AUGMENTED INTERACTION FOR CUSTOM-FIT PRODUCTS BY MEANS OF INTERACTION DEVICES AT LOW COSTS

Direttore della Scuola: Ch.mo Prof. Paolo Colombo
Coordinatore d'indirizzo: Ch.mo Prof. Enrico Savio
Supervisore: Ch.ma Prof.ssa Caterina Rizzi

Dottorando: Andrea Vitali

Table of Contents

Abstract

This Ph.D thesis refers to a research project that aims at developing an innovative platform to design lower limb prosthesis (both for below and above knee amputation) centered on the virtual model of the amputee and based on a computer-aided and knowledge-guided approach. The attention has been put on the modeling tool of the socket, which is the most critical component of the whole prosthesis. **The main aim has been to redesign and develop a new prosthetic CAD tool, named SMA² (Socket Modelling Assistant²) exploiting a low-cost IT technologies** (e.g. hand/finger tracking devices) and making the **user's interaction as much as possible natural and similar to the hand-made manipulation. The research activities** have been carried out in **six phases** as described in the following.

First, limits and criticalities of the already available modeling tool (namely SMA) have been identified. To this end, the first version of SMA has been tested with Ortopedia Panini and the orthopedic research group of Salford University in Manchester with real case studies. Main criticalities were related to: (i) automatic reconstruction of the residuum geometric model starting from medical images, (ii) performance of virtual modeling tools to generate the socket shape, and (iii) interaction mainly based on traditional devices (e.g., mouse and keyboard).

The second phase lead to the software reengineering of SMA according to the limits identified in the first phase. The software architecture has been re-designed adopting an object-oriented paradigm and its modularity permits to remove or add new features in a very simple way. The new modeling system, i.e. SMA2, has been totally implemented using open source Software Development Kit-SDK (e.g., Visualization ToolKit VTK, OpenCASCADE and Qt SDK) and based on low cost technology. It includes:

- A new **module** to **automatically reconstruct the 3D model of the residual limb** from MRI images. In addition, a new procedure based on low-cost technology, such as Microsoft Kinect V2 sensor, has been identified to acquire the 3D external shape of the residuum.

- An **open source software library**, named **SimplyNURBS**, for NURBS modeling and specifically used for the automatic reconstruction of the residuum 3D model from medical images. Even if,

SimplyNURBS has been conceived for the prosthetic domain, it can be used to develop NURBS-based modeling tools for a range of applicative domains from health-care to clothing design.

- A module **for mesh editing** to emulate the hand-made operations carried out by orthopedic technicians during traditional socket manufacturing process. In addition several virtual widgets have been implemented to make available virtual tools similar to the real ones used by the prosthetist, such as tape measure and pencil.

- A **Natural User Interface (NUI)** to allow the interaction with the residuum and socket models using hand-tracking and haptic devices.

- A module to generate the **geometric models for additive manufacturing** of the socket.

The third phase concerned the study and design of augmented interaction with particular attention to the **Natural User Interface** (NUI) for the use of hand-tracking and haptic devices into SMA^2. The NUI is based on the use of the Leap Motion device. A set of gestures, mainly iconic and suitable for the considered domain, has been identified taking into account ergonomic issues (e.g., arm posture) and ease of use. The modularity of SMA^2 permits us to easily generate the software interface for each device for augmented interaction. To this end, a software module, named **Tracking plug-in**, has been developed to automatically generate the source code of software interfaces for managing the interaction with low cost hand-tracking devices (e.g., Leap Motion and Intel Gesture Camera) and replicate/emulate manual operations usually performed to design custom-fit products, such medical devices and garments. Regarding haptic rendering, two different devices have been considered, the Falcon Novint, and a haptic mouse developed in-house.

In the **fourth phase, additive manufacturing technologies** have been investigated, in particular FDM one. 3D printing has been exploited in order to permit the creation of trial sockets in laboratory to evaluate the potentiality of SMA^2. Furthermore, research activities have been done to study new ways to design the socket. An innovative way to build the socket has been developed based on multi-material 3D printing. Taking advantage of flexible material and multi-material print possibility, new 3D printers permit to create object with soft and hard parts. In this phase, issues about infill, materials and comfort have been faced and solved considering different compositions of materials to re-design the socket shape.

In the **fifth phase** the implemented solution, integrated within the whole prosthesis design platform, has been tested with a transfemoral amputee. Following activities have been performed:

- 3D acquisition of the residuum using MRI and commercial 3D scanning systems (low cost and professional).

- Creation of the residual limb and socket geometry.

- Multi-material 3D printing of the socket using FDM technology.

- Gait analysis of the amputee wearing the socket using a markerless motion capture system.

- Acquisition of contact pressure between residual limb and a trial socket by means of Teskan's F-Socket System.

Acquired data have been combined inside an ad-hoc developed application, which permits to simultaneously visualize pressure data on the 3D model of the residual lower limb and the animation of gait analysis. Results and feedback have been possible thanks to this application that permits to find correlation between several phases of the gait cycle and the pressure data at the same time. Reached results have been considered very interested and several tests have been planned in order to try the system in orthopedic laboratories in real cases. The reached results have been very useful to evaluate the quality of SMA2 as a future instruments that can be exploited for orthopedic technicians in order to create real socket for patients. The solution has the potentiality to begin a potential commercial product, which will be able to substitute the classic procedure for socket design.

The **sixth phase** concerned the **evolution of SMA^2 as a Mixed Reality environment**, named Virtual Orthopedic LABoratory (**VOLAB**). The proposed solution is based on low cost devices and open source libraries (e.g., OpenCL and VTK). In particular, the hardware architecture consists of three Microsoft Kinect v2 for human body tracking, the head mounted display Oculus Rift SDK 2 for 3D environment rendering, and the Leap Motion device for hand/fingers tracking. The software development has been based on the modular structure of SMA^2 and dedicated modules have been developed to guarantee the communication among the devices. At present, two preliminary tests have been carried out: the first to verify real-time performance of the virtual environment and the second one to verify the augmented interaction with hands using SMA^2 modeling tools. Achieved results are very promising but, highlighted some limitations of this first version of VOLAB and improvements are necessary. For example, the quality of the 3D real world reconstruction, especially as far as concern the residual limb, could be improved by using two HD-RGB cameras together the Oculus Rift.

To conclude, the obtained results have been evaluated very interested and encouraging from the technical staff of orthopedic laboratory. SMA^2 will made possible an important change of the process to design the socket of lower limb prosthesis, from a traditional hand-made manufacturing process to a totally virtual knowledge-guided process. The proposed solutions and results reached so far can be exploited in other industrial sectors where the final product heavily depends on the human body morphology. In fact, preliminary software development has been done to create a virtual environment for clothing design by starting from the basic modules exploited in SMA^2.

Keywords: Lower limb prosthesis, Socket design and modeling, Mixed reality, Augmented interaction, Low cost hand-tracking devices, Additive manufacturing.

Riassunto

La presente tesi di dottorato è stata sviluppata nell'ambito di un progetto di ricerca che ha come obiettivo lo sviluppo di un framework innovativo per la progettazione di protesi per arto inferiore (per amputazioni sia transfemorali sia transtibiali) che permetta di gestire l'intero processo in un unico ambiente integrato dove ciascuna attività è supportata in modo diretto dalla conoscenza di dominio specifica. L'attenzione è stata posta sullo sviluppo di un'applicazione per la progettazione dell'invaso, il componente più critico dell'intera protesi. L'obbiettivo principale del lavoro di tesi è stato, quindi, la reingegnerizzazione e lo sviluppo di un nuovo sistema CAD prostetico, denominato SMA[2] (Socket Modelling Assistant[2], basato sull'impiego di tecnologie innovative a basso costo, quali ad esempio periferiche per il tracciamento di mani e dita, per permettere l'interazione uomo-macchina il più naturale possibile.

Le attività di ricerca sono state organizzate in **sei fasi principali** come segue.

Durante la **prima fase**, i limiti e le criticità della prima versione del sistema di modellazione (denominato SMA) sono stati identificati. A tal fine, SMA è stato testato con il personale tecnicno dell'ortopedia Panini e con i ricercatori del gruppo di ricerca in ortopedia della Salford University di Manchester utilizzando casi di studio reali. Sono state identificate le seguenti criticità a: (i) difficoltà nella ricostruzione del modello 3D dell'arto amputato partendo da immagine mediche, (ii) prestazioni delle modalità di modellazione 3D dell'invaso non adeguate e (iii) interazione con l'ambiente virtuale basata esclusivamente su periferiche classiche come tastiera e mouse e non era possibile emulare le tipiche operazioni di modifica dell'invaso.

La **seconda fase** ha riguardato la riprogettazione del sistema di modellazione SMA in funzione dei limiti identificati nella prima fase. L'architettura software è stata riprogettata seguendo i paradigmi della progettazione orientata agli oggetti e la sua modularità permette di rimuovere o aggiungere nuovi moduli in modo semplice. Il nuovo sistema di modellazione, SMA[2], è stato totalmente implementato usando Software Development Kit (SDK) con licenza open-source (in particolare Visualization ToolKit VTK, OpenCASCADE e Qt SDK) e basato su tecnologia a basso costo. Il sistema comprende:

- Un nuovo modulo per la ricostruzione automatica dell'arto residuo partendo da immagini MRI. In aggiunta, una nuova procedura basata su tecnologia a basso costo, come Microsoft Kinect v2, è stata identificata ed utilizzata per acquisire il modello 3D dell'arto residuo.

- Una libreria software open-souce, denominata SimplyNURBS, per la modellazione di superfici NURBS e specificatamente utilizzata per la ricostruzione automatica dell'arto amputato partendo dalle imamgini MRI. Anche se SimplyNURBS è stata sviluppata nell'ambito della progettazione di protesi per arti inferiori, essa può essere usata all'interno di altre applicazioni, sia in ambito medico sia in ambito industriale per quei prodotti la cui modellazione richiede l'utilizzo di modelli NURBS, quali ad esempio capi di abbigliamento.

- Un modulo per la modifica di mesh triangolari per emulare le operationi manuali effettuate dai tecnici ortopedici durante il processo di sviluppo prodotto dell'invaso. Inoltre, sono stati sviluppati strumenti di modellazione virtuali per simulare quelli utilizzati dal tecnico ortopedico, quali ad esempio il metro per misurare l'arto residuo o la matita per evidenziare le zone critiche dell'invaso.

- Una Natural User Interface (NUI) che permette di interagire con i modelli dell'arto residuo e dell'invaso attraverso periferiche di hand-tracking.

- Un modulo per generare il modello geometrico dell'invaso per la realizzazione dello stesso mediante tecnologie di produzione additive.

La **terza fase** ha riguardato lo studio e la progettazione dell'interazione aumentata con particolare attenzione allo sviluppo di una Natural User Interface per l'uso di periferiche di hand-tracking e di force-feedback. La NUI è basata sull'uso della periferica di hand-tracking Leap Motion. Un insieme di gesti, principalmente iconici e adatti al dominio considerato, sono stati indentificati tenendo in considerazione vincoli ergonomici (per esempio, la postura delle braccia) e la facilità di utilizzo. La modularità di SMA^2 ha permesso di generare in modo semplice un'interfaccia software per ogni dispositivo di hand-tracking preso in considerazione. Inoltre, è stato sviluppato un modulo software, denominato Tracking plug-in, che permette di generare in automatico il codice sorgente per la creazione di interfacce per l'interazione con dispositivi di hand-tracking, come ad esempio Leap Motion e Intel Gesture Camera. In questo modo è possibile replicare/emulare le operazioni che vengono eseguite durante la progettazione di prodotti custom-fit, quali protesi e capi di abbigliamento. Per quanto riguarda i dispositivi per l'interazione tattile, sono stati considerati il Novint Falcon e un mouse aptico sviluppato ad hoc.

