Workshop on Virtual Reality Interaction and Physical Simulation VRIPHYS (2014)
J. Bender, C. Duriez, F. Jaillet, and G. Zachmann (Editors)

SutureHap: A suture simulator with haptic feedback

E. Ricardez[†1,2] and J. Noguez[2] and L. Neri[2] and L. Munoz-Gomez[3] and D. Escobar-Castillejos[2]

[1]Instituto Politécnico Nacional, ESIME Culhuacán
[2]Tecnológico de Monterrey, Campus Ciudad de México
[3]Tecnológico de Monterrey, Campus Santa Fe

Abstract

Surgical procedures require a high degree of complexity and difficulty. Consequently, extensive preparation in the learning process of medical students is necessary in order to perform suturing tasks successfully. Some authors suggest that a minimum of 750 operations are needed to acquire the experience to perform correctly surgical procedures. Moreover, current laws establish standards if corpses and animals are used as medical learning environments; as a result, the development of skills and processes is hindered. This paper introduces the development of a virtual environment for training suture skills: SutureHap, which uses two Sensable Phantom Omni haptic devices. To create a proper simulation of the human skin which must fulfill graphic and physical characteristics, NVIDIA PhysX libraries were usea. Some of these libraries were originally defined to represent cloths; however, in this work some parameters were adjusted to obtain the desired simulation. An architecture that facilitates the integration of haptic devices was designed. A simplified method of collision detection and haptic feedback generation was created. This enabled the reduction of complexity generated during collision detection, and it diminished the time to develop the virtual environment. Tweezers, thread and needle models were added in the virtual environment. Due to fact that PhysX exploits GPU processing, response time was improved during modeling of the skin. Additionally, suturing tasks were designed by taking into consideration real procedures made by medical experts. The acquisition of skills and competencies in suture process are increased through haptic devices due to the fact that they can send tactile sensations. These environments decrease costs and risks, and provide real sensations as the ones that can be perceived in current learning environments. Finally, an evaluation focused on the perception of this environment was made by students. Preliminary results are promising, and it is expected that this environment facilitates the acquisition of suture skills.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual Reality I.6.3 [Simulation and Modeling]: Applications—

1. Introduction

Considering advances within technology and new laws in medical field, various solutions have been pursued to allow medical students to acquire the necessary skills involved in suture procedures. Some solutions that have been implemented involve the use of animals. However, the use of animals as learning environments must be done in operating rooms, which must have similar conditions as those related to humans. Devices that use synthetic materials to resemble human skin have also been implemented; nevertheless, they fail to emulate real characteristics of human skin. Other solutions suggest the use of virtual environments, which can simulate various situations with different levels of risk. Virtual environments developed so far only exploit the senses of sight and hearing; consequently, they exclude other interaction possibilities. Finally at the next level of technology, haptic interfaces arise as novel solutions in medical field. Haptic devices are characterized by being bidirectional, which enable computer equipment the facility to send tactile and kinesthetic sensations to users.

In recent years, there has been acceptance in the use of haptic devices in medical simulations and simulators in the branch of health science by various research groups

† Email: eusebior@ieee.org

DOI: 10.2312/vriphys.20141226

[CDA99] and related companies in the area [BSL*02]; for instance, Immersion Medical [Med13], Surgical Science [SCI13a] [SCI13b], Mentice [MEN13], and Reachin Technologies [TEC13].

In this work, the process for creating a virtual environment for medical training is described. This simulator allow students to perform suturing tasks using haptic devices, and the system provides a realism that is closer to the one experienced in real suture tasks.

2. Related work

There are several works in the literature that analyze suture performed in virtual environments. However, most of them have problems in the modeling process. For example, an application was designed by Webster [WZM*01]. The environment uses a needle holder connected to a haptic device to add realism, and its graphical representation, as well as needle, suture thread and virtual skin models, is deployed in the virtual world. Although this modeling is quite challenging and it is updated in real-time, it has failures in modeling the task of suture. Even though it is a very good simulation, the methodology includes the use of one hand while doctors use both; in other words, the standard teeth tissue forceps that is used together with the needle holder is omitted.

Brown [BLM04] proposed a new algorithm to handle collisions, control suture thread, and make knots. This algorithm includes haptic feedback; nevertheless, the final application focuses on microsurgery. On the other hand, Miyazaki [MYS*06] developed a simulator for buried suture, which used a Sensable Phantom Omni 1.5/6 DOF. In this environment, the effects from buried suture processes on the surface of the virtual skin is evaluated. Evaluation is made by measuring the accuracy in the suture points. In this work a hand is used to control the haptic device and the other one is used to manipulate a keyboard. It has to be noted that the essential work made by Miyazaki is dedicated to the simulation of the different components, such as thread, needle, and different skin textures. The authors acknowledge their shortcomings in the use of the haptic device, and they even consider the use of another similar device to use jointly both hands in suturing tasks as future work.

There are also researches that involve the modification of existing hardware, such as the one made by [OSO*07]. A design of a haptic device applied to suture experiments was made; however, this application is not focused properly on the work of suture. The development of new simulators relies principally in the modelling part. [MDH03] found convenient to defragment simulated models to create realistic organs, tools and interactions. He developed physical and collision detection modules [LMGC02] [LMGC04] that are applied in a virtual environment dedicated to surgical simulation. The work of Lenoir focused in the development of an environment which enables the simulation of various deformable bodies, and further improvements on the system are proposed.

