Vis Comput (2008) 24: 911–922
DOI 10.1007/s00371-008-0287-z

ORIGINAL ARTICLE

Haptic two-finger contact with textiles

Guido Böttcher · Dennis Allerkamp · Daniel Glöckner · Franz-Erich Wolter

Published online: 7 August 2008

Abstract Real-time cloth simulation involves many computational challenges to be solved, particularly in the context of haptic applications, where high frame rates are necessary for obtaining a satisfying experience. In this paper, we present an interactive cloth simulation system that offers a compromise between a realistic physics-based simulation of fabrics and a haptic application meeting high requirements in terms of computation speed. Our system allows the user to interact with the fabric using two fingers. The required performance of the system is achieved by introducing an intermediate layer responsible for the simulation of the small part of the surface being in contact with the fingers. Additionally we separate the possible contact situations into different cases, each being individually handled by a specialised contact algorithm.

Keywords Virtual reality · Haptics · Deformable objects · Tactile rendering

G. Böttcher (✉) · D. Allerkamp · D. Glöckner · F.-E. Wolter
Welfenlab—Computer Graphics Division, Institute of Man–Machine Communication, Leibniz Universität Hannover, Hannover, Germany
e-mail: boettcher@welfenlab.de

D. Allerkamp
e-mail: allerkam@welfenlab.de

D. Glöckner
e-mail: glockner@welfenlab.de

F.-E. Wolter
e-mail: few@welfenlab.de

1 Introduction

Virtual Reality has a lot of applications ranging from entertainment to mechanical design and medical training. With the appearance of virtual worlds where users can interact via avatars over the Internet (e.g. Second Life), companies started to promote their products as virtual artifacts—another emerging application of Virtual Reality. Virtual Reality systems can be categorised by the modalities they support. In today's systems the modalities of seeing and hearing are the most commonly employed ones as these are also the modalities in which we as human beings mostly exchange information. They require little effort in terms of energy transfer, the corresponding sensory receptors are concentrated in the retina and the cochlea and can be excited remotely with light and sound waves, respectively.

In contrast to seeing and hearing the creation of appropriate haptic stimuli demands very sophisticated hardware. Firstly, the skin with its size of 1.5 to 2 square metres is a very large organ. Therefore, most haptic devices focus on a rather small part of the human body—usually the fingertip. Secondly, forces cannot be transmitted contact-free with current technology. Thus haptic devices always need direct contact with the parts of the skin where the forces are applied. Thirdly, the amount of energy being transferred is relatively high compared to other modalities; e.g., if one wants to simulate the lifting of an object with a mass of 500 g the haptic device has to create a force of approximately 5 N. All these properties make haptic simulation a complex task still presenting a lot of problems awaiting a good solution. But these efforts should result in a richer, more convincing virtual reality making applications like the promotion of textiles via the Internet easier.

Although it is sufficient to simulate many objects as rigid, a lot of applications demand the simulation of deformable objects, e.g. organs for medical training, elastic parts like flexible tubes for mechanical design, or cloth for entertainment. An accurate physical simulation usually employs complex models that have to be solved numerically, whereas the real-time demands of Virtual Reality—especially haptics—require high update rates. A trade-off between these two requirements is offered in this paper for the modelling of fabrics. But the approach used here may be generalised to a larger class of deformable objects.

The work presented here is part of the EU project HAPTEX [13]. Its main goal is to develop a virtual reality system for visuo-haptic interaction with virtual textiles. In addition to the haptic simulation, our VR system simulates fabrics' surface properties by tactile stimulation which is also outlined here. For a more detailed description of the so-called tactile rendering, see [1, 18]. The integration of force feedback and tactile rendering into a single system is described in [11].

2 State of the art

While graphical rendering has become quite sophisticated giving the user a view of objects nearly indistinguishable from reality, the haptic rendering still faces a lot of problems [15]. Most of these problems arise due to the bidirectional interaction with the user and his requirements in having a realistic impression of touching virtual objects. The two main omnipresent problems in haptics are the stability of the force feedback and the short response time of 1 ms. The latter one is easy to solve by using fast and simple algorithms [20, 26], but these are unsuited as to present a realistic simulation of deformable objects to the user. In recent years, many solutions have been proposed to solve the problem of response time, i.e. by pre-computing the forces under deformation [16]. But a more widely used approach especially for deformable objects was first introduced in [2]. The author proposes to use a local buffer model to generate the force outputs for haptic feedback. The local model reproduces the behaviour of the object under local deformation within the contact area. Furthermore, different models have been presented to simulate deformable objects for haptics, i.e. linear FEM [7], different meshes for local and global [25] or pre-computed force functions [14]. The most sophisticated approach in haptic rendering has been proposed by [9]. The method is based on the Signorini contact problem commonly used in contact mechanics to model objects in contact. As a result of the comprehensive treatment of the contact it requires much computation time and cannot satisfy full haptic real-time requirements. Besides the response time the other aforementioned issue in haptics is the stability of the rendering which heavily depends on the device and on the rendering itself. Analysis of the stability has contributed to the multirate rendering algorithms guaranteeing the stability between local and global model running at different update rates [4, 8].

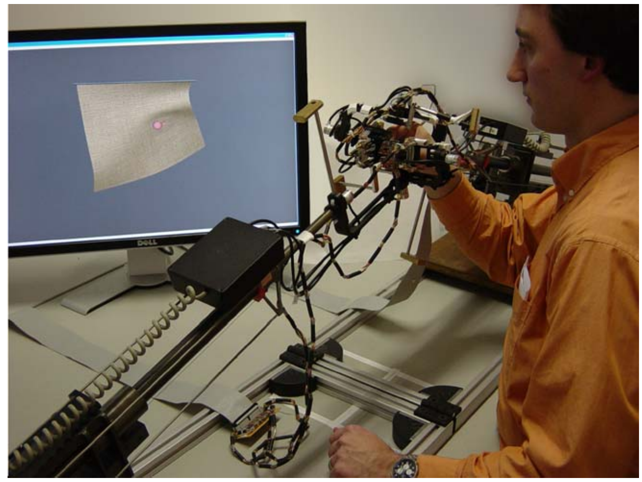

Fig. 1 The final demonstrator of the HAPTEX project

3 VR-system architecture

The goal of the HAPTEX system is to feature virtual fabrics that resemble their real counterparts as much as possible. Therefore a set of real fabrics has been selected and their physical properties have been measured with the Kawabata System (see [22]).