Durante la **quarta fase**, sono state studiate ed analizzate le tecnologie di Additive Manufacturing, ponendo particolare attenzione alla tecnologia FDM. Tale tecnologia è stata utilizzata per realizzare l'invaso di un paziente progettato con SMA^2 e verificarne le potenzialità. Inoltre, sono state condotte attività di ricerca per valutare nuove modalità di progettazione dell'invaso. In particolare, è stata considerata la possibilità di progettare e realizzare l'invaso utilizzando una stampante FDM multi materiale. Sfuttando possibilità di

utilizzare materiali flessibili e la stampa di oggetti con materiali diversi, è possibile realizzare un invaso composto da parti flessibili e parti più rigide. A tal fine sono stati presi in considerazione aspetti, quali le modalità di riempimento (infill) durante la stampa 3D, i materiali per la realizzazione delle diverse parti ed il

Durante la **quinta fase,** il sistema implementato, integrato all'interno dell'intera piattaforma per la progettazione di protesi, è stato sperimentato con un paziente con amputazione transfermorale. Sono state svolte le seguenti attività:

- Acquisizione dell'arto amputato mediante Risonanza Magnetica e sistemi commerciali di acquisizione 3D (sia a basso coste sia professionali).
- Creazione della geometria 3D dell'arto residuo e dell'invaso.
- Stampa 3D multi materiale dell'invaso utilizzando la tecnologia FDM.
- Analisi della camminata dell'amputato mediante un sistema di acquisizione del movimento.
- Acquisizione delle pressioni di contatto tra invaso e arto residuo mediante il sistema Teskan F-Socket System.

Mediante un'applicazione sviluppata ad hoc, è stato possibile rimappare sul modello 3D dell'invaso i valori di pressione acquisiti mediante mappe di colore e simultaneamente visualizzare l'animazione della camminata acquisita. Questa applicazione permette di identificare eventuali correlazioni tra le differenti fasi del ciclo della camminata ed i valori di pressione acquisiti. I risultati raggiunti si sono rilevati d'interesse e sono stati pianificati ulteriori test per validare il sistema con altri pazienti. I risultati ottenuti hanno, inoltre permesso di valutare le potenzialità di SMA^2 ed il suo impiego presso laboratori ortopedici. La soluzione ha il potenziale per diventare un prodotto commerciale e quindi sostituire la procedura manuale seguita per progettazione dell'invaso.

La **sesta fase** ha riguardato l'evoluzione di SMA^2 come ambiente di Mixed Reality. La soluzione proposta, denominata Virtual Othopedic Laboratory (VOLAB), è sempre basata su dispositivi a basso costo e librerie software open source (OpenCL e VTK). In particolare, l'architettura hardware comprende tre Microsoft Kinect v2 per il tracciamento del corpo umano, il sistema di visione 3D Oculus Rift SDK 2 per il rendering dell'ambiente virtuale ed il dispositivo Leap Motion per il tracciamento della mani/dita. Lo sviluppo software è basato sulla struttura modulare di SMA^2 e moduli dedicati che sono stati sviluppati per garantire la comunicazione tra i vari dispositivi coinvolti. Attualmente, sono stati svolti due test preliminari: il primo per verificare le prestazioni real-time dell'ambiente virtuale ed il secondo per verificare l'interazione con i dispositivi di hand-tracking durante l'utilizzo degli strumenti di modellazione virtuale messi a disposizione da SMA^2. I risultati ottenuti sono molto promettenti, ma sono necessari ulteriori miglioramenti. Per esempio, la qualità della ricostruzione dell'ambiente virtuale, in particolare del modello dell'arto residuo, potrebbe essere migliorata utilizzando due camere HD-RGB in aggiunta all'Oculus Rift.

Per concludere, i risultati ottenuti sono stati considerati molto interessanti ed incoraggianti dallo staff tecnico dei laboratori ortopedici. SMA^2 renderà possibile un importante cambiamento del modo di progettare un invaso per protesi d'arto inferiore, da un processo tradizionalmente manuale ad un nuovo processo basato su approccio virtuale e l'impiego di strumenti computer-aided. Le soluzioni proposte ed i risultati raggiunti possono essere utilizzati anche in altri settori industriali dove il prodotto finale dipende fortemente dalla morfologia del corpo umano. Infatti, sono stati sviluppati alcuni moduli software per la progettazioni di capi di abbigliamento partendo dai moduli base sviluppati per SMA^2.

Parole chiave: Protesi per arto inferiore, Progettazione e modellazione dell'invaso, Mixed Reality, Interazione aumentata, Dispositivi di hand-tracking a basso costo, Additive manufacturing.

Chapter 1: Introduction

During last years, there has been an increasing demand of IT solutions being able to virtually design products. However, in some domains, the level of diffusion is still limited, especially when the product requires a high level of customization and represents the interface with the human body or parts of it. An example is artificial prostheses that have to be designed according to the shape of the specific anatomical area.

This Ph.D thesis presents an innovative prosthetic CAD tool, named Socket Modelling Assistants[2]-SMA[2]. It has been developed to support the design of lower limb prosthesis, with particular attention to the socket, which is the custom-fit component and the most critical one. In fact, the correct design of the socket shape is responsible for the good functionality of the whole prosthesis.

The main aim has been to develop a new prosthetic CAD tool based on a fully computer-aided and knowledge guided approach making available a set of interactive and virtual devices that permit to replicate and emulate the hand-made operations performed by technicians during the traditional manufacturing process. The development of SMA[2] has been based on the re-design of a previous version to automate as much as possible the modeling procedure and allow augmented interaction.

The PhD thesis has been developed under the framework of the I4BIO Project (Innovation for Bioengineering Project) co-funded by Fondazione Cariplo and University of Bergamo. Knowledge acquisition, experimentation and validation of the framework have been realized with the collaboration of Ortopedia Panini in Milan, a high-qualified orthopedic laboratory.

The present document is organized as follows:

Chapter 2 introduces the considered context and the platform devoted to the design of lower limb prosthesis totally based on low cost and open source IT technology.

Chapter 3 presents the SMA[2] system that has been the main subject of the PhD Thesis and the kernel of the whole platform. SMA[2] architecture and related developed software tools are described as well as the

socket modeling procedure. SMA[2] encapsulates the knowledge of the orthopedic technicians and a set of automatic and semi-automatic tools to assist the technicians.

Chapter 4 introduces the acquisition of the 3D human body with particular attention to the residual limb around which the socket is designed. Two approaches have been investigated. The first one exploits MRI (Magnetic Resonance Imaging) images to obtain a detailed 3D model of the residuum including both external (skin) and internal parts (e.g., bones). The second approach is based on the use of a low cost 3D scanner to acquire the external shape of the residual limb in a simple and fast way.

Chapter 5 concerns the use of augmented interaction for socket design. Hand tracking devices have been considered to allow the interaction by hands with the 3D models of both the residuum and socket to replicate the operations manually performed by the prosthetist. In fact, traditionally, the technician uses continuously her/his hands to shape the socket positive cast around which the physical prototype is thermoformed. To this end, a natural software interface NUI has been developed.

Chapter 6 concerns with the realization of the physical socket by means of additive manufacturing technology, in particular, a multi-material 3D printer. This required the development of dedicated software modules to export 3D geometric model for multi-material manufacturing.

Chapter 7 describes the experimentation carried out with an amputee. The experimentation is not only related to SMA but the whole prosthesis design platform introduced in Chapter 2. An application has been developed in order to combine each source of data in a unique environment and identify the possible correlation.

Chapter 8 presents an evolution of SMA as a Mixed Reality environment, named Virtual Orthopedic LABoratory (VOLAB). The proposed solution, based on the modular structure of SMA, exploits open source libraries and low cost devices, such as Microsoft Kinect, Oculus Rift for 3D environment visualization and Leap Motion device for hand/fingers tracking.

Chapter 9 draws the conclusions and future developments.

Chapter 2: The Context

The chapter provides an overview of the context and the research project the PhD thesis refers to. Firstly, the traditional manufacturing process of socket design is described in order to introduce the thematic area of the whole thesis work. Then, the virtual platform aimed at integrating low cost technologies to design, test and manufacture lower limb prosthesis is described. Finally, main limits and criticalities of first version of the 3D prosthetic CAD systems are discussed.

2.1 Lower limb prosthesis

In this PhD thesis, we considered both below knee (also called "transtibial") and above knee prosthesis (also called "transfemoral"), realized with the state of the art components in order to obtain the maximum comfort and usability for the amputees.

Last generation of modular components for this kind of prosthesis are: the liner, the socket, the lock, the knee (only for transfemoral), the pipe or the double joint, and a foot, beyond a great variety of adapters. In Figure 1 portrays the scheme of modular prosthesis for both above and below knee amputation. Furthermore, the liner (see example in Figure 2) rolls onto the residual limb and is then inserted and locked into the socket. This is the last generation of suspension systems and can provide improved cosmetics, cushion the residual limb, reduce shear between the residual limb and the socket, and minimize pistoning of the residual limb in the socket. Heat buildup, sensitive skin problems, and decreased proprioception can be drawbacks of this suspension system. Most of these products are standard components, which are chosen from commercial catalogues on the base of amputee characteristics, apart the liner, which can be both standard and custom fit, and the socket, which is always manufactured expressly in relation to the specific anatomy of the patient. In particular the standard liner is normally well fitting on most of residual limbs, only a small number of patients with particular and problematic stump anatomy need a customized liner.
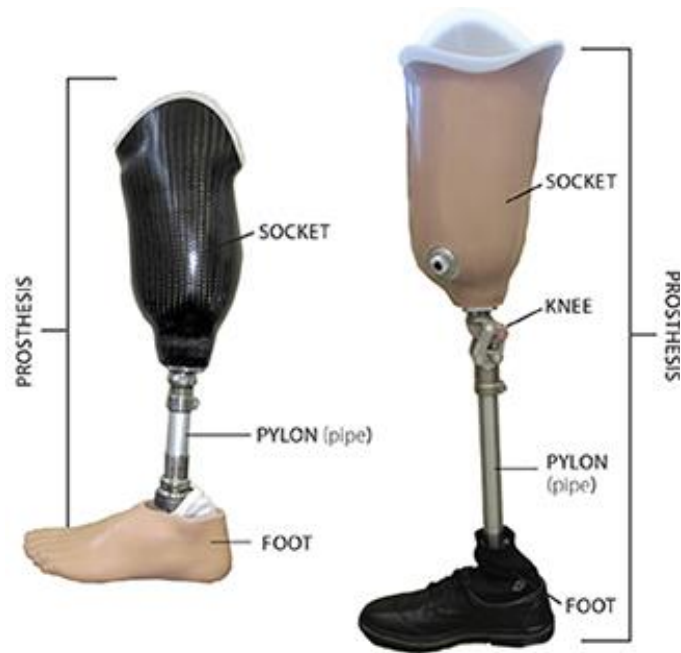
Figure 1: Basic Schema of a lower limb prothesis for above knee and below knee amputation.
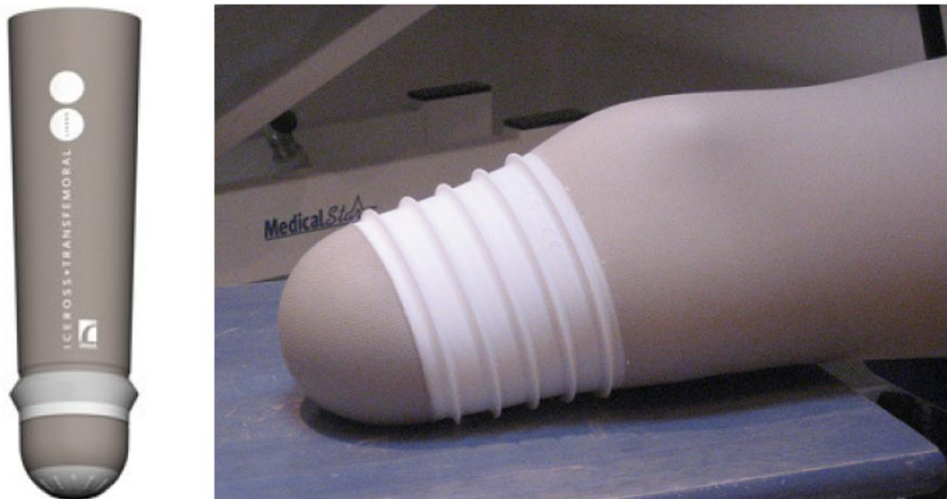
Figure 2 Standard commercial liner for below knee amputation.