Another interesting work in this field was developed by Choi [CCP12], who proposed the use of PhysX, a graphics engine developed by NVIDIA. It has to be noted that this engine provides an effective way to create solid and soft objects. In addition, OpenGL and OpenHaptics were used to collaborate with PhysX libraries in the virtual environment. The prototype uses a workstation, which consists of a computer and two Sensable Phantom Omni devices. Due to the accessibility and advantages provided by its libraries, PhysX was used to create the thread model. These libraries allow researchers to use the GPU of the graphics card instead of the CPU of the computer. This feature allows the CPU to focus on performing other tasks in the system, while the GPU executes and creates the graphical environment. The methodology that Choi proposed has important processing advantages; however, the skin model proposed is unrealistic. Additionally, the simulation uses a straight needle and it has to be considered that in real procedures most sutures are performed with curved needles.

At last, existing commercial systems for suture training have been focused mainly in the field of laparoscopy. Such is the case of LapMentor [CMJ11], a simulator made by Simbionix Company in conjunction with Immersion. This system includes an advanced suturing module, which allows simulation and training of laparoscopic suturing using both hands. Another similar case is the one in the MIT. Through the use Touch Lab [BDM*04], they have developed several algorithms focused on haptic rendering and particle systems simulations, where the last one is based on deformable elastic models to create probes and catheters. Nevertheless, these applications are focused on laparoscopy.

3. Materials and models

Considering the issues raised in the introduction, virtual training environments that incorporate haptic devices represent an important alternative in the learning process of future surgeons. In this paper the development of a virtual medical training environment is proposed. This system supports the acquisition of suture skills, and it is able to generate similar sensations to those produced in real suture procedures.

3.1. Architecture of the application

This work is based on the architecture proposed by [SCB04]. However, according to the components needed for a virtual environment of this type, an adaptation of the proposal was made (Fig. 1). Following paragraphs are going to detail each of the blocks that make up this architecture:

Simulation.- PhysX [NVi12] v2.84 was used as the core of the simulation. The core of the system is where the virtual world is created and it responds to the laws of classical physics. This physics engine is able to simulate the behavior of the cloth in the GPU, and it can manage collisions between objects. However, nothing is reported if collisions between solid and deformable objects occurred. To overcome this deficiency and generate an appropriate haptic response between the skin (represented as a cloth) and the haptic cursor, the simplified method described in section 3.2.2 is used.

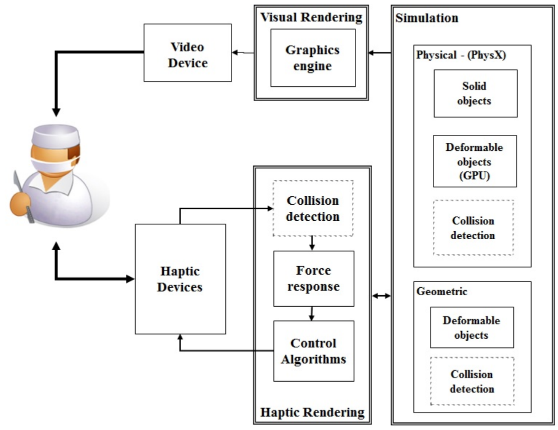

Figure 1: *Architecture of SutureHap.*

Haptic Rendering (HR).- For purposes of this work, three blocks of HR were developed (Fig. 1). Collision detection and calculation of force feedback response between the cloth and the haptic cursor is performed by simplified model. On the other hand, response force between solid objects was developed according to "penalty method" standard technique [SCB04]. In the conditioning section, the maximum force is restricted and routines to improve the perception, according to the haptic device that is being employed, are inserted.

Visual Rendering (HR).- In this section, the necessary operations to create the environment and display it in the video device are done. Basic deployment techniques are used and no hardware acceleration is used.

3.2. Calculating collisions and haptic rendering

When haptic devices are incorporated into graphical virtual environments, it is necessary to send a force to the haptic, which allows the user to perceive the feeling of interacting with the environment. Due to the presence of solid and deformable objects in virtual environments, a major problem for this task is to determine, calculate and generate the needed force feedback.

Knowing that force feedback must be sent to the haptic device in a minimum frequency of 1 KHz to update it, this calculation must be done quickly and accurately.

GPU processing helps during the reduction of computational time. This feature is used to solve large number of calculations within the creation of a deformable object using mass-spring models.

Furthermore, acceptable visual quality during training sessions in virtual environments is a challenge that need to be considered; therefore, the graphical presentation needs to be updated with at least 30 fps.

The current way to generate haptic rendering involves the development of complete systems, which includes mathematical modeling and application programming. As a result, a customized modeling should be made. According to [BS02], the main methods to generate haptic rendering for deformable objects are based on geometry and/or physical models. Considering the technological advance in processing power, in this work a method based on physical modeling for skin and geometry modeling for the thread was developed.

3.2.1. Physical basis

A cloth can be considered as a flat array of particles with a mass connected by springs (Fig. 2, see for example [WZM*01]). An analysis of the forces on the particles of the array is discussed. This analysis generates the force that is going to be sent to the haptic device when the haptic avatar interacts with the cloth.

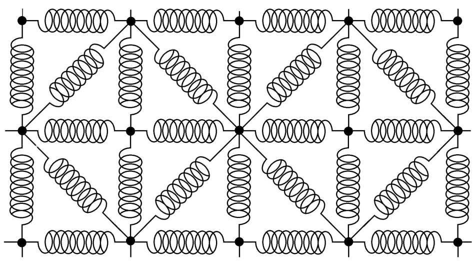

Figure 2: *Bidimensional particle array that generates the cloth.*

Assuming that the interaction force between each pair of particles follows Hooke's law plus a damping term, the force on particle i caused by particle j can be written as follows:

$$\vec{F}_{ij} = -k|\Delta x_{ij}|\widehat{r}_{ij} - b\vec{v}_{ij} \quad (1)$$

In equation 1, $|\Delta x_{ij}|$, k and b are, respectively, the elongation from the equilibrium position of the spring connecting the particles i and j, the force of the spring constant, and the damping constant. Furthermore, $\widehat{r}_{ij}$ is the unit vector that connects particles i an j, and $\vec{v}_{ij}$ is the velocity of particle i relative to the particle j. When the sphere strikes the cloth, the N particles that are found in the contact area are displaced in different directions and magnitudes. This behavior depends on the direction and magnitude of the force that the sphere applies on the cloth, which pulls neighbor particles in the array. The value of N depends on the size of the sphere: the greater the diameter of the sphere is, the greater the value of N will be.