The simulated fabrics are square shaped with a side length of 20 cm. The user can select a fabric from the property database which is then simulated hanging from a stand (see Fig. 1). The user can touch the virtual fabric with his thumb and index finger. The fabric can be squeezed, stretched, rubbed and lifted.

During the interaction with the fabric the position of the user's fingers and the shape of the fabric change. Both changes are considered by the contact model that computes the forces affecting the finger and the fabric. The aforementioned interaction is called the haptic loop. Here a problem becomes evident: to create a convincing illusion and to avoid stability problems the system needs to react to the user's motion within one millisecond, which is not possible involving the slower simulation of the fabric. However, note that the haptic interaction loop can also be seen as two loops: the one between the user and the contact model, and the other between the simulation and the contact model. This view led to the solution described in Sect. 4, which employs a dual-layer approach to allow the two loops to run at different speeds.

In the following sections we describe the hardware that allows the user to interact with the system (Sect. 3.1) and the simulation of the fabric (Sect. 3.2). Later on, in Sect. 5, several aspects of the contact are treated. There we describe the solution chosen for the HAPTEX project.

3.1 System hardware

The GRAB device is a force-feedback device consisting of two identical and independent robotic manipulators (see Fig. 1), each having the base link fixed to the desktop and the end-effector (contact part) attached to the palmar surface of the user's thumb or index fingertips. Each manipulator measures the absolute position and orientation of the contact part. These are used by the haptic renderer to compute an appropriate force. The manipulator is able to generate this force on the contact part within a workspace of 400 mm in width, 300 mm in height and 200 mm in depth. The workspaces of both manipulators overlap, so that only 300 mm width is reachable with both fingers. The resolution of the position sensors is 0.032 mm in axial direction and 1.26×10^{-4} radians in other directions. In the region covered by the textile this results in a worst-case resolution of 0.11 mm.

Force errors are limited in a range of about $+/-10$ g (0.1 N). The device can exert peak forces up to 20 N while continuous force output is possible up to 6 N (in the centre of the workspace). The device was initially developed in another project to provide blind people's access to virtual three-dimensional scenes. For this project both manipulators have been extended with a gimbal and a three-dimensional force sensor. The gimbal is responsible for allowing and measuring the rotation of the user's finger. These are passive elements, so no torque can be applied to the finger. The force sensor has been placed directly at the end-effector and has been developed especially for the space requirements in this project. The control loop tries to match the measured force to the desired force and can thus achieve smaller force errors than without this feedback.

The control loop for the force-feedback device runs on a separate computer at a rate of about 5 kHz. A standard Ethernet connection provides the link to the main computer. Given a low network latency, the UDP based protocol could in theory allow remote application usage. The modification to the GRAB and a detailed analysis of the behaviour and the control loop is presented in [5].

For the tactile simulation of surface texture, 24 vibrating actuator pins are arranged in 6×4 grid with a spacing of 2 mm. These pins are attached to the force plate such that the mechanoreceptors in the ventral part of the thumb or index finger can be excited by the vibrations. For each contactor pin the amplitude of the vibration at 40 and 320 Hz can be controlled separately. For the technical details of the tactile display we refer to [1] and [11].

3.2 Textile simulation method

The simulation of the fabric has to ensure that the mechanics are modelled appropriately while keeping the computational costs low to reach real-time requirements. For the simulation of large-scale deformations it is necessary to use a nonlinear model to reflect the behaviour of textiles correctly. At the global level we therefore use the textile simulation library from [23], which uses spline curves to reproduce this nonlinear strain–stress relationship according to real measurements.

We analysed different approaches [3, 10, 12, 23] to realise a local model, but finally stuck to using the same library as for the large-scale model with adapted parameters for achieving higher speeds. This approach ensures good transitions between the models while having sufficient flexibility in contact formulations due to its moderate computational demands. Therefore, we are able to increase the complexity of our contact model ensuring the precision we need to drive the haptic interfaces.

3.2.1 Local mesh topology

The textile is discretised using particles that incorporate the physical properties of the surface. The surface topology itself is stored in a triangle surface defining relationships between the particles. Calculating contact forces involves algorithms that need to refer from one element of the mesh to adjacent elements. The computation of these parameters without efficient data structure for the different kinds of geometric queries can be costly. Thus, we decided to use a half-edge data structure [6] which is well suited for such queries.

3.2.2 Force computation

In the particle system the occurring forces in the deformation of the textile in each triangle are evaluated by looking at the change of the unit warp and weft vectors. Afterwards, the forces are integrated over the triangle and distributed among the particles.

Stretch/tensile force The stretch forces are measured by the elongation in warp and weft directions. For the simulation of these forces we have to know the elongation of the unit warp and weft vectors, which define the rest state, at each point. Assuming the stretch is constant over a triangle leads to a single computation for warp and weft directions.

Shear forces Deformations resulting in a change from the initial orthogonality between warp and weft vectors to a different angle create shear forces. These forces measured by the Kawabata System are determined by the inner angle between warp and weft directions. Therefore, the scalar product of the warp and weft vectors is computed and used as an estimate for the shear strain.

Bend forces Forces generated by bending are important for folding behaviour. The simulation library allows to create elements that calculate forces as weighted sums of particle positions. These are used to implement the bending model described in [24].

Damping forces All aforementioned internal forces result from the absolute position of the textile. But it is also important to incorporate forces which relate to the movement. These damping forces have to be considered not only for modelling the energy dissipation during the deformation, but also for the stability of the simulation which would otherwise cause the simulation to oscillate. The damping forces are counteracting to stretch, shear and bend motion. As each case is independent, the impact of damping is computed separately. Additional external viscous and higher order velocity-dependent damping forces are introduced to further stabilise the calculations and simulate an equivalent of air drag.

4 Dual-layer approach

Achieving a convincing virtual textile simulation requires a good compromise between the need for accuracy in the material representation and the need for speed for obtaining simulation frame rates compatible with real-time perception. These factors have to be considered both in the visual and the haptic fields. However, the graphics rendering loop has different requirements compared to the haptic rendering loop in terms of refresh frequencies. While in graphics a refresh rate of 30 Hz is quite acceptable, in haptics depending on the stiffness of the simulated object a response frequency of at least 500–1000 Hz is needed to ensure stable interaction. A dedicated structure has therefore been defined for adapting the different frame rates required by the mechanical simulation and the haptic rendering computations. Hence, two separate computation threads were implemented: The first is a low-frequency thread for dealing with the complex large-scale simulation of the whole cloth surface. This is an accurate particle system representation integrated with state-of-the-art numerical methods for achieving quantitative accuracy of the nonlinear anisotropic behaviour of cloth in real-time. The second is a high-frequency thread for computing the local data necessary for haptic rendering and for accurately sending haptic forces back to mechanical simulation.