Instead, the socket has to be specifically designed for each patient, since it is the interface between the residual limb and the prosthesis. It must not only protect the residual limb, but also appropriately transmit the forces associated with standing and ambulation.

2.2 Traditional socket development process [1]

In the traditional manufacturing process, the technician uses continuously her/his hands to reach the optimal socket shape. Initially, s/he makes an evaluation of the amputee and creates a negative cast manipulating by hands plaster patches directly on patient's residuum (Figure 3).

Figure 3: The creation of the negative model made by an orthopedic technician.

Then, s/he realizes the positive model that is modified by adding and removing chalk in specific zones and according to stump measurements and patient's characteristics (e.g., tonicity).

Three main manual operations are performed: initial plaster circumference reduction, identification and marking of critical zones, and critical zones manipulation. The first one consists in reducing the positive plaster cast according to the stump conditions. For example, the socket must be more fitting for young or recently amputated patients, while for elderly patients it needs to be more comfortable and not too much close fitting to allow an easier deambulation or physical therapy. In the second operation the technician marks with a pencil on the positive cast the areas that have to be modified (Figure 4.a). The third one consists in adding or removing materials in highlighted critical zones (Figure 4.b). The following critical areas, different for transtibial and transfemoral amputee, can be identified:

- *Load zones* where there are no bony protuberances or tendons. In this case the plaster has to be removed in order to have a socket tighter and also self-supporting.
- *Off-load zones* where there are bony protuberances or tendons. In this case the technician adds material on the positive plaster cast since in that zones the socket does not have to press the stump and be quite loose.

After this, a thermoformable check socket is manufactured directly on the modified positive model and tested with the amputee (Figure 5). If necessary, modifications are marked on the positive model to realize a more comfortable and well-fitting final socket.

Finally, the definitive socket is realized and all the prosthesis components are assembled for the final static set-up.

Summarizing, the socket is manufactured following a hand-made procedure and the final shape heavily depends on the prosthetist's skills and experience according to residual limb morphology and conditions.
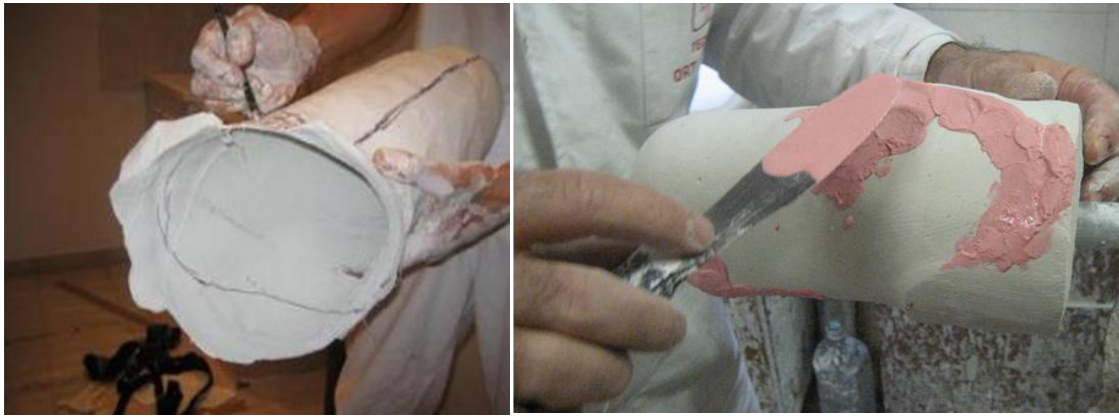
<div align="center">(a)   (b)</div>

Figure 4: Definition of load and off-load zones (a) and the adding/removing operation along the positive model of the residual limb (b).

Figure 5: Thermoformed socket of a socket for above knee amputation.

2.3 State of the art of prosthetic CAD SYSTEMS

In Italy the vast majority of prosthetists are small businessmen or employees of small businesses and the design and manufacture of a socket is almost a hand-made activity carried out by skilled technician.

On the market we can find some commercial CAD/CAM systems [2]–[7], which can assist the orthopedic technicians in some steps of the development process.

In the following, the most diffused CAD/CAM prosthetic systems available on market are described.

2.3.1. Infinity CAD systems: AutoScanner & AutoSculpt [5]

AutoScanner instantly acquires prosthetic 3-D surfaces by gathering measurements made by smoothly sweeping a handheld laser scanning wand. Practitioners can scan AK's / BK's / AFO's / Body Jackets / Head / Hands. AutoScanner is built on reflection technology. A laser beam emitted through the handheld laser wand determines the scanning surface coordinate position. The computed points of data (point cloud) form a 3-D image of the scanned prosthetic shape. Thanks to the cutting edge, laser reflection technology used in AutoScanner, the collection of scanned surface data points will be hundreds of thousands per second,

17

ensuring that the scanning process is fast reliable and accurate. AutoScanner software uses various scientific formulas to calculate non-scanned data points, hence it forms a fine, high resolution 3-D prosthetic image. AutoSculpt is computer aided design software exclusively developed for modifying prosthetic 3-D images. User-friendly features of AutoSculpt enable prosthetist/practitioners to implement all necessary changes to their 3-D images according to the weight bearing specification of their patients. AutoSculpt prosthetic modification software has state-of-the-art features, which enable practitioners to modify their patients' residual limb 3-D images accurately and efficiently. The following features demonstrate its ability:

- Designer utilities enable Above Knee/ Below Knee/ AFO/ Spinal Jacket modifications.
- Curves creation and modifications for depressions and elevations.
- Area creation, modification and transformation.
- Global and segment volume modifications.
- General utilities to make changes to resolution. End cap utilities, slice addition and subtractions.
- Volume reissue feature to change circumferential volume.
- BK design tools, with A-P, M-L modification.
- AK, spinal jacket, AFO – Design library, Foot, Insole modification.
- A pattern utility allows practitioners to create custom pattern shape libraries. Such patterns are reusable and can be applied to any other prosthetic 3-D images.
- Using utilities like volume/ area/ length, practitioners can change a prosthetic image global or segmental properties.
- An image smoothing feature allows fine resolution adjustments to the prosthetic image.
- In addition to the 3-D image view, horizontal and vertical cross section view options enable practitioners to view changes from all angles.
- Modifications are restorable. Practitioners can undo changes or redo.
- Anterior/ Posterior/ Medial/ Lateral views and freehand rotate options of help practitioners make modifications at the desired surface of the prosthetic image.
- Apart from these utilities, AutoSculpt has many additional features that enable successful 3-D prosthetic image modification tool-kits, including coloring tools, visualization tools, ambient lighting, an orthographic camera, a headlight property and other features.

An example of the application is showed in Figure 6.

Figure 6: AutoScanner (left) and AutoSculpt (right).

2.3.2. Biosculptor, BioScanner, BioShape Software & DSS Digital Socket System [2]

Scan a patient in 10 seconds. Both patients and referral sources will be amazed by the efficiency, speed, and advanced technology of the BioScanner™. Since registration marks are not required, there is no preparatory work involved. You simply click and go. Change the mode to Optical Stylus and instantly place landmarks and alignment marks as you scan. Any portion of the body may be directly scanned for orthoses or prostheses. There is no size limitation. A miniature transmitter is placed on the body to accommodate for any movement. The BioScanner™ is able to image negative and positive models, allowing you to use the clinical techniques required for each patient. With scan-through-glass technology, you may position the body horizontally for a TLSO or utilize a weight bearing table for AFOs or foot orthotics. The BioScanner™ automatically corrects the refraction. The precision of the BioScanner™ is as impressive as its speed. Capture shapes to an accuracy of 0.178mm. To further improve quality, the software streamlines the final scan to equally distribute the scan sweeps. You receive the most accurate scan available without added processing time. The BioShape Manipulation Software is the most powerful CAD software available for O&P. Designed by Prosthetists and Orthotists, it was developed specifically for the clinician. With BioShape, you are able to modify upper and lower extremity prostheses, spinal orthoses, lower and upper extremity orthoses, pediatric shapes, cranial helmets, liners and face masks. It seamlessly integrates with all of the Biosculptor® products. DSS™ Digital Socket System is the next generation of socket-by-numbers. Our unique software utilizes oblique, transverse and circumferential measurements, we use the crucial anatomical data. This means your test sockets will fit more precisely, saving precious time. We can also template your personal socket design for your exclusive use. Plus, you may receive your virtual model for approval. Now, you can provide CAD/CAM test sockets to your patients with no investment. An example is shown in Figure 7 and Figure 8.
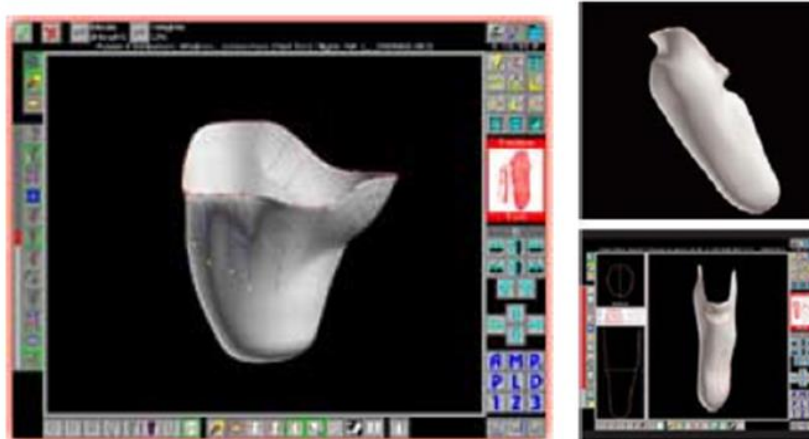
Figure 7: BioShape Software.

Figure 8: BioScanner (left) and DSS (right).

2.3.3. Rodin4D: FastScan 3D & Software [4]

Easy to handle, fast, accurate, the Fastscan no-contact 3D digitizer enables you to digitize freely and easily the most complex forms. Indispensable for the realization of non-symmetrical forms, with it you can:

1. Digitize your patients in order to work directly on their form afterwards.
2. Create accurate and realistic library forms.
3. Digitize your plaster casts before destroying them.

Rodin4D now enables you to add control measurements on your 3D shape. In order to have better control of the measurements on your shape, a list is available allowing you to add various measurements. These measurements are displayed at the top left of the 3D scene and are modified dynamically during your rectification.

- 3D Line allows you to measure a linear distance between 2 points on the shape.
- Perimeter on section measures the circumference from 3 points on the shape.

- Partial volume measures the volume of the section of the selected form. The unit of measurement for partial volume is now dm3.

An example is shown in Figure 9.
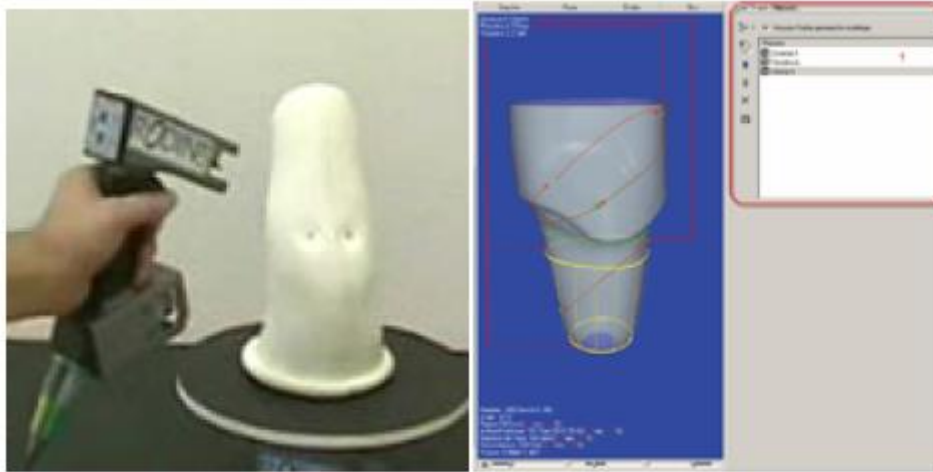
Figure 9: FastScan 3D (left) and Software (right).