The resulting force on the portion of the cloth that is in contact with the sphere is therefore the vectorial sum of i) the force applied by the sphere over the region of contact, ii) the spring forces on each of the N particles from the contact region due to their N_i neighbor particles (8 or 4 depending on the particle, as it is shown in figure 2), and iii) the weight of each of the particles of the cloth in the contact region. This is formulated according to Newton's Second Law:

$$\vec{F}_{on\,cloth} = \vec{F}_{sphere\,on\,cloth} + \sum_{i=1}^{N}(\sum_{i=1}^{N_i}\vec{F}_{ij}) + \sum_{i=1}^{N}\vec{W}_i = (\sum_{i=1}^{N} m_i)\vec{a}_{CM} \quad (2)$$

In equation 2, the term $\sum_{i=1}^{N_i}\vec{F}_{ij}$ represents the force on particle i produced by its N_i neighboring particles; thus the term $\sum_{i=1}^{N}(\sum_{i=1}^{N_i}\vec{F}_{ij})$ is the resultant force on all N particles in the contact region due to their neighboring particles. It should be noticed that the inner forces between particles of the contact region cancel each other due to Newton's third law. Consequently, only the external forces exerted by neighbor particles that are not on the contact region are considered. On the other hand, the terms $\sum_{i=1}^{N}\vec{W}_i$ and $\sum_{i=1}^{N} m_i$ represent, respectively, weight and mass of all the particles of the cloth that are in contact with the haptic sphere . Finally, $\vec{a}_{CM}$ is the center of mass acceleration of the contact area.

From equation 2, the force exerted by the sphere on the cloth is:

$$\vec{F}_{sphere\,on\,cloth} = (\sum_{i=1}^{N} m_i)\vec{a}_{CM} - \sum_{i=1}^{N}(\sum_{i=1}^{N_i}\vec{F}_{ij}) - \sum_{i=1}^{N}\vec{W}_i \quad (3)$$

Therefore, the force exerted on the haptic sphere is obtained applying Newton's Third Law:

$$\vec{F}_{cloth\,on\,sphere} = \sum_{i=1}^{N}(\sum_{i=1}^{N_i}\vec{F}_{ij}) + \sum_{i=1}^{N}\vec{W}_i - (\sum_{i=1}^{N} m_i)\vec{a}_{CM} \quad (4)$$

Equation 4 represents the force that must be calculated and modeled for the haptic system. This task may be complicated due to the following reasons:

i) PhysX provides values for average stretching stiffness of cloths, but it does not provide the values of the constants k and b.

ii) Since it is a dynamic system, speeds are different for each particles; moreover, their value decreases as the particles move away from the contact point. Consequently, the calculation of relative speeds in real time of adjacent particles $\vec{v}_{ij}$ within the cloth in virtual environments is not easy.

iii) In the same way as in the case of speeds, calculation of the accelerations of particles in the array is also complicated. Therefore, $\vec{a}_{CM}$ has to be calculated independently. One possibility would be to manipulate the haptic sphere at a constant speed and perpendicular to the cloth, without passing through it. In this case it can be considered that the center of mass of the portion of cloth in contact will also move with an approximate constant speed; as a result, the term $\vec{a}_{CM}$ of equation 4 is nullified. However, moving the haptic sphere with a constant velocity is difficult when haptic devices are used. The device exerts a reaction force on the user, whose magnitude depends on the degree of deformation of the cloth. In particular, it is more difficult to move the sphere near the ends of the mesh than move it in the middle. This is due to the fact that the ends of the mesh are fixed. Moreover, in case that the haptic sphere is tangentially displaced to the cloth even with constant speed and without penetrating it, there would be a movement upwards and downwards of the particles in a perpendicular direction to the cloth; consequently, the term $\vec{a}_{CM} \neq \vec{0}$.

Accordingly, this paper proposes a simplified method, which takes advantages of the positions of the particles provided by PhysX. This enables the calculation of the force that the cloth exerts on the haptic sphere, as it will be presented in section 3.2.2. Haptic perception resulting from this method will be evaluated in later sections.

3.2.2. Simplified method

To solve the problem stated in section 3.2.1, a novel algorithm has been developed. It is based on the addition of multiple rays emanating from the center of the haptic avatar (represented as a sphere) in different directions (Fig. 3). Through these rays, it is possible to calculate the distance between the haptic avatar and the cloth.

When this contact occurs, the number of rays that interacts with the cloth are calculated and the contact points are identified. As the sphere is moved against the cloth and for each ray that interacts with the cloth, the initial position point and the position of this point in subsequent times, provided by PhysX, are stored. Thereby, the displacement vector for these two positions is generated. Subsequently, a weighted sum of movement vectors is made, which enables the computation of the force to be displayed in the haptic device. The full method is described in the following paragraphs.

The cloth is represented in PhysX as a particle network interconnected by spring-damper arrangements (Fig. 3). This cloth has physical properties such as width, length, thickness and density, as well as qualities of elasticity and breaking capacity.

