structure (including the skull, skin surfaces, muscles and tendons), and the NURBS surface approximations of muscles. The surfaces would need to be hole free due to the sampling procedure that approximates the proportion of overlap between mesh volumes and voxels. However, as most parts of our model creation process, including the sampling procedure, are independent, they could be easily altered or replaced. A different overlap computation technique could therefore be used, which may be more robust to noisy data or volumes with holes, and require less surface mesh clean-up when working with medical data. Once a single reference surface mesh has been created, either manually or from medical data, existing approaches could be used to deform the surfaces of this, such as the muscles, tendons and skull, for easy creation of a new surface mesh for a different face (e.g. based on range scan data) [KHYS02, AZ10].

7 CONCLUSION

This work has presented an automatic process to easily create animatable multi-layered non-conforming hexahedral FE simulation models to which surface meshes are bound, focussing on facial soft-tissue models. Starting with any closed surface mesh, this involves discretising the enclosed volumes into voxels, and calculating model properties (such as skin layers, and element material and muscle properties) based on the proportion of overlap between the volumes and voxels. Boundary conditions are also computed, enabling nodes to be set as rigid (fixed) or sliding (bound by a surface) based on a collection of non-conforming surfaces. The models are optimised with resourceful data storage and organisation for efficient GPU simulation. Examples have demonstrated the complexity and flexibility of models that can be created, and their ability to produce animation of realistic large and fine-scale soft-tissue behaviour.

However, various improvements could be made to the model creation process. Multi-resolution models could be created to enable a higher resolution to be used where necessary, such as along the outer skin surface where wrinkles are produced. Future work will focus on attaching shell elements to the outer skin surfaces to more accurately model the thickness of the outer epidermal layer, and modifying the surface meshes to generate more accurate models. These models will be used with anisotropic viscoelastic materials to produce more realistic animations of different-aged facial movement.

8 REFERENCES

[AZ10] O. O. Aina and J. J. Zhang. Automatic Muscle Generation for Physically-Based Facial Animation. In *SIGGRAPH Posters*, pages 105:1–105:1, 2010.

[Bis06] J. E. Bischoff. Reduced Parameter Formulation for Incorporating Fiber Level Viscoelasticity into Tissue Level Biomechanical Models. *Ann. Biomed. Eng.*, 34(7):1164–1172, 2006.

[BJTM08] G. Barbarino, M. Jabareen, J. Trzewik, and E. Mazza. Physically Based Finite Element Model of the Face. In *Proc. ISBMS*, pages 1–10, 2008.

[CPL00] B. Couteau, Y. Payan, and S. Lavallée. The mesh-matching algorithm: an automatic 3D mesh generator for finite element structures. *J. Biomech.*, 33(8):1005–1009, 2000.

[DGW11] C. Dick, J. Georgii, and R. Westermann. A real-time multigrid finite hexahedra method for elasticity simulation using CUDA. *Simul. Model. Pract. Theory*, 19(2):801–816, 2011.

[FM08] C. Flynn and B. A. O. McCormack. Finite element modelling of forearm skin wrinkling. *Skin Res. Technol.*, 14(3):261–269, 2008.

[Fra12] M. Fratarcangeli. Position-based facial animation synthesis. *Comput. Animat. Virtual Worlds*, 23(3-4):457–466, 2012.

[ISS09] Y. Ito, A. M. Shih, and B. K. Soni. Octree-based reasonable-quality hexahedral mesh generation using a new set of refinement templates. *Int. J. Numer. Methods Eng.*, 77(13):1809–1833, 2009.

[KHYS02] K. Kähler, J. Haber, H. Yamauchi, and H.-P. Seidel. Head shop: Generating animated head models with anatomical structure. In *Proc. SCA*, pages 55–63, 2002.

[KPB08] A. V. Kumar, S. Padmanabhan, and R. Burla. Implicit boundary method for finite element analysis using non-conforming mesh or grid. *Int. J. Numer. Methods Eng.*, 74(9):1421–1447, 2008.

[KRG$^+$02] R. M. Koch, S. H. M. Roth, M. H. Gross, A. P. Zimmermann, and H. F. Sailer. A Framework for Facial Surgery Simulation. In *Proc. SCCG*, pages 33–42, 2002.

[KSY08] O. Kuwazuru, J. Saothong, and N. Yoshikawa. Mechanical approach to aging and wrinkling of human facial skin based on the multistage buckling theory. *Med. Eng. & Phys.*, 30(4):516–522, 2008.

[Lam09] B. P. Lamichhane. Mortar Finite Elements for Coupling Compressible and Nearly Incompressible Materials in Elasticity. *Int. J. Num. Anal. Model.*, 6(2):177–192, 2009.

[LLT11] M.-F. Li, S.-H. Liao, and R.-F. Tong. Facial hexahedral mesh transferring by volumetric mapping based on harmonic fields. *Comput. Graph.*, 35(1):92–98, 2011.

[Lo12] S. H. Lo. Automatic merging of hexahedral meshes. *Finite Elem. Anal. Des.*, 55:7–22, 2012.

[LST09] S.-H. Lee, E. Sifakis, and D. Terzopoulos. Comprehensive Biomechanical Modeling and Simulation of the Upper Body. *ACM Trans. Graph.*, 28(4):99:1–99:17, 2009.