2.3.4. Össur: Design TF and Design TT [7]

Design TF is a software, which allows the orthopedic technician to realize a transfemoral check socket using only some measurement taken from the residual limb. The software creates a 3D model of the check socket and allows also the technician to modify its shape. All the data are then sent by email to Össur, which realizes the socket. Design TT is a system composed by a digital camera, which takes pictures of the residual limb and calculates circumferences and volume of the stump, and a software, which using these data creates a 3D model of a transtibial check socket and allows also the technician to modify its shape. All the data are then sent by email to Össur, which realizes the socket. An example is shown in Figure 10.
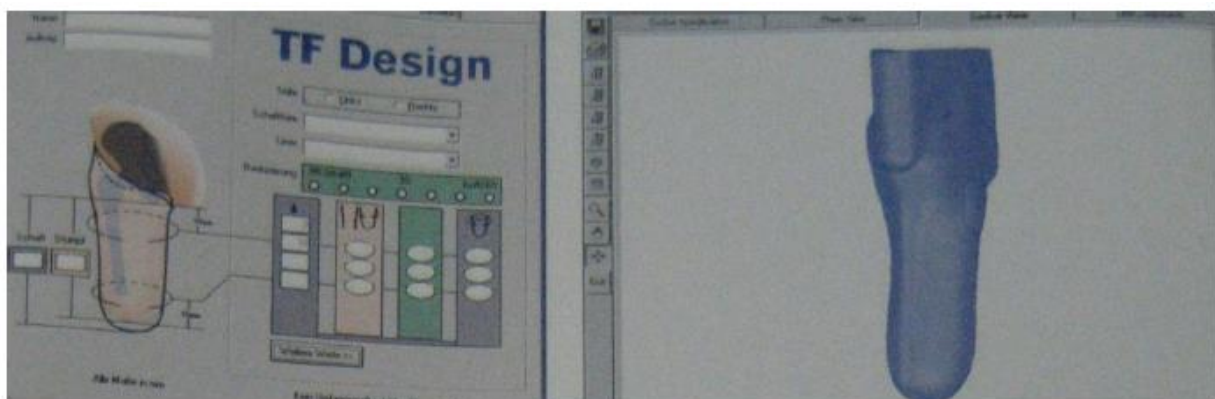
Figure 10: Design TF (left) and Design TT (right).

2.3.5. Orten: ComfORTAC, Orten PIX & CAD Software [6]

ComfORTAC acquires 360° trunk and lower limbs measurements by structured light projection. This system also enables morphological reference points to be recorded from their direct position on the patient. These reference points are used by our CAD software for the design of orthopedic devices. Another OrtenPIX protocol is based on measurements only. The file is quickly computed in the desired shape and does not require further modification. This technique is quickly mastered, and the time saved is greatly maximized. This method is particularly adapted to Above the Knee prostheses, mattresses, seating and standing systems. A transtibial socket is easy to design from the 3D modeling of the stump. In few clicks, you can apply compression and create build-up areas. The design of a cosmetic cover is also straightforward. It can be created as the symmetric of the healthy leg. In the case of leg orthosis, more tools (such as flexion) are available. Examples are shown in Figure 11 and *Figure* 12.
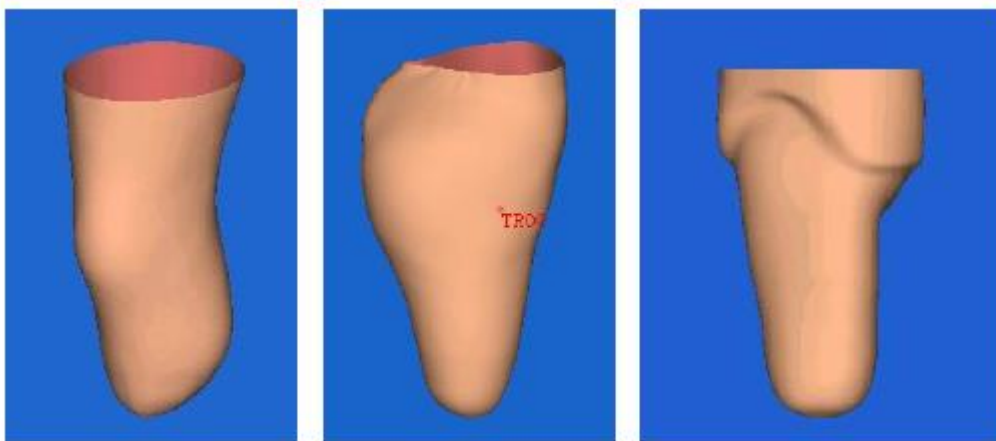
Figure 11: ComfORTAC (left and center) and Orten PIX (right).
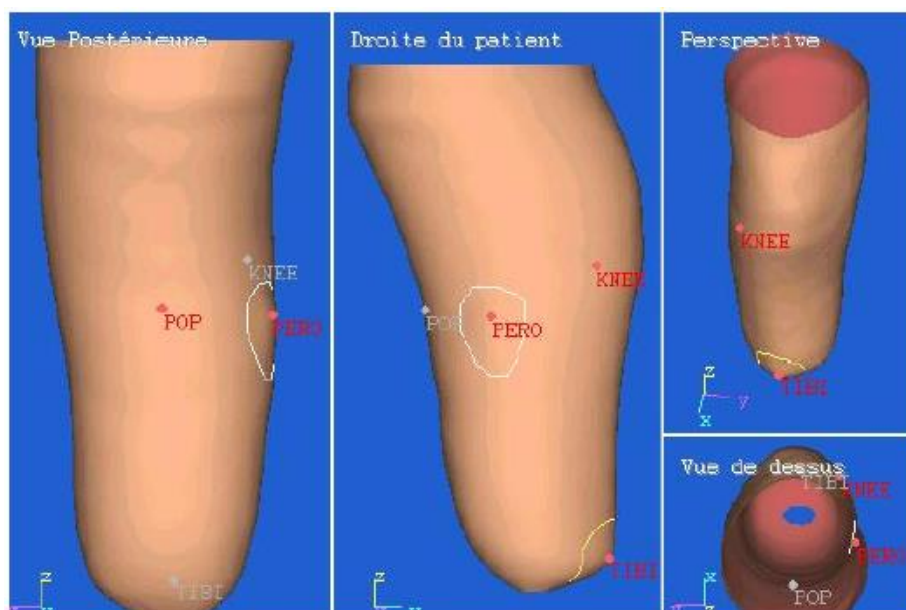
Figure 12*: CAD Application.*

2.3.6. Vorum: Canfit P&O System [3]

The CanfitTM P&O CAD/CAM System provides an integrated suite of tools for acquiring shape data, designing and modifying shapes, and carving positive models. Tailored specifically for the prosthetics and orthotics industry, the components are designed to work most effectively as a single unit. A flexible configuration of hardware and software enables you to assemble a system, which meets your specific requirements and budget.

1. Methods for Acquiring Patient Shape Data:
   - Digital Input using CanfitTM Laser Scanner: The hand-held, non-contact laser scanner digitizes 3-dimensional anatomical surface data with a level of accuracy comparable to traditional methods of clinical shape measurement. Any digitized shape can be imported into a general shape modification program. Measurements can also be extracted from digital files for use in the measurement-based design applications.
   - Manual Measurement Input: The measurement-based design applications accept standard clinical measurements. Once this data is entered, users have the option to scale a library shape to the measurement input provided. Extensive libraries of reference shapes, which contain typical corrective modifications are available. Custom shape libraries, which reflect your clinical experiences and preferences can also be established.

2. Designing and Modifying Shapes: CanfitTM design applications are automated shape processing programs, which enable shape data to be stored, modified and easily retrieved. The CanfitTM suite of design tools are the most flexible and powerful available. A combination of large area and regional modifications and overlays, based on individual design and casting techniques, offer ultimate versatility in generating custom modifications:
   - Digital Input: the design software for use with digitized shape data (actual 3D-cast or patient surface shape data) is the general shape modification program, CanfitTM P&O Design;
   - Manual Measurement Input: The design applications, which accept measurement-based shape data and provide convenient reference library shapes include CanfitTM System II-AK Design and CanfitTM System II-BK Design; 3. Design below knee sockets in minutes with the Canfit-XTM System II − BK Design software. Designed specifically for the prosthetics field, this CAD/CAM program provides strict attention to detail and is easy and economical to operate. Socket data can be obtained by fitting an appropriate proximal brim to the patient and taking a series of circumferential measurements or only by taking measurements. Since no cast is required, you eliminate the use of plaster in the design process. This also means that no digitizing is necessary, saving valuable time and money. Conveniently, the intimately fit proximal brim is used as a check socket during the measurement procedure. The measurement data entered into the design program generates a fully modified BK socket and displays it on the computer screen. This shape may be carved "as is," or further changed via interactive on-screen

modification. Whatever your choice, it is extremely simple to achieve success, and most importantly, consistency in your sockets with this CAD/CAM software. The Canfit-XTM BK Design software may easily be used in conjunction with a central fabrication facility and an associated carver. The CANFIT-PLUSTM Remote Communications program transmits your designed socket data via modem or internet/e-mail to the manufacturing facility. Figure 13 portrays some examples.
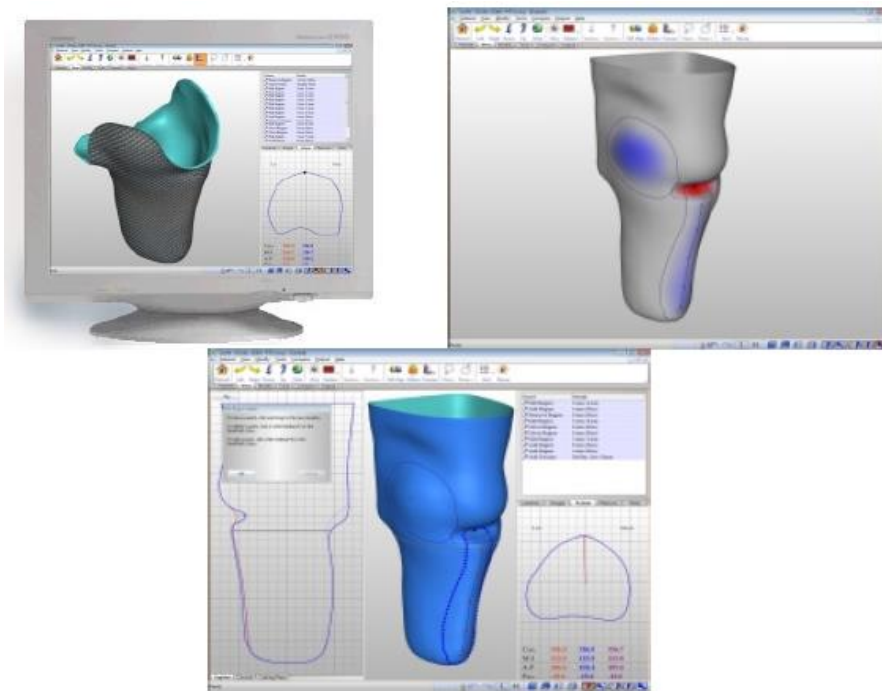
Figure 13: Canfit-XTM BK Design.

2.3.7 Discussions and conclusions

ICT tools can support the specific phases of the product development process, but they do not offer any kind of assistance to the prosthetists. All the design process decisions and actions are taken on the base of their experience and personal skills. Some of these systems are used only by the company, which produces them to develop different prosthetic components. In the case of residual limb, these systems derives the geometry of the check socket or the positive chalk from the external shape of the stump, also using libraries of standard models. Then, the realization of the positive model is guided with a CAM module, onto which the socket is thermoformed. This procedure is always linked to the production of a check socket, which is tested on the patient and then modified. However, none of these systems provides: (i) automatic or semi-automatic procedures that embed technicians' knowledge and assist them during socket modeling; (ii) innovative interaction devices (e.g., hand/tracking devices) to interact by hands with the socket 3D model; (iv) the possibility to use CAE systems to analyze and optimize the product and (iii) the possibility to create multi-material check socket with additive manufacturing technology.