Taking into consideration that the user will explore with a deformable elastic surface, represented as a cloth, the algorithm establishes that the exploration will be made using a

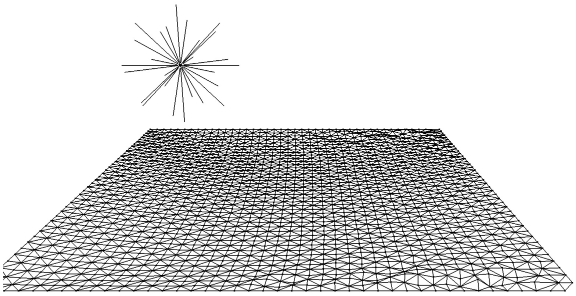

Figure 3: *Deformable particle mesh formed by interconnected springs and dampers, and scanning sphere with rays.*

haptic device. The device is represented by a sphere with determined radius in the virtual environment. The exploration rays allow the measurement of distances for collision detection and calculation of forces. These rays are originated from the center of the haptic sphere (Fig. 3).

For each ray, an estimation is made to determine whether there is contact between the sphere and the cloth. In order to obtain this value, the distance from the surface of the sphere to the cloth is calculated. If the distance is less than or equal to half of magnitude of the thickness of the cloth, then the algorithm establishes that a contact has occurr and this point is considered as the initial contact point (P_{i1}) (Fig. 4a).

When a ray strikes the deformable surface, the distance to the nearest point on the cloth is calculated regardless the beam is incident on a particle of the array or not. Figure 4b shows the generation of new initial contact points P_{i2} and P_{i3} when the sphere is moved. Figure 4c represents current contact points P_1, P_2 and P_3. Then, a series of displacement vectors, which considers the distance and direction between the initial contact point P_{in} and current contact point P_n, are generated (Fig. 4d). Thus, for the n-th contact point:

$$\vec{R}_n = P_n - P_{in} \tag{5}$$

From all the displacement vectors obtained from the rays that are in contact with the cloth, the sum vector is computed using equation 6.

$$\sum_{n=1}^{N} \vec{R}_n = \vec{R}_1 + \vec{R}_2 + ... + \vec{R}_N \tag{6}$$

Finally, the calculation of the haptic feedback force is performed as follows:

$$\vec{R}_{cloth\,over\,haptic} = -\alpha \frac{\sum_{n=1}^{N} \vec{R}_n}{N} \tag{7}$$

In equation 7, α is a factor that depends on the elasticity of the material, and N is the number of rays that intersect the cloth area. In this work the value of α is selected such that there is consistency between the physical effort to move the haptic device and the visual perception of the avatar motion on the screen.

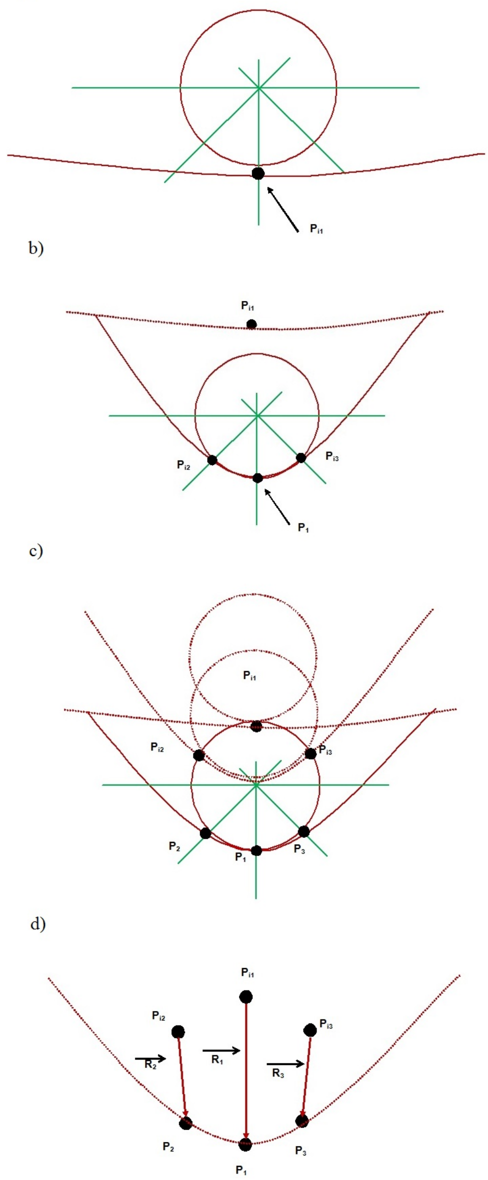

Figure 4: *Construction of displacement vectors. a) Initial contact b) Contact of the following rays c) Movement of the contact points d) Generation of displacement vectors.*

The direction of the resultant vector obtained corresponds to the sum of the calculated vectors. Consequently, the opposite direction is used to send the haptic feedback force to the user.

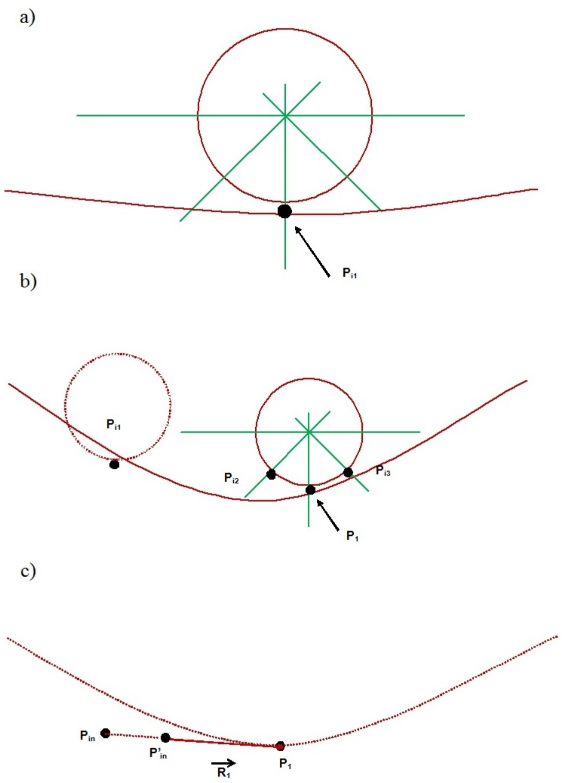

Figure 5: *New displacement vector a) Initial contact b) Sliding c) Vector generation.*