The force-feedback thread is created by the driver of the GRAB device. A high resolution timer guarantees a reliable frame rate of 1 kHz. The connection between the haptic renderer and the driver is formed by a call-back, which receives the position and orientation of the contact parts of both robotic manipulators and sends appropriate forces back to the driver.

Maintaining the update rate in the high-frequency thread requires that we restrict ourselves to a small section of the surface for physically accurate interactions. The motion area is well defined by the position of the cursor and its limited velocity. Hence, we can define a proximity region given by a bounding sphere restricting the volume where the cursor and the surface may possibly have contact. By narrowing our local consideration and calculations to be performed in the haptic loop to the parts of the surface in the bounding sphere allows us to reduce our computation effort to a minimum. This so-called local geometry is used for our simulation in the high-frequency thread.

Due to the small deformations occurring in a time frame of milliseconds, simpler algorithms can be used to model the local geometry without losing much accuracy compared to the physically precise mechanical model.

To prevent the local geometry, i.e. the small section of the surface in contact with the haptic cursor, from diverging too far from the main mechanical simulation, we assume constant position at its border. This is achieved by disregarding internal and external forces on the border in the local simulation. External forces, including the forces on the local border, are accumulated and later on applied to the global model.

The flow of data within the haptic rendering is depicted in Fig. 2. The architecture of the renderer is conceived to functionally separate the stages in the haptic interaction. Apart from performance gains on multi-core systems the modular design allows to work independently on different parts relevant for the complete integration of all hard- and soft-ware components provided by the partners of HAPTEX.

Figure 3 depicts the chronological order of events in the communication between the components. In the initial stage, all threads are running at their dedicated update rate. The force-feedback thread is constantly fetching new positions from the force-feedback device. These positions are processed to predict the user's motion and to estimate the next position. At the same time, the (global) textile simulation thread is computing the deformations of the global model caused only by gravity, while the local simulation thread awaits new local geometries to simulate.

At each simulation step of the global thread, it receives fingertip dimensions, the current position and the predicted position. The global thread analyses its underlying mesh for potential collisions with the fingertip for the next time step. These parts are sent to the local thread to be refined and introduced into the local simulation. Afterwards, both simulation threads continue to run according to their data.

With the newly added local mesh, the local thread checks if a collision has taken place in-between the last two local simulation time steps. In case of a contact the occurring

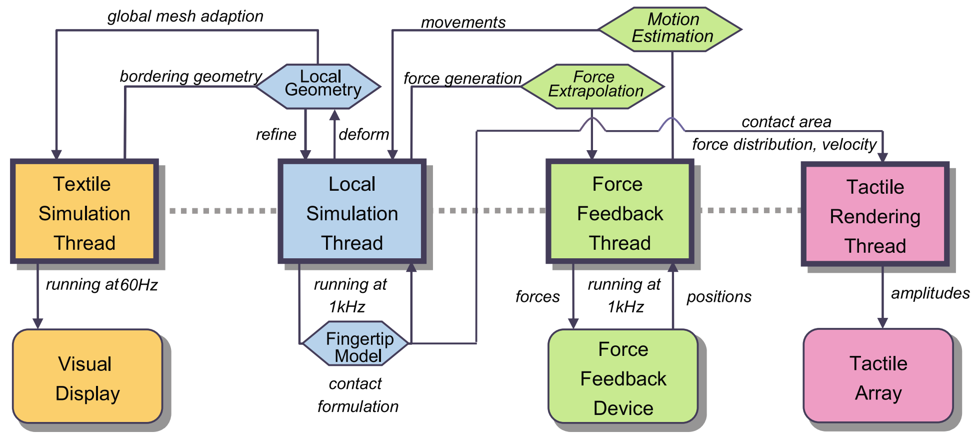

Fig. 2 Threads of the HAPTEX system

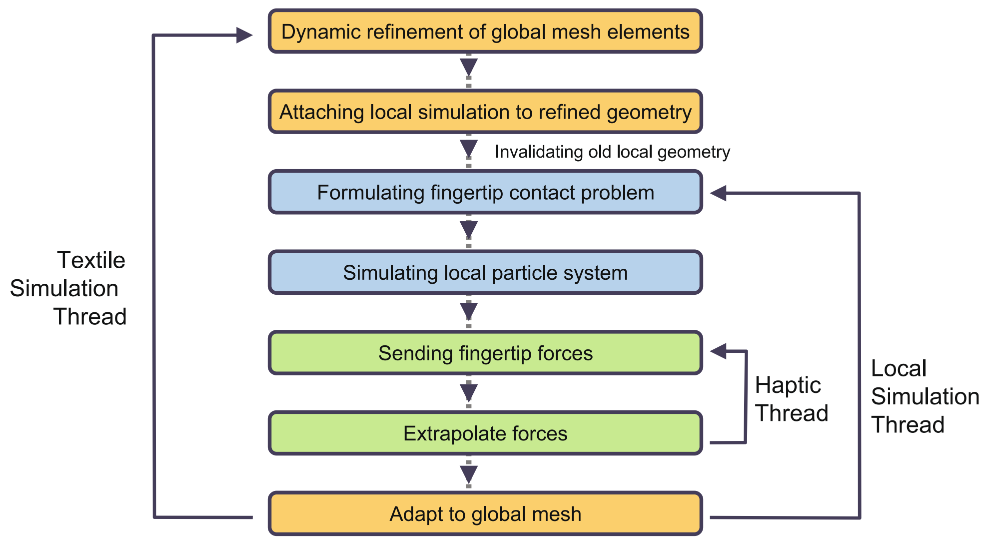

Fig. 3 Sequential order of communication

deformation of the local part of the textile is computed according to the fingertip model being used. The forces at the fingertip generated during the contact are sent to the force-feedback thread. When the next global time step is reached the changed geometry of the local mesh and the present forces in the local simulation are transferred to the global model.

Table 1 Possible contact states

Fingers touch	No		Yes	
Friction	Static	Dynamic	Static	Dynamic
No textile contact	I		II	III
One contact point	IV	V	–	–
Two contact points	VI	VII	VIII	IX

5 Contact rendering

It is certainly desirable to express all possible contact states within a single contact model. However, real-time demands and stability issues dictate the use of several contact models specialised in different contact states.