2.4 The Platform for Lower Limb Prosthesis Design

The new design platform for lower limb prosthesis is centered on the patient's virtual model and based on a computer-aided and knowledge-guided approach (

Figure 14). The main idea has been to develop a digital human model of the amputee to be used by the prosthetic to design, configure and test the prosthesis. [1]

The digital patient, around which the prosthesis is designed, is the backbone of the whole system. A biomechanical model and a set of patient's data compose it.

The platform makes available the prosthetist with a set of interactive tools to design, configure and test the prosthesis. It comprehends two main environments:

1. *Prosthesis modelling lab (PML)* to configure and generate the 3D model of the prosthesis starting from the 3D model of the residual limb.
2. *Virtual testing lab (VTL)* to virtually set up the artificial leg and simulate patient's postures and movements validating prosthesis functionality and configuration.

The Prosthesis modelling lab (PML) permits to generate the 3D model of the assembled prosthesis, crucial to virtually study the prosthesis set-up and patient's walking. It integrates three main modules:

— The Socket Modelling Assistant (SMA) implemented ad hoc to model the socket directly around the digital residual limb, following rules and procedures, which replicate the activities performed in an orthopedic laboratory [8].
— A simulation tool based on the finite-element method, to analyze the stump–socket interaction. At present, a commercial FEA system (Abaqus) is integrated within the platform [9].
— A commercial three-dimensional CAD system (SolidEdge) to configure the prosthesis and generate the three dimensional models for the standard parts and final assembly.

The *Virtual testing lab* permits to set up and evaluate prosthesis functionality simulating postures and movements of the virtual amputee with LifeMOD, a biomechanical simulation tool based on MSC ADAMS solver. It permits to create a detailed biomechanical model of a human body using rigid links connected through joints to simulate the skeleton and flexible elements to represent soft tissues. First, the digital patient has to wear the assembled prosthesis. This one is imported from the Virtual Modelling lab and the correct positioning is obtained taking into account the prosthesis height and foot rotation respect to the vertical line. Then, the amputee's avatar can be used to perform static alignment and gait analysis during various patients' activities. The underlying idea is to make available to the prosthetics a library of laws of motion specialized for amputees wearing the prosthesis. To this end, it is necessary to acquire several patients' movements and postures during typical daily-activities and then, derive motion laws for non-natural joints [10].
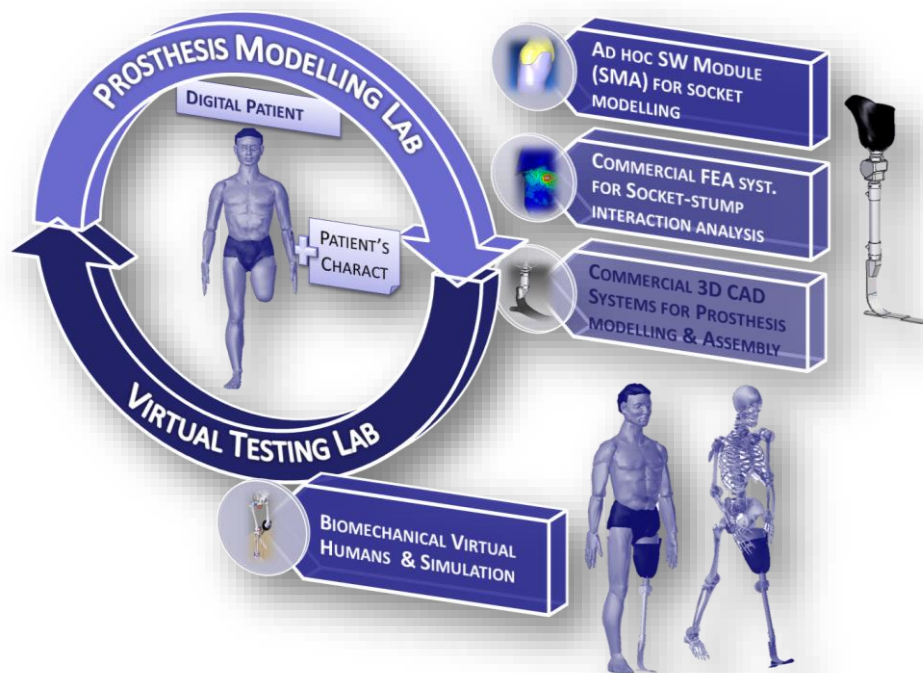
Figure 14 Virtual platform for lower limb prosthesis

2.5 3D Socket Modelling

This Ph.D thesis focused the attention on Socket Modelling Assistant (SMA). SMA is an ad-hoc knowledge guided CAD system developed by Virtualization and Knowledge group at University of Bergamo. It embeds rules defined starting from a functional analysis of the traditional manufacturing process for socket design of lower limb prosthesis (for further details see [11]. This first version has been tested in collaboration with Ortopedia Panini, an Italian orthopedic laboratory located in Milan and with an orthopedic research group at Salford University, Manchester, UK. This permitted to identify some limits and criticalities that guided the re-design and development of a totally new prosthetic CAD system. They are:

- The absence of an automatic procedure to generate the 3D model from MRI images since, usually, technicians have no specific skills on IT technology and current tools for 3D model reconstructions from medical images still require many manual operations and are general purpose. In addition, it was not possible to use low cost technology to acquire the external shape of the residuum.

- The interaction with the socket and residuum 3D models could be done only with traditional devices such as keyboard and mouse. This is not natural for the technicians who usually manipulate/define the socket shape by hands.

- The socket model was represented by NURBS surfaces generated using the free NURBS++ library. Tests carried out highlighted some criticalities related to the socket geometry generation. Figure 15shows two examples: the first is related to the creation of a closed surface and the second to the

fitting of the socket surface starting from an ordered points cloud. Furthermore, researchers of Salford University evaluated not sufficiently adequate the modification of socket surface by using NURBS control points.

- There were no specific tools to define the upper part of the socket using trim-lines.
- The software architecture was not modular. This does not easily permit to integrate within the system new software modules as required to solve above-mentioned limits.
- It was possible to generate directly from SMA the file for AM equipment, especially for 3D multi material 3D printer.
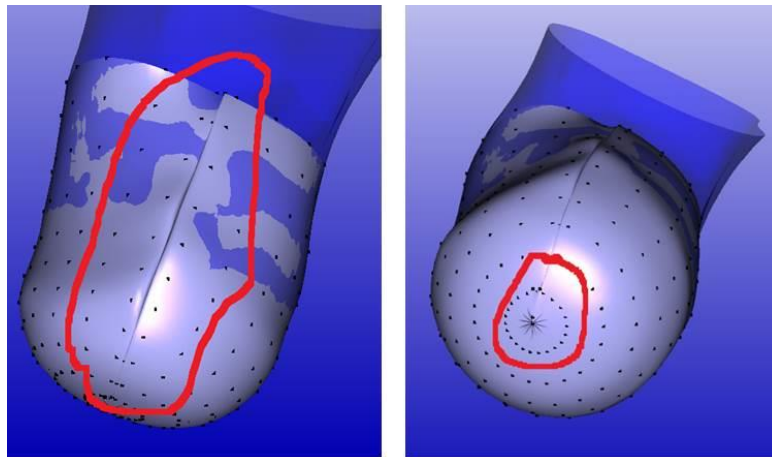
Figure 15: Problems with NURBS surface generation with the first version of SMA.

Form the analysis above-mentioned limits the new prosthetic CAD systems should meet the following requirements:

- Based on the Object Oriented Programming, low cost technologies and open source libraries [12].
- Automatic generation of the residual limb geometry from data acquired both with MRI and low-cost 3D scanning systems [13].
- Virtual modeling tools to:
    - Directly interact and edit the model the socket shape as done during real manufacturing process.
    - Take measurements along the 3D socked and residuum models;
    - Automatically generate and modify the trim-line of the upper part of the 3D socket
    - Generate files for 3D multi-material printer.
- Software modules for interfacing in a simple way new devices for augmented interaction [14].
- Data Exchange using different formats (e.g., IGES, step and STL) to transfer data to and from CAE tools (e.g., FEA and multi-body systems) [9] and additive manufacturing equipment [15].

27

Chapter 3: SMA^2: the new Socket Modelling Assistant

The chapter provides a description of the new prosthetic CAD system, named SMA^2. First, its architecture is presented focusing the attention on the virtual modeling tools specifically develop to manipulate the socket shape. Then, an open source library, named SimplyNURBS, developed for NURBS modeling in medical and health-care domains is described [16].

3.1 SMA^2 Architecture

To solve limits described in the previous section a new version of SMA has been implemented. Figure 16 shows the main architecture of SMA^2. It consists of the following main modules:

1. **3D automatic reconstruction**. Two modules are available to acquire the 3D model of the residual lower limb. The first method consists in a simple procedure based on the use of a Microsoft Kinect v1 as scanner and a low cost commercial application for mesh editing after the acquisition [17]. The second module permits to automatically reconstruct the 3D model of the residual limb from MRI images [13]. The generated 3D model that includes both internal and external parts is necessary to study the interaction socket-residual limb by means of FEA (Finite Element Analysis) tools.

2. **Socket modeling.** The SMA^2 comprehends a set of virtual tools to manipulate the socket shape. They permit to execute the main operations usually performed by the technicians to manufacture the socket. Some examples are: definition of the load and off-load zones, deformations for the shape of the socket model according to the defined zones, set of templates for the trim-line definition for the upper part of the socket and socket thickness generation.

3. **Socket test.** It is available a software module that permits to automatically execute the simulation of the contact pressures between socket and residual lower limb using a commercial FEA system. This has been developed in a previous Ph.D thesis. For further details see [18].

4. **Socket manufacture**. A set of modules permits to generate and export STL files for additive manufacturing both using mono and multi-materials [15].

Figure 16: SMA² high-level architecture.

At present, the user interaction with SMA² is possible using both the hand tracking and traditional devices (i.e., mouse and keyboard). Low cost devices has been exploited either to verify their potential and feasibility in a context where computer aided tools are not commonly used or to make available modeling tools affordable by orthopedic labs [12]. The Leap Motion has been adopted as hand tracking device [17].

The whole application has been developed in C++ language and using a set of open source software development kit (SDK) as follows:

- Visualization ToolKit (VTK) to manage 3D rendering. VTK is an open source software system for 3D computer graphics, image processing and visualization. It supports a wide variety of visualization

algorithms and advanced modeling techniques. VTK has a suite of widgets that are used by user to interact with system [19].

- Qt SDK to create user interface. It allows the integration of VTK in easy way [20].

- Leap Motions SDK that supplies a set of classes and interfaces to use the raw signals of Leap Motion Device. The set of C++ classes is used to develop the source code that allows the association of application events (e.g., move the mouse arrow or rotate/translate a 3D object) with gesture detected by Leap Motion device [21].

- SimplyNURBS, a software development kit specifically developed to manage NURBS surfaces. It has been used to automatically generate the 3D model of residual limb starting from MRI images. The permit to recreate from point interpolation the NURBS surface of residual limb that will be successively re-tessellated in order to get the 3D triangulated mesh [16].