If the initial contact point remains fixed, as it is considered so far, the current point of contact can produce a force with high magnitude for large displacements, which is not desired. To ease this situation and improve the haptic perception, a modification of the coordinates of the initial point of contact is proposed in this work (Eq. 8). This procedure is made to limit the maximum force that is going to be sent to the user.

$$P'_{in} = P_{act} - (\vec{R}_N * r) \tag{8}$$

where:
P'_{in} = New initial point of contact.
P_{act} = Current position of the haptic sphere.
$\vec{R}_N$ = Displacement vector calculated using equation 5.
r = Constant of maximum distance. This constant is chosen to limit the magnitude of the maximum force that is going to be sent to the user.

The recalculation is done according to figure 5, where in first instance the initial contact is displayed and the initial point of contact P_{i1} is marked (Fig. 5a). In figure 5b, the next contact point and the initial contact points for two of the beams are shown. Finally, figure 5c shows the construction of the new displacement vector:

$$\vec{R}_n = P_n - P'_{in} \tag{9}$$

Considering the case in figure 5, a resultant vector corresponding ($\sum_{n=1}^{N} \vec{R}_n$) is constructed for each of the rays. Finally by using equation 7, the resultant force that will be sent to the haptic device is obtained. This is the algorithm that will be implemented in this work, both for movement of the sphere perpendicular to the cloth or horizontal slipping on it.

This technique can be used in any environment that simulates the behavior of a mesh made by particles interconnected by springs. However, this algorithm can be used only if the following data can be obtained:

i) The contact point of each of the rays emerging from the haptic sphere.
ii) Closest vertexes to the point of contact.
iii) The positions of each of the above.

The faster these positions are calculated, the better the haptic response will be. This is justified due to the fact that the optimum refresh rate of haptic devices is $\geq$ 1kHz. As a result, it is recommended that these calculations are performed on the GPU. These features are provided by NVIDIA PhysX, which is the engine used in this work.

3.3. Thread model

To create a proper thread model, this work used as basis the technique shown in [LC06] combined with the ones in [BLM04]. This methodology was implemented to achieve proper interaction in the GPU between the simulated skin and the thread, which should have a strict control of collision detection. Since PhysX engine was used in this research, an additional mechanism of collision detection was added. To build the movement of the thread, the "follow the leader" (FTL) algorithm described [BLM04] was used. Figure 6 shows a segment of the thread. In this model particles are represented by spheres. The line segments connecting the center of each particle can be appreciated and the cylinders generated from the particles are observed.

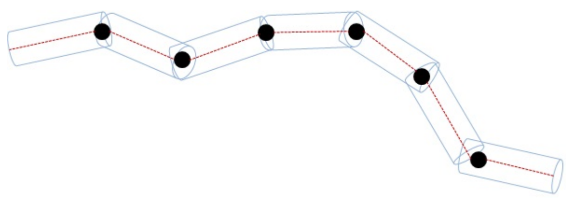

Figure 6: *Thread segment made from particles, represented as spheres.*

For collision detection between the thread segments and between the thread and other objects, a method based on geometric properties of line segments, formed by joining the centers of the particles (Fig. 6), is proposed in this work.

The following paragraphs describe the collision detection method, which considers the situation shown in figure 7.

Let p, p', q and q' be centers of four spheres that form the thread. v and w are vectors made from the points p, p', q and q respectively. Finally, A and B are the points on lines $\overline{pp'}$ and $\overline{qq'}$ that are closest between them.

To achieve this condition, it is necessary that the line formed between A and B is perpendicular to $\overline{pp'}$ and $\overline{qq'}$ lines.

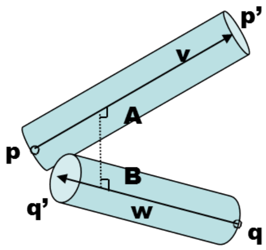

Figure 7: *Collision of a pair of cylinders that make up the thread.*

To calculate the position of A and B, the following procedure is made. Vectors v and w are defined by:

$$v = p - p' \tag{10}$$

$$w = q - q' \tag{11}$$

Therefore,

$$A = p + tv \tag{12}$$

$$B = q + \lambda w \tag{13}$$

where $0 \leq t \leq 1$ and $0 \leq \lambda \leq 1$.

At the same time, the condition of perpendicularity is conceived as:

$$v \cdot (B - A) = 0 \tag{14}$$

$$w \cdot (B - A) = 0 \tag{15}$$

The condition to consider that a collision between segments occurred is:

$$||\vec{AB}|| < 2r \tag{16}$$

where r is the radius of each cylinder.

Developing equations 10-16 and considering the condition of perpendicularity, equations 17a-17b are obtained.

$$v \cdot (q + \lambda w - p - tv) = 0 \tag{17a}$$
$$w \cdot (q + \lambda w - p - tv) = 0 \tag{17b}$$

Rearranging:

$$(v \cdot w)\lambda - (v \cdot v)t = v \cdot (p - q) \tag{18a}$$
$$(w \cdot w)\lambda - (v \cdot w)t = w \cdot (p - q) \tag{18b}$$

Equations 18a and 18b are solved for λ and t and according to the conditions stated above, the values of λ and t should be between 0 and 1 to ensure the collision.