Table 1 lists the possible contact states. The fabric is in contact either with zero, one or two fingers. The model of the system also depends on whether the fingers touch each other or not. Note that "touch" is also used in this context if there is a fabric between the fingers, i.e. the fabric is squeezed by the two fingers. The third property determining the contact state is the kind of friction to be used, i.e. whether friction is static or dynamic.

When there is no contact at all, i.e. the fingers neither touch each other nor touch the fabric, friction certainly does not occur. Therefore the contact state is independent of the kind of friction in this case.

In the HAPTEX system the index finger is always assumed to be in contact with the back side of the fabric only when the thumb's contact is restricted to the front side. Furthermore, the mechanical setup restricts the contact between the fingers to the palmar part of the fingertip.

The different contact states shown in Table 1 can be expressed with the models which are described in the following sections. Later on, in Sect. 5.5, the tactile rendering of surface properties is outlined.

5.1 Contact state I: zero force

In this contact state the fingers neither touch each other nor touch the fabric. As a consequence, no contact forces occur. The implementation of this contact model is very simple as it always returns a contact force of 0 N.

5.2 Contact states II and III

These states define the rendering of the contact force of the two fingers when no textile is in-between. The Coulomb friction model is chosen to determine the (friction) states of the haptic response when the fingers are moved against each other. By observing the movement of the index finger in the reference frame of the thumb, we can apply the proxy model [26] (the concept of which is explained in more detail in Sect. 5.3) since we have only one finger moving whereas the other is fixed. Note that it makes no difference which finger is chosen as frame of reference because the force will act on both fingers and change its sign only. In case of an intersection of the two fingers the proxy is placed on the thumb surface in relation to the touching point of the fingers. The penetration depth d is used for computing the normal force. In [17] an exponential function based on the instantaneous elastic response of fingertip tissue was suggested. It was evaluated to describe the normal force F^N, but caused instabilities in our system. On the other hand, the stiffness of the finger is already included in the system by the real fingers being inside the thimbles. Therefore, an infinite stiffness would be needed at this point. As a compromise between stability and realism, we used a linear spring with an experimentally determined stiffness of 1.25 N/mm.

We also need to address the tangential friction due to the movement of the fingers against each other. A potential next proxy position is given by the device at a new time step and the aforementioned positioning rule. Based on the distance Δx between both proxy positions, the tangential force F^T is estimated by a spring with an artificial stiffness parameter. Together with the obtained normal force and the static friction coefficient μ_s, the stick–slip condition

$$F^T \leq \mu_s F^N$$

is then checked to determine the proxy movement. If the condition holds, the proxy remains on the actual position and the combined forces F^N, F^T are sent to both fingers. Otherwise the proxy is moved towards the next proxy position to fulfil the condition and the tangential force is recomputed according to the dynamic friction.

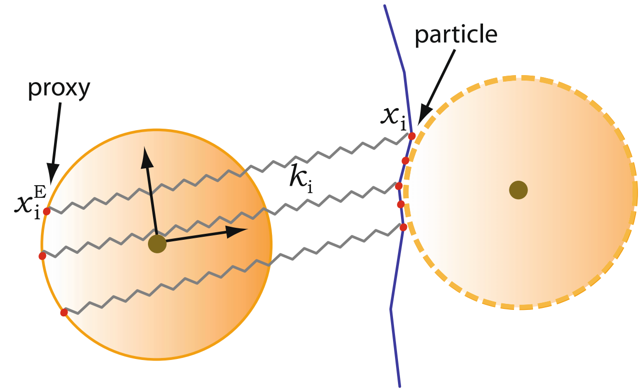

Fig. 4 Correspondence of mesh particles to the inverted proxies

5.3 Contact states IV, V, VI and VII: V-proxy

For dealing with contact situations in haptic interactions, an abstract object called *God-object*, or *Proxy*, was introduced by [26]. It is a point-like object unable to penetrate other objects. It therefore follows, in contrast to the position of the force-feedback device, the physical laws defined in our virtual environment. Thus the proxy follows the force-feedback device, as long as the user's movement does not intersect an object. However, in the contact states we address here, the finger touches the fabric whereas the proxy is set on the object's surface and a virtual spring is placed between the force-feedback device's position and the proxy. By moving the proxy on the surface to the point with the shortest distance from the device's position the spring's potential energy will be minimised. An extension of the algorithm was made in [20] and [19] to support contact rendering with arbitrarily shaped proxies. Obstacles and collisions are observed within a so-called configuration space. The configuration space is defined as the space of possible positions the proxy may attain, possibly subject to external constraints. Instead of looking for a valid position on the obstacle, a position on the border of the configuration space is chosen.

Our model additionally uses an inverted proxy model, where each mesh particle has a proxy that cannot penetrate the finger and thus sticks to its surface. At first a proxy for the finger is calculated by the common model that is then used to find the particles that should be forced onto the finger. Springs are used to connect these particles x_i to their corresponding point x_i^E on the finger surface (see Fig. 4). The force resultant of the force-feedback device is computed by

$$F^d = \sum_i k_i \left(x_i - x_i^E\right).$$

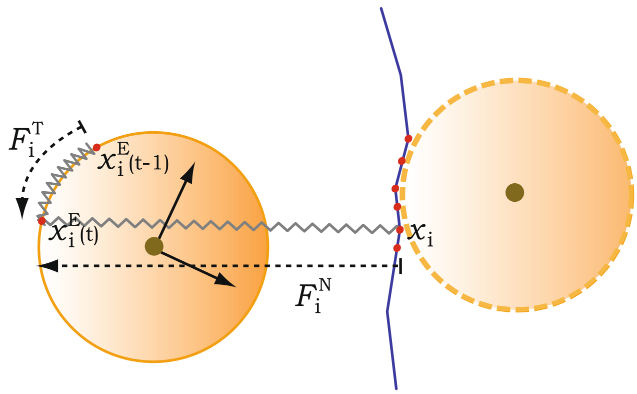

Fig. 5 Computation of force components for each proxy

For friction modelling, the movement of the spring ends on the finger is controlled as follows. To distinguish between static and kinetic friction, we take the spring length as the normal force F_i^N and place a curved spring between the previous end and the new desired end on the surface of the finger to model the tangential force F_i^T. If the forces satisfy the stick condition equation as seen below, we use static friction for this mesh particle:

$$F_i^T < \mu_S F_i^N.$$

If this condition holds, the spring end is constrained to the old position. Otherwise we need to satisfy the equation for dynamic friction by placing the end x_i^E at the correct position between the old and the desired position on the curved surface (see Fig. 5).