Figure 17 portrays the mind map of the software architecture.
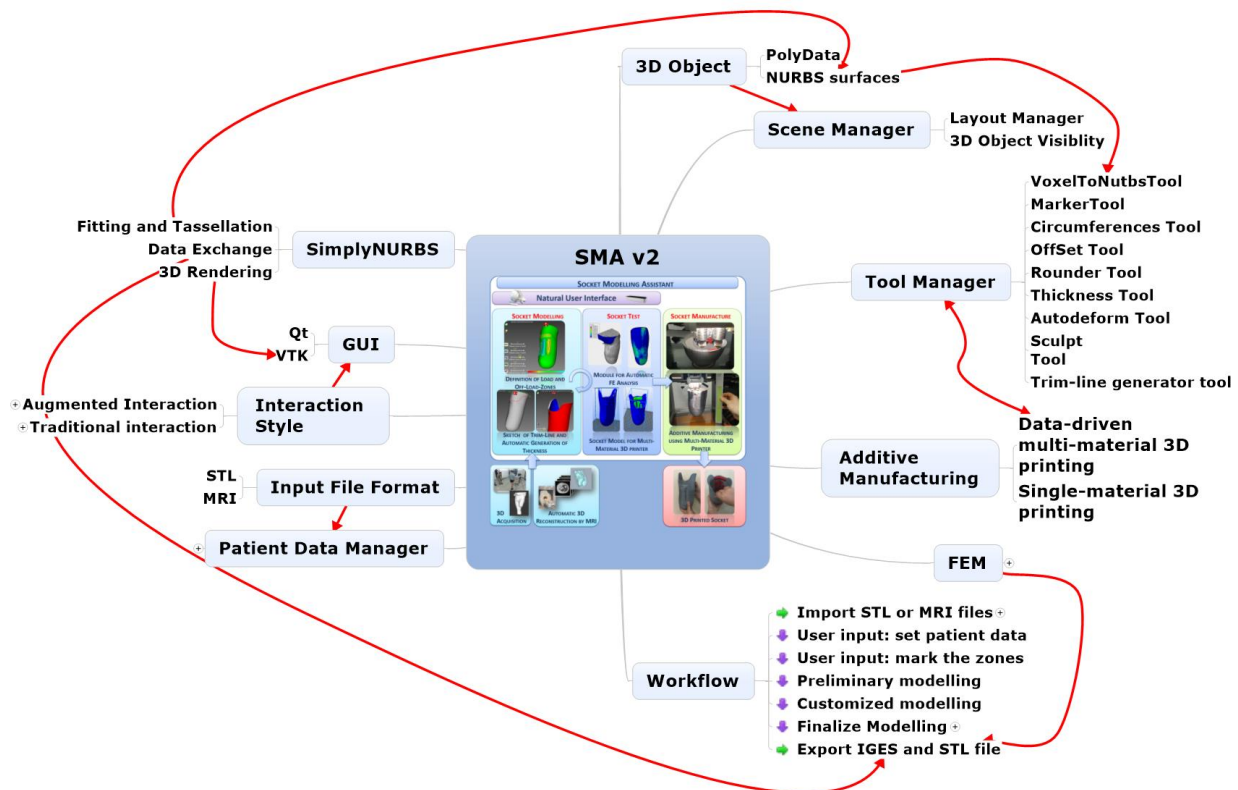
Figure 17: Mind map of SMA.

Inside SMA[2], the 3D objects are triangular meshes and they are managed through the `vtkPolyData` class. SimplyNURBS has been exploited through the `vtkNURBSSurface` class, which are used during the automatic reconstruction of the 3D model by MRI volume. Rendering of the objects is possible by using the `SceneManager` class, which permits the visualization and positioning of objects in the 3D virtual environment according to the used virtual tool. The `ToolManager` class manages the information (e.g., patient's data) that each tool has to consider to execute the corresponding operation on the 3D model.

The `Tool`, class is an abstract class that provides virtual methods to manage 3D objects features such as 3D object visibility and the VTK widget position on the screen. The `ProjectManager` class makes available a set of methods to save/load the project for each new patient. A Qt user interface has been defined to insert patient's data. Furthermore, inserted data are loaded into a data structure that is encapsulated into the `PatientData` class. The `ExportSocket` class permits to export the socket model either with GES or STL format. The STL exporting can be done according to the chosen 3D printing techniques, which are single or data-driven multilateral printing.

SMA[2] is an event-driven application; each tool may allow actions which are executed when an interaction event happens. There are two type of events. The first one use Qt classes and permits the management of every happened event on the user interface and it is able to manage happened events within 3D environment.

VTK event handler allows to manage several parametric widget, such as sliders and color maps. They are used to modify the parameter of 3D objects (e.g., the ray of a sphere). Figure 18 shows the developed UML structure to manage the application events.
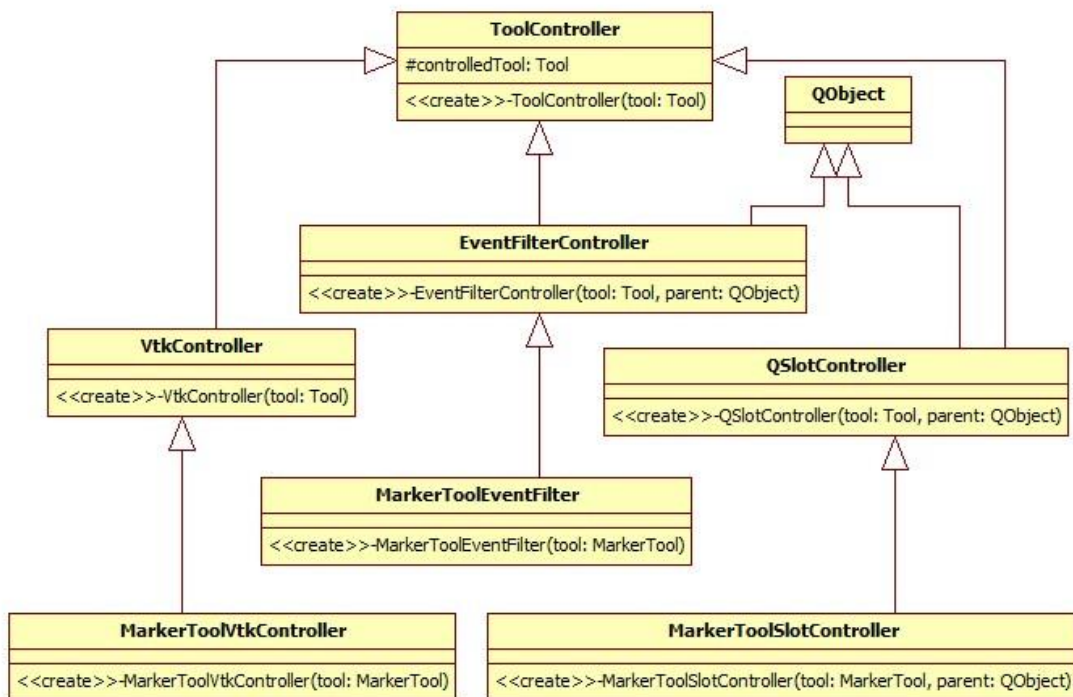
Figure 18: UML diagram of the classes to manage events inside SMA.

All actions are developed within new subclasses of the abstract class `ToolController` and each one is associated at one or more event. In particular, if the event is handled by Qt event handler the developer have to define a subclass of `QSlotController` class (e.g., to click mouse button or to push a button) or `EventFilterController` class (e.g., to hand events from combination of buttons). To use VTK widgets the developer have to create a subclass of `VTKController` class. The association between action and

event is defined within the tool that manages and shows the results of action. These parent classes initialize a set of attributes that allow the typical creation of event handler for both Qt classes and VTK classes.

The `vtkDynamicSculpting` class, which extends the basic classes of VTK, has been developed to permit 3D mesh editing. Furthermore, the custom class `vtkContourWidget` has been developed by us. The custom version of contour widget can be applied on a surface and each modification of it follows the surface of the 3D mesh on which it lie. This widget is used to create the trim-line of socket model and to emulate the tape measure. Tape measure is used to get a measurement of a particular part of the residual limb by the prosthesis (Figure 19).
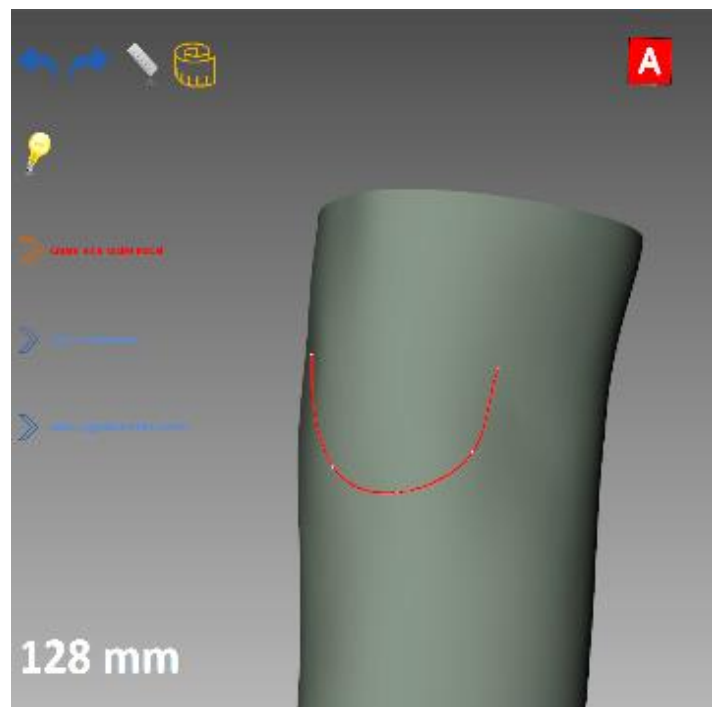
Figure 19: The red line on the 3D model is the virtual tape measure used inside SMA. The measurement is visualized on the left bottom corner on the picture.

A Class structure has been developed to link SMA[2] with a FEA system. At this end, the `FemManager` abstract class has defined and it is composed by virtual methods to be used to implement the main steps of FEA (i.e., pre-processing, post-processing and simulation). At present, SMA[2] use the `AbaqusManager` that class allows interfacing with the commercial product Abaqus in automatic way. We are able to show the results of FEA within a tool easily through an instance of this class. Furthermore, a new module is under development to permit the use of an open source module CalculiX instead of the current commercial product Abaqus. In this way, we have a whole application based on free software or low-costs software.

To permit the innovative interaction style using Leap Motion device, the `InteractorStyle` class, which is a subclass of VTK `vtkInteractorStyle` class. It provides a set of methods to manage all events within SMA (e.g., mouse clicking, mouse moving, and internal timer of application). For the interaction with traditional devices, we redefined the methods by exchanging the left button action (i.e., rotation of a 3D

object) with the right button action (i.e., 3D picking). The interaction through Leap Motion device is based on an event handler that keeps listening until a gesture is detected by Leap Motion device when the associated action (e.g., the rotation of 3D object) is executed. This type of event handler is provided by VTK and uses a timer that starts when SMA^2 application is initialized. The timer generates a time event each timestamp; then, the event handler controls the gesture detection for each timestamp using the classes of Leap Motion SDK. If the event handler detects a gesture the associated source code is executed. The choice of interaction style is possible through a Boolean variable and the user select the interaction style every time s/he want.

In the following sections, the modeling procedure and the virtual tools are described as well as SimplyNURBS.

3.2 Socket modeling

Figure 20 shows the main interface of the modeling environment.
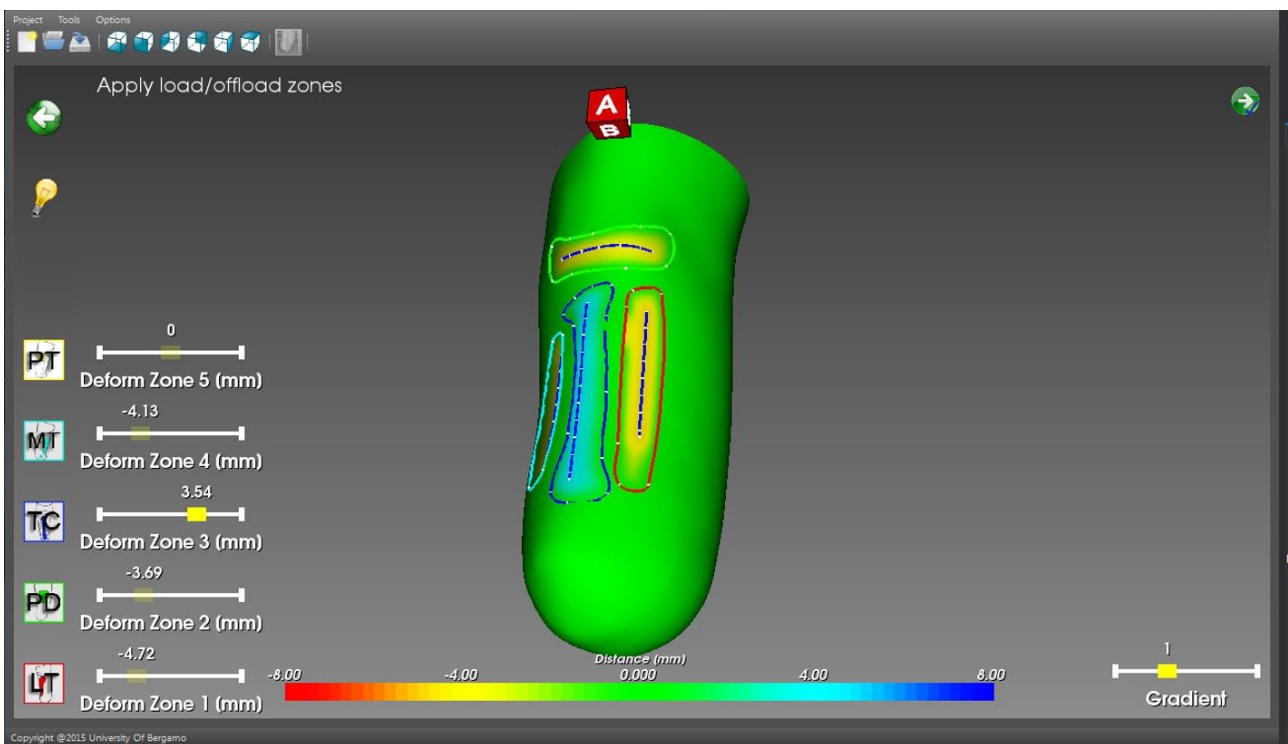
Figure 20: SMA^2 user interface.