When segments are too close and it is not possible to move them to avoid a collision, the proposed algorithm consider that a knot can exist if and only if the following conditions are met:

a) 10 or more colliding cylinders must exist.
b) Colliding cylinders are in at least two different segments.
c) Each block contains at least four contiguous cylinders.

The above conditions are met in the example of figure 8, where two segments of the thread, one green and one red, are shown. Cylinders ABCDEF, which are contiguous and belong to the red segment, are colliding against cylinders abcde, which are also contiguous and belong to the green segment.

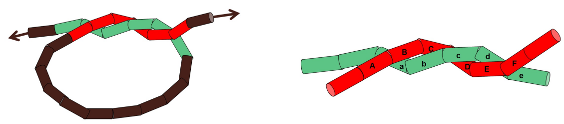

Figure 8: *Knot detection example.*

It is important to highlight that in order to detect colliding cylinders that belong to different segments, there must be a distance interval of at least five cylinders between the segments.

At the time that the possible existence of a knot is detected, the distance of these segments with respect to the skin (cloth) is evaluated. If the knot comes to a predetermined distance, it is considered that the knot is getting closer and it will be fixed. Consequently, the system will immobilize all segments that forms the knot.

3.4. Surgical tools

In this research, it was necessary to prepare several models of surgical tools. The models were created using AutoDesk Maya (Fig. 9). For the initial stage of this research, only the blunt-nosed thumb forceps, Kelly forceps (needle holder), and Metzenbaum surgical straight scissor were used.

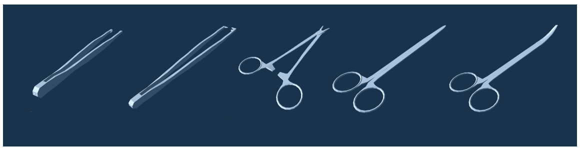

Figure 9: *Various models of instruments developed to be used in SutureHap. From left to right: Blunt-nosed thumb forceps, Standard teeth tissue forceps, Kelly forceps, Metzenbaum surgical straight scissor, and curved Metzenbaum surgical curved scissors.*

A curve needle model was created (Fig. 10).This needle is attached to the suture thread, which is created at the time of execution without a predefined pattern because it possess a dynamic behavior.

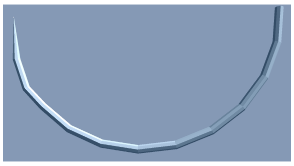

Figure 10: *Model of the suture needle.*

To simulate movements in SutureHap, each tweezers model has been segmented in separated parts, with the pivot in the right place for the rotation of each piece. In the environment, the user can rotate them as he/she desired, and he/she can interact in the virtual environment by opening or closing of the gripper of each tool.

3.5. Skin model

To render the skin, a deformable mesh based on the representation of the cloth was used (Fig. 3). This mesh is placed in the center of the skin, and it is configured with a certain width or thickness, which improves skin characteristics. To achieve different consistencies, the elasticity and damping parameters are adjusted. To generate the skin, PhysX NxCloth class is used. In the case of the haptic avatar, a Kinematic Actor from PhysX NxActor class is created. A special routine was developed to generate and control the rays, which are used to detect collisions between the haptic avatar and the skin. Additionally, the subroutine is used to calculate the feedback force vectors. Finally, to improve the appearance, a similar texture of human skin is added.

4. Evaluation and results

In Figure 11, the graphic appearance of SutureHap and the sequence of steps required to perform a simple suture knot. The sequence of steps implemented is mentioned in [BTB*04], as it follows:

A) Insert the needle into the skin: the needle is inserted into the skin while the skin is grabbed by the tissue forceps. In this case the user must take into consideration the path and position of the needle. The puncturing of the skin by the needle is provided by PhysX. When the skin elasticity limit is exceeded in a given point contact between the needle point (simulated as a small haptic sphere) and the skin, the feedback force that is sent to the haptic device is set back to zero. An analysis of the rendering forces involved when an haptic sphere goes through a cloth will be presented in more detail in an upcoming work.

B) Pull the thread: At this point, the needle has pierced both sides of the wound, and the thread is going through both holes. The user is pulling the thread with the needle. In the case of suture-thread the magnitude of force to be sent to the haptic device is calculated from the difference between the current length and the ideal length. The direction of the force is calculated from the vector sum of each segment of the thread.

C) Preparation of knot: the thread is wounded on the clip holder; therefore, the loose end of the thread can be grabbed to create the first knot.

D) Creation of the first knot: a knot is formed and it is closed near the surface of the skin.

E) Fixed knot in the skin: the knot has been fixed in the skin and a second knot is being created.

F) Second suture knot: the used is preparing a second knot by winding the thread on the tweezers.

G) Cutting the thread: after ending the second suture knot and fixing it, the thread is cut with the scissors.

H) End of the task: the suture point has been finished after cutting the thread and the user can create a new point.

A series of user tests were performed using the following equipment: DELL T7500 Workstation, Intel processor XEON E5620 at 2.4 GHz, 12 GB RAM, NVIDIA GeForce 9800 GX2 video card, Windows 7 64 bits operating system, and two Sensable Phantom Omni haptic devices.

These tests were conducted with 12 volunteers from different areas. None of them are related to medical fields, and all completed undergraduate studies in the areas of mathematics, physics, electronics and computing.

The objective of this test was to evaluate how users appreciate the behavior of SutureHap simulator. The aspect to be evaluated in this test was the graphics of the system, modeling and haptic response.