5.4 Contact states VIII and IX: two-finger model

These contact states deal with the two-finger textile contact. In the previous section we have applied a finger contact model based on the virtual proxy model, but for this specific case of having the textile between the fingers the proxy model is unsuitable. This is mainly due to the algorithm finding the local minimum of the distance between proxy and the haptic device position. If we have a situation of the proxies lying on opposite sides of the hanging textile, a slight movement in a direction normal to the textile can cause the proxies to move around each other. Experiments showed that this unwanted movement is a result of the deformation caused by the corresponding proxy in the movement direction. The pushing finger will increase the curvature at the contact point and produce a focal point near the surface leading to an unstable solution in the minimal distance search for the other proxy.

For that reason we created a more stable approach enforcing the fingertips to remain at the contact in the following way: Firstly, reaching the state of the two-finger model

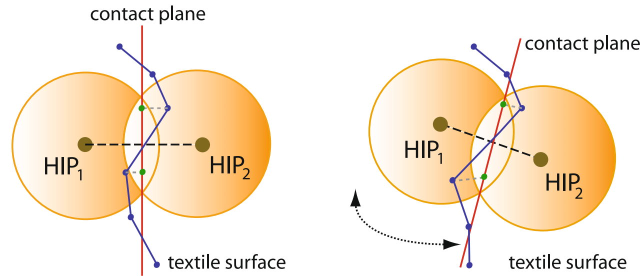

Fig. 6 Handling of textile nodes with two-finger contact

requires one or a set of collision points on the textile which are common to both fingers. These are equally distributed points on the part of the textile that should be in the intersecting area of the fingers. Their number is proportional to the intersecting area and larger than the number of particles in the area to achieve smoother friction. These points are treated solely by this model and are disregarded by the previous models. We assume the contact surface of both fingers to be planar with respect to the finger deformation. This induces a shared frame of reference for a 3D coordinate system with the line connecting the centres of the two fingers and its perpendicular bisectors as axes. According to the orientation of the fingers the colliding points are projected onto the plane as illustrated in Fig. 6. The desired position of the textile points results in this projection for every update of the haptic device. In the initial state we assume the points to be in static contact. Consequently, we keep the initial positions in the local frame of reference to define the new positions upon the haptic update. As in Sect. 5.3, the distance between ideal and actual positions of each textile point is related to the force applied using a linear spring. By doing so, we force each vertex of the textile mesh inside the two-finger collision area to move towards the contact plane defined by the current finger positions.

Friction is calculated similarly to the single-finger contact by using a model where we change the ideal position of a point according to the equation of its friction state. Points outside the intersecting area of the fingers are thought to cause negligible friction and are simply forced outside the finger. This special treatment of particles outside the intersection makes it possible to fold the textile over one finger while having it tightly grasped with both fingers, and guarantees a smooth transition to single-finger contact.

5.5 Tactile rendering

The tactile renderer needs two ingredients to work properly: a computer representation of a fabric and the trajectory of the fingertip on the fabrics surface. Figure 7 gives an overview of the tactile rendering. Before any rendering

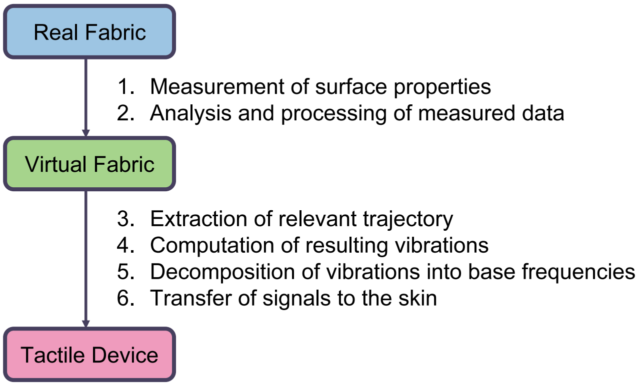

Fig. 7 Overview of the tactile rendering

can be done, the real fabric has to be converted into a virtual tactile fabric. Unlike the real-time rendering, this pre-processing is not time-critical. Thus it is desirable to use as much pre-computation as possible to speed up the time-critical rendering. The pre-processing is described in more detail in [18].

Vibrotaction, the response of the mechanoreceptors to varying forces on the skin, plays an important role in the perception of fine surface textures. Therefore, we compute the vibrations occurring in the fingertip while moving along the trajectory. These are decomposed in only two basic frequencies intended to directly stimulate the Pacinian and non-Pacinian receptors. We describe this process in more detail in [1].

6 System evaluation

The evaluation of the proposed model is required to ensure that all relevant features are transmitted by the system allowing the user to assess the mechanical properties of the hanging fabric. For the visual perception the most important feature is the dynamic behaviour of the textile, whereas it is the static behaviour for the haptic perception. As to verify the correct transmission of forces between the models in the static case, we compared the simulation with the real deformation of a hanging fabric. The test setup consisted of a fixed fabric with dimensions of 20 × 38 cm, as seen in Fig. 8. The simulated fabric is made up of 448 triangles. We applied a constant force at the bottom of the fabric and measured the elongation in the force direction. Moreover, we varied the force from 0.4 N up to 2.4 N to determine the deviation of the model. As the result we got a maximal error of 4 mm at 2.4 N in the simulation which is smaller than 1% of the total length of the stretched textile.

At the next step, we verified the contact forces that are sent to the haptic device. We did this by defining a fixed movement of the fingers reproducing the grasp and stretch.

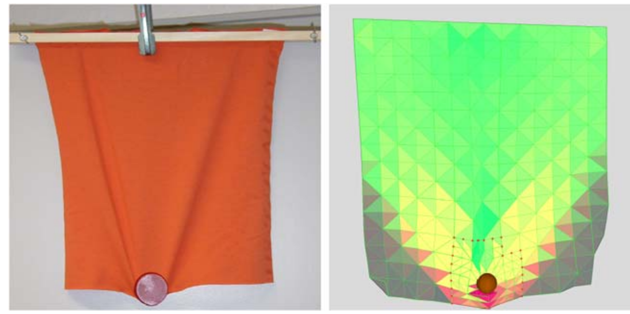

Fig. 8 Comparison between real and simulated deformations during contact

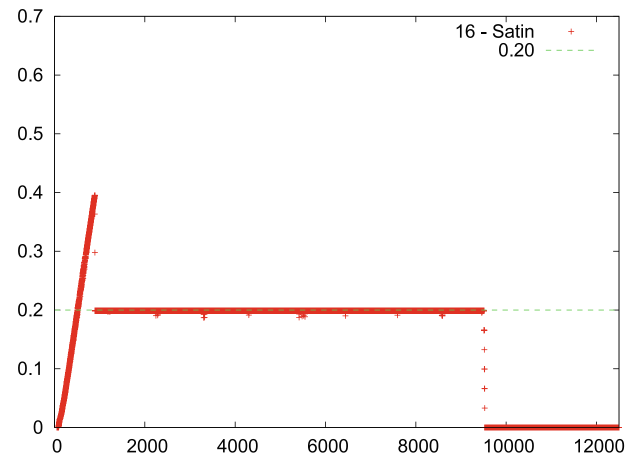

Fig. 9 Artificial user testing a fabric with friction $\mu_s = 0.4$ and $\mu_d = 0.2$; F^T/F^N on ordinate, time in ms on abscissa