The residual limb and the socket geometry are represented by triangular meshes. We decided to adopt this representation because we can emulate shape deformation through a developed self-adaptive topology algorithm [22] as showed in the famous web application named SculptGL [23]. SculptGL is tiny sculpting application, powered by WebGL. It features dynamic subdivision, decimation, uniformization and adaptive sculpting. Classical sculpting tools such as Brush, Inflate, Smooth and Flatten are available. Self-adaptive topology algorithms are based on local subdivision and decimation of the mesh triangles. These two basic

33

operations permit to deform the mesh and, therefore, emulate all the operations performed by the orthopedic technician to manipulate the positive model of the socket adding/removing chalk from the positive model of the socket.

3.2.1 Socket modeling procedure and Virtual tools

SMA^2 permits to model the socket by means a set of interactive virtual tools and interfaces with a FEA commercial software to analyze the pressure distribution at the socket-residual limb interface. It also permits to export data for 3D printing with data-driven multi-material option. The prosthetist is guided step-by-step by SMA^2 that applies in automatic or semi-automatic way design rules and modeling procedures. Implemented design rules and modeling procedures have been studied and defined in a previous Ph.D thesis and details can be found in [11].

Furthermore, SMA permits to interact with the virtual models by exploiting the hand-tracking device Leap Motion. Details about the augmented interaction are described in Chapter 5.

In the following the modeling procedures and related virtual modeling tools are described.

The 3D socket modeling process is structured into four main phases:

1. **Acquisition of patient's data**

As shown in Figure 21, first the orthopedic technician starts inserting patient data (e.g., weight, muscles tonicity, skin conditions and residual limb stability), which are necessary for the following steps to apply rules and/or suggest the most appropriate procedure to the user during each step of the socket design process.

Figure 21: Form to insert patient data.

After inserting patient's data, SMA^2 requires the 3D model of the residual limb. As mentioned before, two ways are available to get this model: from MRI images or from 3D scanner acquisition.

If the orthopedic technician selects the first option, two 3D models (in STL format) representing respectively external shape of residual limb and the internal bones are generated. Figure 22 shows an example for a transfemoral amputee. A color map shows the distances of points of the external shape from the bones.

Figure 22: External shape of the residuum and internal bone automatically reconstructed by MRI volume.

The second approach is based on a procedure using 3D scanner acquisition. The external surface of the residual limb is acquired with Microsoft Kinect v1 and corresponding triangulated model (STL format) is generated with Skanect [24]. Figure 23 shows and example of the residuum 3D model for a transfemoral amputee acquired with mentioned tools.
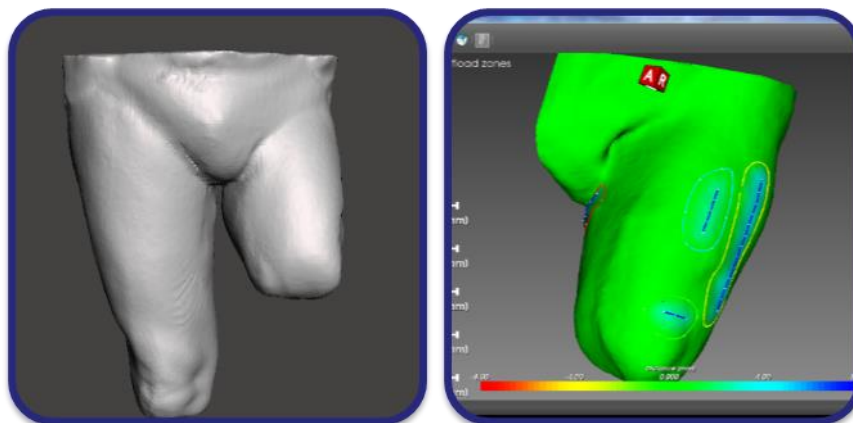
Figure 23: Left picture shows the anatomical district of amputee acquired using Microsoft Kinect v1 and Skanect. The right picture shows the same virtual residual limb used into SMA during zones definitions.

Further details about the adopted acquisition techniques and procedure to generate the 3D model of the residual limb are provided in Chapter 4.

2. **Preliminary modeling**

The main operations during this phase are carried out almost completely in automatic way according to patient characteristics and traditional process.

Five virtual tools have been implemented:

1. The first one, named **scaling tool**, permits to scale the initial model. In fact, in the traditional process the first operation applied on the positive cast is the rasping procedure to reduce the volume. This is done since the socket, manufactured directly on the positive model, has to be perfectly close-fitting on the patient's residual limb. In particular the technician first identifies on the plaster cast the same reference circumferences previously measured on the patient's residual limb, and then starts to file harmoniously the plaster until these circumferences are reduced of the desired percentages. Through a set of cross section planes are defined as shown in Figure 24 and the user can decide the reduction percentage in correspondence of each of them. The range of percentage varies from 1% to 6%. It is not uniform on the stump, but it starts with 1% at 4 cm over the stump top, and it increases gradually going up until the stump upper part. For this procedure the system first identifies the socket top, calculating the lowest point of the geometric model. Then, starting from this point, the system selects 4 reference cross-section plane at a distance of 4 cm from each other. For each of the 4 sections it is calculated the middle point and then the distance of each circumference point is scaled by the appropriate percentage. All the other sections situated between these 4 reference ones are scaled by interpolated values, in relation to their position on model.
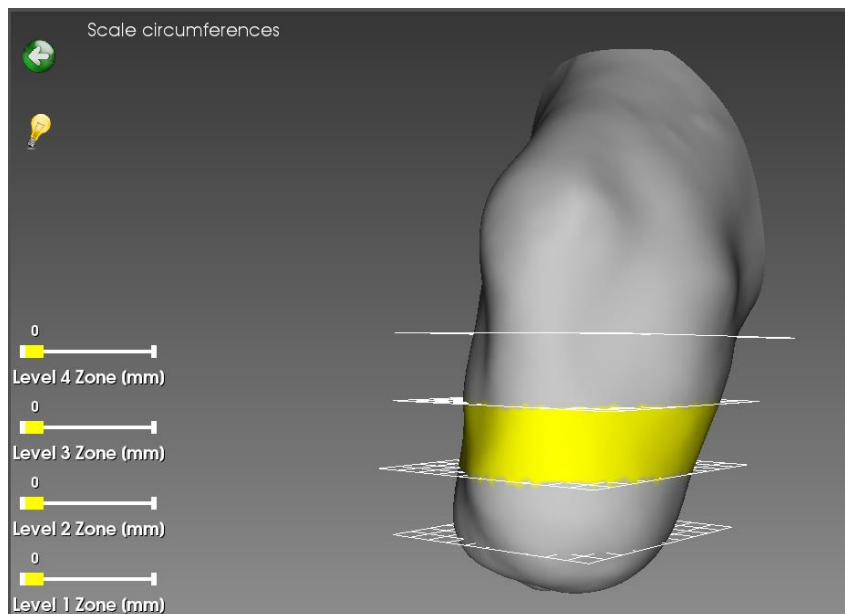
Figure 24: Circumferences scaling tool.

2. The second tool, namely **marker tool**, allows the user to mark on the surface of the virtual residual limb off-load and load zones with different colors. *Figure* 25 shows an example of colored critical zones for a transtibial residual limb. The colored zones are available every time the orthopedic technician wants to know what happen to residual limb due to a modification of socket shape.
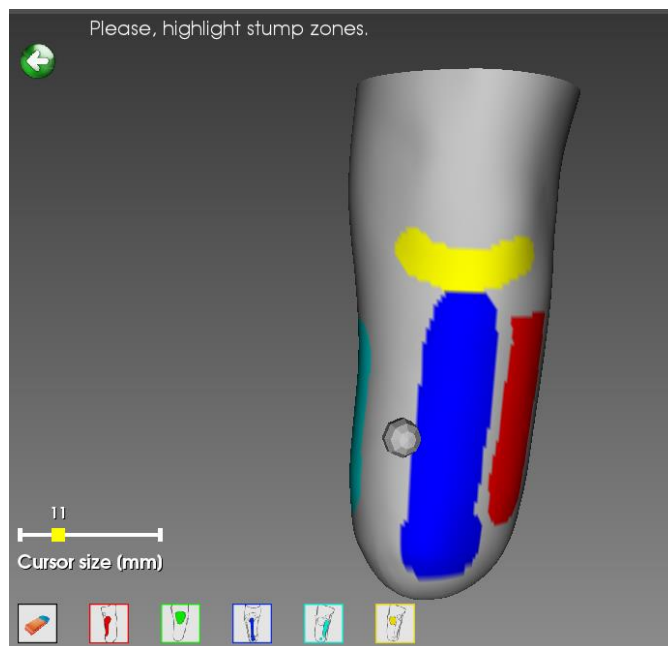
Figure 25: Marker tool.

3. Starting from highlighted zones, the third tool, named **deformation tool**, emulates the operation of adding/removing chalk during traditional process and is automatically executed. If the zone is an off-load zone the mesh of the marked area is pushed inside of a certain quantity according to patient's characteristics, specifically the residuum tonicity; otherwise if it is a load zone the mesh inside the contour is pushed outside (Figure 26). This tool permits also to interactively define and modify the contour line of the load and off-load zones.
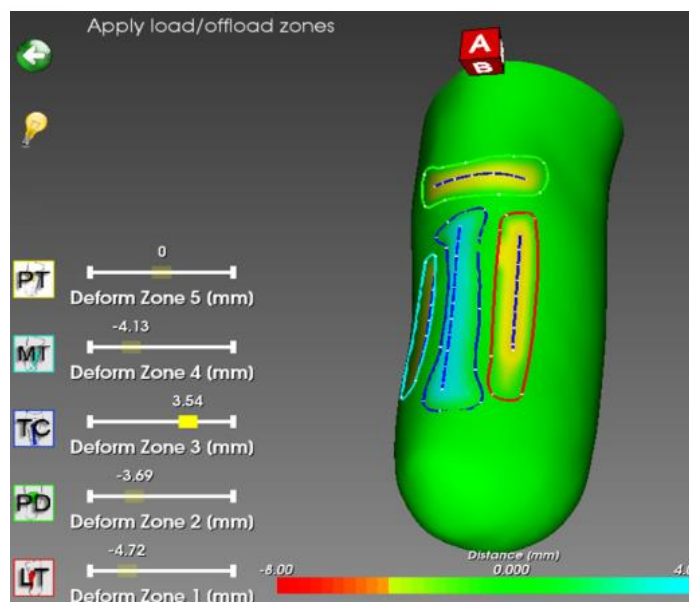
Figure 26: Automatic deformation tool.