For this test, the user was located in front of the machine (Fig. 12). A brief introduction about the use of haptic devices was given, and two videos were shown. The first video shows how a surgeon performs a simple suture point in a pigskin. On the other hand, the second video shows the suture procedure applied in the simulator. Samples of this video are shown in figure 11. Additionally, users can watch both videos whenever they want at any time until they are prepared to use SutureHap.

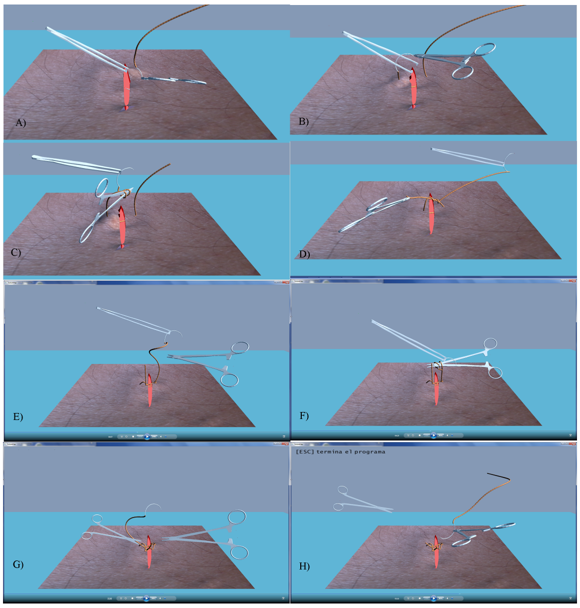

Figure 11: *Graphic appearance of SutureHap and the sequence of steps for a simple suture knot.*

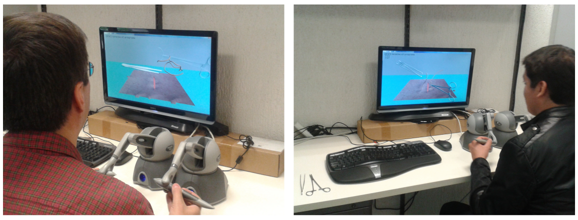

Figure 12: *Users performing usability tests in SutureHap.*

Testing time for each user was approximately one hour, which was divided into two parts. In the first part the participants were instructed to become familiar with the movements of the instruments, the behavior of the thread, behavior of the skin, and the location of the virtual elements. This approach was implemented to adapt users to 3D simulations.

After performing the test, a questionnaire was applied to participants. Results obtained from this evaluations states that 75% felt that the skin is represented correctly, and the force feedback is consistent with the graphical virtual environment. Moreover, 66.6% felt that the thread is properly represented. Consequently, 100% of the participants agreed or strongly agreed that the distribution of the elements is appropriate, the use of the elements in the environment itself is intuitive, and they stated that they will recommend the environment. Finally, from the 100% of users, 50% had used a virtual environment, and whitin that 50%, only 33.3% had used a system with tactile feedback.

To conclude the test, an open field was used to ask users for suggestions or general opinions about SutureHap. Most of the comments points out the difficulty of perception found in 3D environments, and it was also stated that the process was difficult to complete.

5. Conclusion and future work

According to this research, it is considered that virtual learning laboratories represent an important alternative for students to acquire various useful skills is their education. In the field of medicine, they can help resolve the problems in medical training originated by current laws. These laws prohibit the use animals unless same surgical conditions as the ones used for human beings are implemented, which increases significantly costs in training processes.

The incorporation of haptic devices in virtual environments increases greatly the ability of students to promote meaningful learning. They improve the acquisition of skills and competencies, and through them tactile sensations similar to real can be send to users. Moreover, haptic devices decrease costs and risks within the learning process, and they provide support to standardize and recreate surgical situations as often as it is necessary.

Due to the fact that virtual haptic environment possess the ability to use the same hardware and software in various experiments and the flexibility to modify them, development of this kind of laboratories applied in the medical field provides competitive advantages against those that currently exist until the development of this work. There are studies that perform suturing operations; however, they do not integrate all the components that are required for suturing tasks with haptic feedback in a single environment.

Results obtained in this paper allowed the development of a virtual environment for suturing training tasks. This was achieved by using NVIDIA PhysX environment to simulate the skin as a cloth. Since preliminary testing showed that users felt the tactile perception closer to real palpable feelings, necessary adjustments were made to provide a human skin sensation.

According to statements made by users who tested the integrated system, medical instruments used in the environment provided near real behavior in appearance and mobility. Furthermore, the overall system showed a stabilized performance and it responded smoothly during tests. The interaction between the thread, generated by geometrical methods, and the skin, created using physical methods, worked correctly.

Even though the development of this research indicate that modifications to approximate reality are needed as future work, some improvements can be addressed to attain better performance levels. Some of these adjustments are: implementing collision handling and haptic feedback between medical devices and the various elements that are included in the environment; for instance, interaction between tools, interaction between the tools and the table, interaction between tools and fields, etc; however, in order to generate new interactions, the use of haptic devices with full 6 DOF feedback may be necessary.

Current work is focused on applying environment tests to medical students, addition of shadows of objects to improve the perception of the 3D environment, and addition of an intelligent assistant to assess and help students to enhance their surgical performance. Future work will be focused on modification or creation of haptic devices in order to add real forceps on them, evaluation of adding gravity forces, and addition of facilities to modify and increase the degree of difficulty of surgical processes.