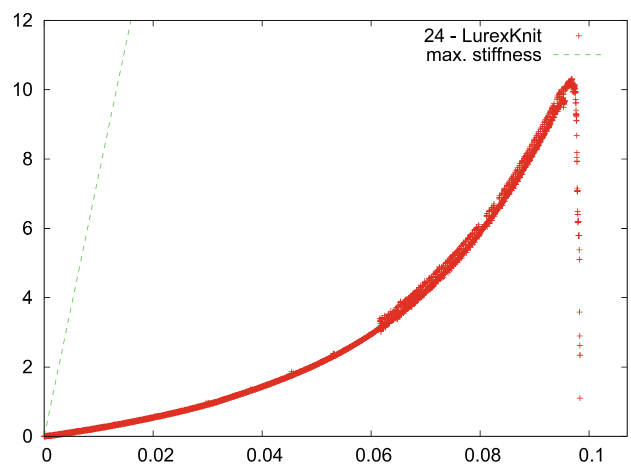

Fig. 10 Artificial user testing tensility; F^T on ordinate, elongation in meter on abscissa

We were able to observe the transitions between the friction states (Fig. 9) and the nonlinear strain–stress functions (Fig. 10).

A more thorough evaluation compared the subjective feel of real-world textiles to the one of the simulated textiles.

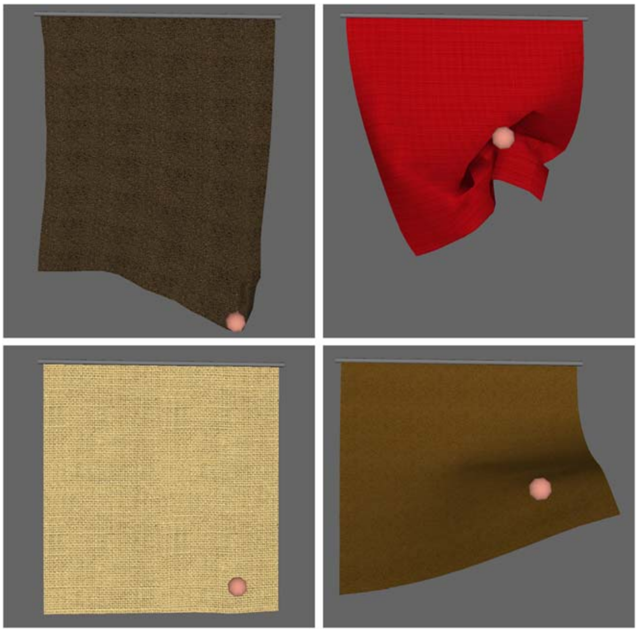

Fig. 11 Reference samples 1 (*top row*) and 5 (*bottom row*) for tensility (*left*) and bending (*right*)

A set of five textiles had to be rated in the range 1–5 for the four properties: tensility, roughness, friction, and bending. For each property two other textiles, called reference 1 and reference 5, were provided to define the scale. Two subjects (A, B) performed the test on real textiles; the two other (C, D) on the HAPTEX system. The tests on the virtual textiles were performed twice by each person. While the tests on the textiles were supposed to be performed blindfold, it proved to be too difficult to do this with the simulator. Therefore, the user was allowed to see a coarse wireframe model of the textile. Consequently, the tensile and bending tests were performed with vision on the real textiles as well.

The results of the evaluation were analysed by calculating the correlation of two successive runs of the same subject ("repeatability"), the correlation of the results of C and D ("consistency") and the correlation of the mean ratings of A and D with the mean ratings of C and D ("realism"). Tensility and roughness gave the best results with a correlation coefficient of at least 0.9 for repeatability, consistency and realism. Friction had a realism coefficient of 0.93, although its repeatability correlation coefficient was on average about 0.66. Unfortunately, bending showed no real correlation, giving coefficients as low as 0.07 for successive runs with the same subject. The reason for the latter was that with the haptic feedback being unable to represent the small forces resulting from bending, the user had to rely on what one could see to rate the bending property. However, rating the bending property by visual assessment was most likely impaired by the coarseness of the mesh not giving enough hints to the user. For complete results, see [21].

7 Conclusion

Haptic interaction with deformable physical objects poses many interesting challenges. Due to conflicting requirements in terms of physical accuracy and computation speed, a compromise has to be found. In this paper, we presented our approach by and large satisfying both requirements. The idea is to introduce an intermediate layer simulating the part of the deformable object that is in contact with the user. In this way we can accurately model the mechanics of the contact in real-time while still considering the global behaviour of the deformable object. The solution presented in this paper is tailored to the interaction with virtual fabrics but may be generalised to a larger class of deformable objects.

Unlike many other haptic rendering systems, our renderer allows for more than one-finger contact interaction; namely, with the user's thumb and index finger in contact situation. We handled the resulting complexity of the contact situation by dividing the problem into several cases treated by different algorithms, making good use of the computation power available. Our hybrid approach of using the proxy philosophy in combination with an inverted formulation allowed us to distribute the contact force more precisely. Furthermore, we are convinced that our algorithms can easily be extended to support an arbitrary number of fingers in contact. In simulations the system proved to be physically accurate and stable. Finally, a subjective evaluation showed that our VR system was able to reproduce the strong aspects of the physical interaction with textiles. Especially the combination with tactile feedback gave promising results.