[MBTF03] N. Molino, R. Bridson, J. Teran, and R. Fedkiw. A Crystalline, Red Green Strategy for Meshing Highly Deformable Objects with Tetrahedra. In *Proc. IMR12*, pages 103–114, 2003.

[MHSH10] K. Mithraratne, A. Hung, M. Sagar, and P. J. Hunter. An Efficient Heterogeneous Continuum Model to Simulate Active Contraction of Facial Soft Tissue Structures. In *Proc. WCB*, pages 1024–1027, 2010.

[MLC01] D. Ma, F. Lin, and C. K. Chua. Rapid Prototyping Applications in Medicine. Part 1: NURBS-Based Volume Modelling. *Int. J. Adv. Manuf. Technol.*, 18(2):103–117, 2001.

[NRP11] M. Nieser, U. Reitebuch, and K. Polthier. CUBECOVER – Parameterization of 3D Volumes. *Comp. Graph. Forum*, 30(5):1397–1406, 2011.

[RP07] O. Röhrle and A. J. Pullan. Three-dimensional finite element modelling of muscle forces during mastication. *J. Biomech.*, 40(15):3363–3372, 2007.

[SG05] H. Si and K. Gärtner. Meshing Piecewise Linear Complexes by Constrained Delaunay Tetrahedralizations. In *Proc. IMR14*, pages 147–163, 2005.

[SKO$^+$10] M. L. Staten, R. A. Kerr, S. J. Owen, T. D. Blacker, M. Stupazzini, and K. Shimada. Unconstrained plastering – Hexahedral mesh generation via advancing-front geometry decomposition. *Int. J. Numer. Methods Eng.*, 81(2):135–171, 2010.

[SNF05] E. Sifakis, I. Neverov, and R. Fedkiw. Automatic Determination of Facial Muscle Activations from Sparse Motion Capture Marker Data. *ACM Trans. Graph.*, 24(3):417–425, 2005.

[SSLS10] M. L. Staten, J. F. Shepherd, F. Ledoux, and K. Shimada. Hexahedral Mesh Matching: Converting non-conforming hexahedral-to-hexahedral interfaces into conforming interfaces. *Int. J. Numer. Methods Eng.*, 82(12):1475–1509, 2010.

[SZM12] L. Sun, G. Zhao, and X. Ma. Quality improvement methods for hexahedral element meshes adaptively generated using grid-based algorithm. *Int. J. Numer. Methods Eng.*, 89(6):726–761, 2012.

[TCO08] Z. A. Taylor, M. Cheng, and S. Ourselin. High-Speed Nonlinear Finite Element Analysis for Surgical Simulation Using Graphics Processing Units. *IEEE Trans. Med. Imaging*, 27(5):650–663, 2008.

[TSB$^+$05] J. Teran, E. Sifakis, S. S. Blemker, V. Ng-Thow-Hing, C. Lau, and R. Fedkiw. Creating and Simulating Skeletal Muscle from the Visible Human Data Set. *IEEE Trans. Vis. Comput. Graph*, 11(3):317–328, 2005.

[TW90] D. Terzopoulos and K. Waters. Physically-Based Facial Modeling, Analysis, and Animation. *J. Vis. Comput. Animat.*, 1(2):73–80, 1990.

[TZT09] C. Y. Tang, G. Zhang, and C. P. Tsui. A 3D skeletal muscle model coupled with active contraction of muscle fibres and hyperelastic behaviour. *J. Biomech.*, 42:865–872, 2009.

[Wat87] K. Waters. A muscle model for animation three-dimensional facial expression. In *Proc. SIGGRAPH*, pages 17–24, 1987.

[WJC$^+$10] A. Wittek, G. Joldes, M. Couton, S. K. Warfield, and K. Miller. Patient-specific non-linear finite element modelling for predicting soft organ deformation in real-time; Application to non-rigid neuroimage registration. *Prog. Biophys. Mol. Biol.*, 103(2–3):292–303, 2010.

[WM12] M. Warburton and S. Maddock. Creating Animatable Non-Conforming Hexahedral Finite Element Facial Soft-Tissue Models for GPU Simulation. In *Proc. WSCG*, pages 317–325, 2012.

[WM13] M. Warburton and S. Maddock. Physically-Based Forehead Animation including Wrinkles. In *Proc. CASA*, 2013.

[WMSH10] T. Wu, K. Mithraratne, M. Sagar, and P. J. Hunter. Characterizing Facial Tissue Sliding Using Ultrasonography. In *Proc. WCB*, pages 1566–1569, 2010.

[XLZH11] S. Xu, X. P. Liu, H. Zhang, and L. Hu. A Nonlinear Viscoelastic Tensor-Mass Visual Model for Surgery Simulation. *IEEE Trans. Instrum. Meas.*, 60(1):14–20, 2011.

[ZHB10] Y. Zhang, T. J. R. Hughes, and C. L. Bajaj. An Automatic 3D Mesh Generation Method for Domains with Multiple Materials. *Comput. Methods Appl. Mech. Eng.*, 199(5–8):405–415, 2010.

[ZHD06] S. Zachow, H.-C. Hege, and P. Deuflhard. Computer-Assisted Planning in Cranio-Maxillofacial Surgery. *J. Comp. Inf. Technol.*, 14(1):53–64, 2006.