6. Acknowledgments

This work was supported by a grant provided by Tecnológico de Monterrey, Campus Ciudad de México, through eLearning research group, and Instituto Politécnico Nacional through Comité Técnico para el Otorgamiento de Becas de Estudio, Apoyos Económicos y Licencias con Goce de Sueldo (COTEBAL). Last but not least, the authors would like to offer our special thanks to Surgeon Gabriela Ugarte Torres and Surgeon Gerardo Aguilar Gómez, who helped and adviced us during the creation of SutureHap and PhD Moises Alencastre-Miranda for their collaboration in the development of surgical tools models.

References

[BDM*04] Basdogan C., De S., Muniyandi M., Hyun K., Srinivasan M. A.: Haptics in minimally invasive surgical simulation and training. *IEEE Computer Graphics and Applications 24*, 2 (Mar. 2004), 56–64. doi:10.1109/MCG.2004.1274062. 2

[BLM04] Brown J., Latombe J., Montgomery K.: Real-time knot-tying simulation. *The Visual Computer*, 1 (2004). URL: http://www.springerlink.com/index/2KF57QANC169YM4Q.pdf. 2, 6

[BS02] Basdogan C., Srinivasan M. A.: Haptic rendering in virtual environments. In *Handbook of virtual environments*, Hale K. S., Stanney K. M., (Eds.). Lawrence Erlbaum Associates, 2002, pp. 117–134. 3

[BSL*02] Brown J., Sorkin S., Latombe J.-C., Montgomery K., Stephanides M.: Algorithmic tools for real-time microsurgery simulation. *Medical Image Analysis 6*, 3 (Sept. 2002), 289–300. doi:10.1016/S1361-8415(02)00086-5. 2

[BTB*04] Berkley J., Turkiyyah G., Berg D., Ganter M., Weghorst S.: Real-time finite element modeling for surgery simulation: an application to virtual suturing. *IEEE transactions on visualization and computer graphics 10*, 3 (2004), 314–25. doi:10.1109/TVCG.2004.1272730. 8

[CCP12] Choi K.-S., Chan S.-H., Pang W.-M.: Virtual suturing simulation based on commodity physics engine for medical learning. *Journal of medical systems 36*, 3 (June 2012), 1781–93. doi:10.1007/s10916-010-9638-1. 2

[CDA99] Cotin S., Delingette H., Ayache N.: Real-time elastic deformations of soft tissues for surgery simulation. *IEEE Transactions on Visualization and Computer Graphics 5*, 1 (1999), 62–73. doi:10.1109/2945.764872. 2

[CMJ11] Coles T. R., Meglan D., John N. W.: The Role of Haptics in Medical Training Simulators: A Survey of the State of the Art. *IEEE Transactions on Haptics 4*, 1 (Jan. 2011), 51–66. doi:10.1109/TOH.2010.19. 2

[LC06] Lian L. L., Chen Y. H.: Haptic Surgical Simulation: An Application to Virtual Suture. *Computer-Aided Design and Applications 3*, 1-4 (Jan. 2006), 203–210. doi:10.1080/16864360.2006.10738457. 6

[LMGC02] Lenoir J., Meseure P., Grisoni L., Chaillou C.: Surgical Thread Simulation. In *ESAIM: Proceedings* (2002), vol. 12, pp. 102–107. 2

[LMGC04] Lenoir J., Meseure P., Grisoni L., Chaillou C.: A suture model for surgical simulation. *International Symposium on Medical Simulation 3078* (2004), 105–113. doi:10.1007/978-3-540-25968-8_12. 2

[MDH03] Meseure P., Davanne J., Hilde L.: A physically-based virtual environment dedicated to surgical simulation. In *Surgery Simulation and Soft Tissue Modeling*, Ayache N., Delingette H., (Eds.), no. Cd. Springer Berlin Heidelberg, Juan-Les-Pins, France, 2003, pp. 38–47. doi:10.1007/3-540-45015-7_4. 2

[Med13] Medical I.: Company Overview, 2013. URL: http://www.immersion.com/about/company-overview/index.html. 2

[MEN13] MENTICE: About us, 2013. URL: http://www.mentice.com/about-us/. 2

[MYS*06] Miyazaki K., Yoshimoto Y., Sakaguchi M., Sano A., Fujimoto H.: Development of Suture Simulator That Can Express 3D Deformation of Wounds and Those Vicinities. In *2006 9th International Conference on Control, Automation, Robotics and Vision* (2006), IEEE, (Ed.), IEEE, pp. 1–6. doi:10.1109/ICARCV.2006.345224. 2

[NVi12] NVidia: PhysX, 2012. URL: http://www.geforce.com/hardware/technology/physx. 2

[OSO*07] Oshima N., Solis J., Ogura Y., Hatake K., Takanishi A.: Development of the suture/ligature training system WKS-2 designed to provide more detailed information of the task performance. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Oct. 2007), IEEE, pp. 58–63. doi:10.1109/IROS.2007.4399351. 2

[SCB04] Salisbury K., Conti F., Barbagli F.: Survey - Haptic rendering: introductory concepts. *IEEE Computer Graphics and Applications 24*, 2 (Mar. 2004), 24–32. doi:10.1109/MCG.2004.1274058. 2, 3

[SCI13a] SCIENCE S.: About us, 2013. URL: http://www.surgical-science.com/surgical-science/about-us/. 2

[SCI13b] SCIENCE S.: LapSim, 2013. URL: http://www.surgical-science.com/portfolio/haptic-system/. 2

[TEC13] TECHNOLOGIES R.: About Reachin Technologies, 2013. URL: http://www.reachin.se/companyinfo/. 2

[WZM*01] Webster R. W., Zimmerman D. I., Mohler B. J., Melkonian M. G., Haluck R. S.: A prototype haptic suturing simulator. *Studies in health technology and informatics 81* (Jan. 2001), 567–9. URL: http://www.ncbi.nlm.nih.gov/pubmed/11317811. 2, 3