7.1 General perspective

On a general level the system at hand requires a description of the energy transfer between parts of the deformable objects and between the deformable object and the human user (see Fig. 12). Typically, this energy exchange takes place and becomes visible via the boundary surfaces (e.g. via their deformation) of the 3D objects being involved, suggesting the development of appropriate boundary models of the respective objects mimicking the physics of 3D volumetric objects. The transmission of energy defining the physical process becomes visible and measurable via physical signals displaying changes of the physical states of the respective objects. Some of the physical signals result in "biological signals" in the human user involved in the interaction with deformable objects. To some extent the aforementioned biological signals create "perceivable signals",

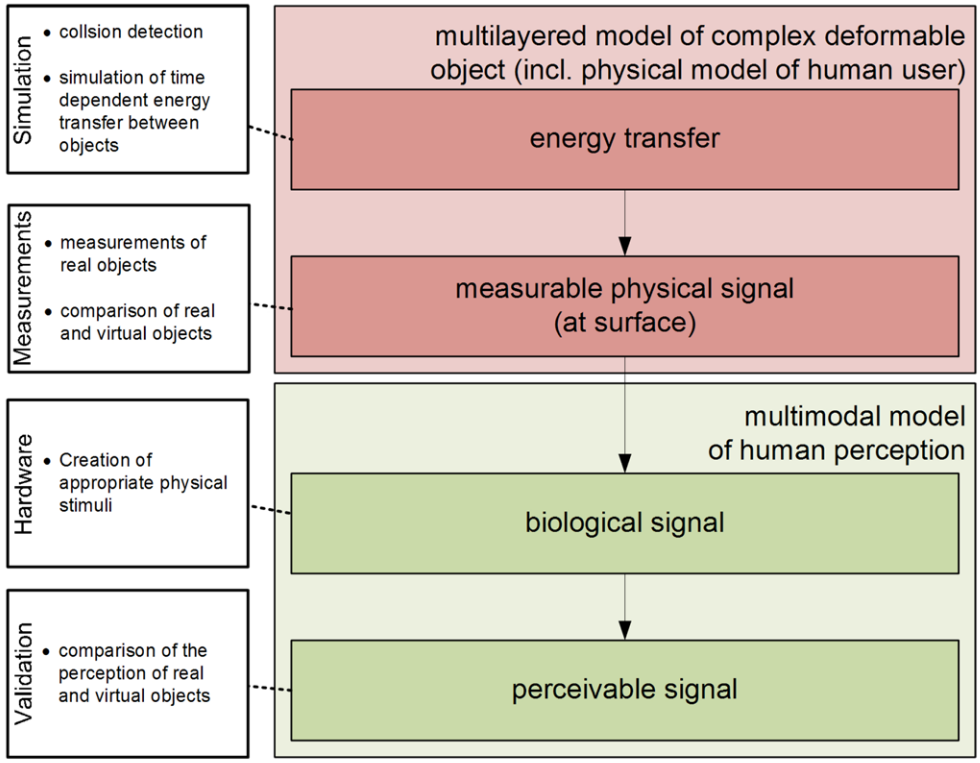

Fig. 12 Overview of the issues and problems involved in a systematic treatment of haptic rendering

e.g., needed by humans to control the interaction with objects (see Fig. 12). At this point, two key problems become important. In order to create a realistic haptic/tactile illusion by our VR-system, we have to find simplified models telling what physical or biological signals create equivalently perceived signals or illusions. Examples for the latter are presented by the RGB-, CMY-, CIE-colour models describing different physical signals creating equivalently perceived colours. Other examples for this are also given by our recent research (see [1]) on tactile colours modelling different vibrations causing equivalent perceptions of roughness. Furthermore, it is well known that in acoustic data compression psychoacoustic models are used to describe simplified acoustic signals being perceived as equivalent to more complicated acoustic signals. The other way, we must also deal with the inverse problem: to find out when and under what conditions can our perception distinguish physically different (multimodal) signals and attribute them to different, e.g. haptic, tactile, visual, features? Therefore the latter problem may be viewed as the task to model some basic cognitive capabilities of our haptic/tactile perception. The work presented in this paper may be seen as initialising a special instance of this general perspective.

Acknowledgements The work was conducted under the project "HAPtic sensing of virtual TEXtile" (HAPTEX) which is a research project funded by the Sixth Framework Programme (FP6) of the European Union (Contract No. IST-6549). The funding is provided by the Future and Emerging Technologies (FET) Programme, which is part of the Information Society Technologies (IST) programme and focuses on novel and emerging scientific ideas. Its mission is to promote research that is of a long-term nature or involves particularly high risks, compensated by the potential of a significant societal or industrial impact.

References

1. Allerkamp, D., Böttcher, G., Wolter, F.-E., Brady, A.C., Qu, J., Summers, I.R.: A vibrotactile approach to tactile rendering. Vis. Comput. **23**(2), 97–108 (2007). DOI 10.1007/s00371-006-0031-5
2. Balaniuk, R.: Using fast local modeling to buffer haptic data. In: PUG99: Proceedings of the Fourth PHANTOM Users Group Workshop (1999)
3. Baraff, D., Witkin, A.: Large steps in cloth simulation. In: SIGGRAPH '98: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, pp. 43–54. ACM Press, New York (1998). DOI 10.1145/280814.280821
4. Barbagli, F., Salisbury, K., Prattichizzo, D.: Dynamic local models for stable multi-contact haptic interaction with deformable objects. In: Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2003. HAPTICS 2003. Proceedings. 11th Symposium on, pp. 109–116 (2003). DOI 10.1109/HAPTIC.2003.1191248
5. Bergamasco, M., Salsedo, F., Fontana, M., Tarri, F., Avizzano, C.A., Frisoli, A., Ruffaldi, E., Marcheschi, S.: High performance haptic device for force rendering in textile exploration. Vis. Comput. **23**(4), 247–256 (2007). DOI 10.1007/s00371-007-0103-1
6. Botsch, M., Steinberg, S., Bischoff, S., Kobbelt, L.: OpenMesh—a generic and efficient polygon mesh data structure. In: Proceedings of OpenSG Symposium 2002 (2002)
7. Bro-Nielsen, M.: Finite element modeling in surgery simulation. Proc. IEEE **86**(3), 490–503 (1998). DOI 10.1109/5.662874

8. Cavusoglu, M.C., Tendick, F.: Multirate simulation for high fidelity haptic interaction with deformable. In: Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on, vol. 3, pp. 2458–2464 (2000). DOI 10.1109/ROBOT.2000.846397
9. Duriez, C., Andriot, C., Kheddar, A.: Signorini's contact model for deformable objects in haptic simulations. In: Intelligent Robots and Systems, 2004 (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on, vol. 4, pp. 3232–3237 (2004). DOI 10.1109/IROS.2004.1389915
10. Etzmuß, O., Keckeisen, M., Straßer, W.: A fast finite element solution for cloth modelling. In: PG '03: Proceedings of the 11th Pacific Conference on Computer Graphics and Applications, pp. 244–251. IEEE Computer Society, Washington (2003)
11. Fontana, M., Marcheschi, S., Tarri, F., Salsedo, F., Bergamasco, M., Allerkamp, D., Böttcher, G., Wolter, F.-E., Brady, A.C., Qu, J., Summers, I.R.: Integrating force and tactile rendering into a single VR system. In: Cyberworlds, 2007. CW '07. International Conference on, pp. 277–284 (2007). DOI 10.1109/CW.2007.40
12. Hutchinson, D., Preston, M., Hewitt, T.: Adaptive refinement for mass/spring simulations. In: Proceedings of the Eurographics Workshop on Computer Animation and Simulation '96, pp. 31–45. Springer, New York (1996)
13. Magnenat-Thalmann, N., Volino, P., Bonanni, U., Summers, I.R., Bergamasco, M., Salsedo, F.: Wolter. F.-E.: From physics-based simulation to the touching of textiles: The HAPTEX project. Int. J. Virtual Real. **6**(3), 35–44 (2007)
14. Mahvash, M., Hayward, V.: High-fidelity haptic synthesis of contact with deformable bodies. Comput. Graph. Appl. IEEE **24**(2), 48–55 (2004). DOI 10.1109/MCG.2004.1274061
15. Mark, W.R., Randolph, S.C., Finch, M., Van Verth, J.M., Taylor, R.M.: Adding force feedback to graphics systems: issues and solutions. In: SIGGRAPH '96: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, pp. 447–452. ACM Press, New York (1996). DOI 10.1145/237170.237284
16. Mazzella, F., Montgomery, K., Latombe, J.C.: The forcegrid: a buffer structure for haptic interaction with virtual elastic objects. In: Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on, vol. 1, pp. 939–946 (2002). DOI 10.1109/ROBOT.2002.1013477
17. Pawluk, D.T.V., Howe, R.D.: Dynamic lumped element response of the human fingerpad. J. Biomech. Eng. **121**(2), 178–183 (1999). DOI 10.1115/1.2835100
18. Peinecke, N., Allerkamp, D., Wolter, F.-E.: Generating tactile textures using periodicity analysis. In: Cyberworlds, 2007. CW '07. International Conference on, pp. 308–313 (2007). DOI 10.1109/CW.2007.38
19. Ruspini, D.C., Kolarov, K., Khatib, O.: The haptic display of complex graphical environments. In: SIGGRAPH '97: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, pp. 345–352. ACM Press/Addison-Wesley, New York (1997). DOI 10.1145/258734.258878
20. Ruspini, D.C., Kolarov, K., Khatib, O.: Haptic interaction in virtual environments. In: Intelligent Robots and Systems, 1997. IROS '97. Proceedings of the 1997 IEEE/RSJ International Conference on, vol. 1, pp. 128–133 (1997). DOI 10.1109/IROS.1997.649024
21. Summers, I.R.: Final demonstrator and final integration report. Tech. rep., HAPTEX—HAPtic sensing of virtual TEXtiles (2008). http://haptex.miralab.unige.ch
22. Volino, P., Davy, P., Bonanni, U., Luible, C., Magnenat-Thalmann, N., Mäkinen, M., Meinander, H.: From measured physical parameters to the haptic feeling of fabric. Vis. Comput. **23**(2), 133–142 (2007). DOI 10.1007/s00371-006-0034-2
23. Volino, P., Davy, P., Bonanni, U., Magnenat-Thalmann, N., Böttcher, G., Allerkamp, D., Wolter, F.-E.: From measured physical parameters to the haptic feeling of fabric. In: Proceedings of the HAPTEX'05 Workshop on Haptic and Perception of Deformable Objects, Hanover, pp. 17–29 (2005)
24. Volino, P., Magnenat-Thalmann, N.: Accurate anisotropic bending stiffness on particle grids. In: Cyberworlds, 2007. CW '07. International Conference on, pp. 300–307. IEEE Computer Society, Washington (2007). DOI 10.1109/CW.2007.11
25. Zhuang, Y., Canny, J.: Haptic interaction with global deformations. In: Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on, vol. 3, pp. 2428–2433 (2000). DOI 10.1109/ROBOT.2000.846391
26. Zilles, C.B., Salisbury, J.K.: A constraint-based god-object method for haptic display. In: Intelligent Robots and Systems 95. 'Human Robot Interaction and Cooperative Robots'. Proceedings. 1995 IEEE/RSJ International Conference on, vol. 3, pp. 146–151. IEEE Computer Society, Los Alamitos (1995). DOI 10.1109/IROS.1995.525876

Guido Böttcher studied mathematics and computer science at the Leibniz University of Hannover and wrote his diploma thesis on medial axis and haptics. He received his diploma in mathematics (master's equivalent) in January 2005. Since then he is working at the Division of Computer Graphics of the University of Hannover as a research assistant and Ph.D. candidate. Currently, he is mainly involved in the HAPTEX project.

Dennis Allerkamp studied mathematics and computer science at the Leibniz University of Hannover and wrote his diploma thesis on the computation of medial sets and respectively curved surfaces in Euclidean space. He received his diploma in mathematics (master's equivalent) in December 2004. Since then he is working at the Computer Graphics Division of the University of Hannover as a research assistant and Ph.D. candidate. His research interests are in the area of haptics, human–computer interfaces, as well as medial axis. Currently, he is mainly involved in the HAPTEX project.

Daniel Glöckner studies mathematics and computer science at the Leibniz University of Hannover and writes his diploma thesis on real-time simulation and haptics. Since 2006 he is working at the Computer Graphics Division of the University of Hannover as a student assistant, being mainly involved in the HAPTEX project.

Franz-Erich Wolter has been a full professor of computer science at the Leibniz University of Hannover since the winter term of 1994/1995 where he directs the Division of Computer Graphics and Geometric Modeling called Welfenlab. Before coming to Hannover, Dr. Wolter held faculty positions at the University of Hamburg (in 1994), MIT (1989–1993) and Purdue University in the USA (1987–1989). Prior to this he developed industrial expertise as a software and development engineer with AEG in Germany (1986–1987). Dr. Wolter obtained his Ph.D. in 1985 from the department of mathematics at the Technical University of Berlin, Germany, in the area of Riemannian manifolds. In 1980 he graduated in mathematics and theoretical physics from the Free University of Berlin. At MIT Dr. Wolter co-developed the geometric modelling system Praxiteles for the US Navy from 1989 to 1993 and published various papers that broke new ground applying concepts from differential geometry and topology on problems and design of new methods used in geometric modelling and CAD systems. Dr. Wolter is a research affiliate of the MIT department of mechanical engineering.