4. The last tool permits to create the lower part of the socket starting from the initial shape of the residual limb. This operation is very important because it defines the lower part of the socket

that will be merged with the 3D model of the socket plug. Also in this case, the distance between the lowest part of the socket and the residual limb can be obtained automatically starting from patient data. Some sliders are available in order to change the distance and the roundness of the final part of shape without following the automatic procedure (Figure 27).
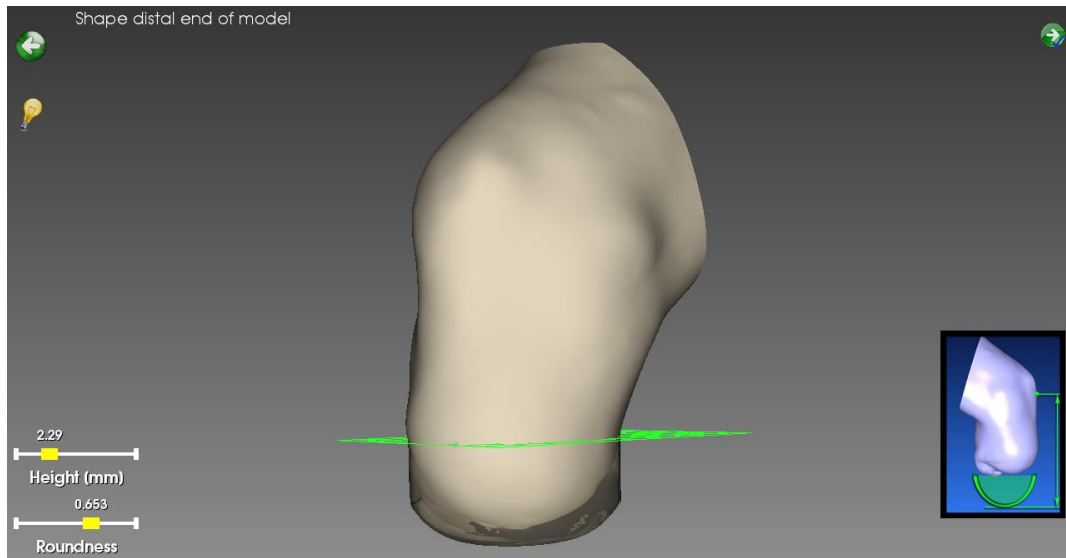
Figure 27: Virtual tool for the definition of socket lower part.

3. **Customized modeling**

   This step permits to customize and refine the model obtained in the previous phase. The user can proceed with an interactive shape manipulation using the modeling tool, named **sculpt tool.** The operations allowed on the mesh are in/deflate, smoothing and flattening. It allows the locally mesh editing and to remove details from 3D mesh of residual limb, such as scars (Figure 28). Also in this case, load and off-loads zones are taken into account in order to inform the orthopedic technicians about the consequences of 3D modification on the residual limb.

a)   b)
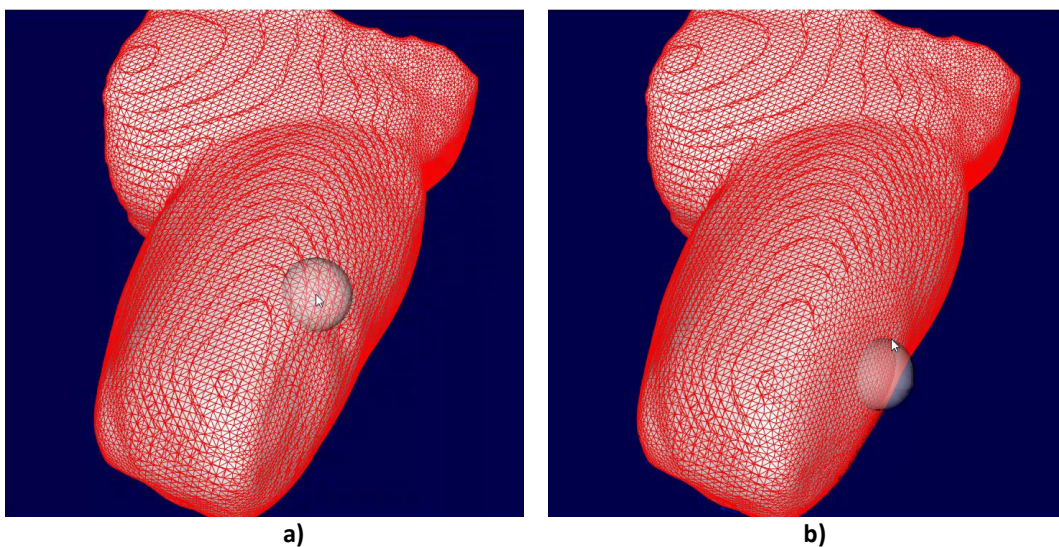
Figure 28: a) virtual model with a scar on the right side of the residual limb. (b) smoothed model using the sculpt tool.

4. **Completing the socket geometric model.**

The last step consists in shaping the upper edge of the socket in an automatic or semi-automatic way. The system provides different templates for the socket trim-line identified in collaboration with orthopedic technicians previously mentioned. Figure 29 shows some examples. The user select the template and the trim-line contour is automatically generated on the mesh of socket. Then, the user can modify the trim line moving the control points along the surface of the 3D residual limb model. Once defined the trim line, SMA^2 removes the upper part of the model and the final shape of socket is created (Figure 29.a**)** and defined the thickness of socket. (Figure 29.b). Finally, the last tool permits to create the hole to assemble the valve.

Once the first 3D socket model is generated, the system automatically executes the FE analysis to verify contact pressure and then optimize the socket shape (Figure 30). FEA results, imported in SMA^2, are analyzed and socket geometry is modified, using the sculpt tool until the optimal shape is reached.

Finally the socket model can be exported for additive manufacturing. The option for multi-material 3D printing has been available and permits to create a socket using different materials according to patient's data and load and off-load zones.

(a) (b)
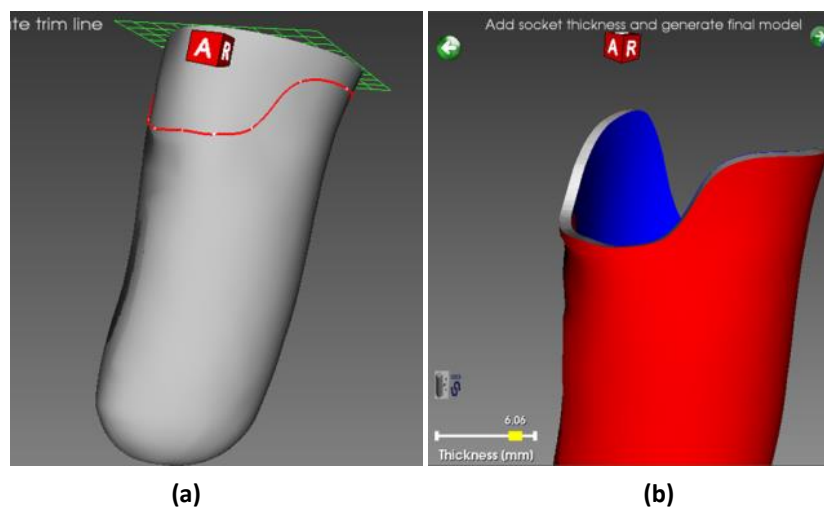
Figure 29: Sketch of trim-line and automatic generation of thickness.
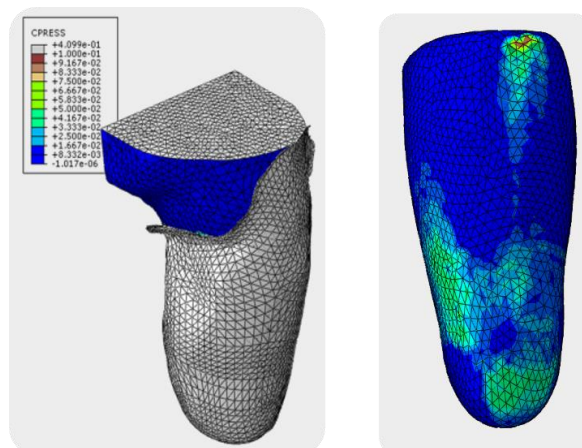
Figure 30 Finite element analysis.

3.3 SymplyNURBS [16]

SymplyNURBS has been developed to solve problems mentioned in 2.5. Initially it should be used to manage the socket model during the modeling procedure, but we decided ad mentioned before to use a triangular mesh to represent the socket geometry. Therefore, it has been mainly used to automatically reconstruct the surface model of the external skin and bones through surface fitting of the points cloud extracted from MRI images. Anyway, SimplyNURBS can be considered an important output of this Ph.D thesis. It is an open source and general-purpose library that can be used for any application field, which requires computer-aided tool to model products around the human body, or anatomical district and organs. An example is the virtual design of custom-fit garments around the avatar of a customer.

3.3.1 SimplyNURBS Layered architecture

SimplyNURBS integrates the key features of existing open source libraries to fulfill the requirements of the considered domain. Many modeling libraries (both commercial and free) are available to create NURBS models. Considering the commercial ones, the most interesting is SMS NLib™[25]. It offers a comprehensive suite of geometric modeling libraries, among which NLib™ the NURBS based library. NLib™ is written in C language and provides an extensive set of robust functions for generating and manipulating NURBS curves and surfaces [26]–[28]. It also embeds a set of modules to interface with other applicative software (e.g., CAD and CAE systems).

On the other hand, during last decades, several open source NURBS based libraries and 3D modeling tools have been developed (e.g., NURBS++, OpenCASCADE, Sintef, Nurbana and OpenNURBS http://www.sintef.no/, http://nurbana.sourceforge.net/ and http://www.rhino3d.com/opennurbs). Among them, two libraries, NURBS++ [29], OpenCASCADE [30], and the 3D modeling environment AYAM [31] have been considered since they are broadly used, include many basic operations on NURBS models, and are sufficiently user-friendly.

NURBS++ it is a C++ library and, as already mentioned, it has been the first one we experimented to develop SMA. It hides the underlying mathematical representation and offers many functions to generate NURBS models from point dataset. However, its development stopped in 2002 and we experienced some troubles as shown in Figure 16.

OpenCASCADE is a software development platform that includes C++ components for 3D surface and solid modeling, visualization, data exchange, and rapid application development. OpenCASCADE has a modular structure, which allows developing complete CAD/CAM/CAE application in an easy way. However, it permits to interact with NURBS models only through the control points. For our purpose, the most

interesting feature is related to data exchange; in fact, NURBS models can be exported in different formats, such as IGES, STEP and STL, with few instructions.

The third one, AYAM, is not a library but a free 3D modeling environment that supports NURBS modeling. We focused the attention on its mathematical kernel completely developed in ANSI C. It permits to solve previous mentioned criticalities and has better performances than NURBS++ kernel.

Anyway, none of the mentioned open source tools is able to meet the requirements mentioned in §2.5. Therefore, we decided to develop an open source library, named SimplyNURBS, extracting from each of them the features that are meaningful for our applicative domain.

SimplyNURBS has been developed upon AYAM, OpenCASCADE and NURBS++. Its architecture is structured in different modules according to considered features of mentioned libraries and the standards for high performance graphics (i.e., OpenGL).

3D Rendering is based on the software suite VTK. VTK is an open-source software system for 3D computer graphics, image processing and visualization. It is mainly used to develop 3D application in research domain, such as medical and physics fields.

This also permits to introduce how to use SimplyNURBS for other applications. In fact, the library has been mainly designed to meet the requirements of the prosthetics field, but it can be used for other applications in health care that require 3D surface models of anatomical districts and organs.

SimplyNURBS is based on a layered structure. Figure 31 shows the architecture and main feature are the following ones:

- Mathematical Kernel. This part is completely written in ANSI C, and it has been extracted from AYAM.
- Interaction paradigm. It permits to modify NURBS surface acting directly on the surface point and without the use of the control points. This feature is based on a part of code extracted from NURBS++ and specifically updated for our purpose. Furthermore, this module permits to introduce another way to interact with 3D models using hand-tracking and haptic devices [12].
- Fitting. It permits to create a curve/surface interpolating points cloud [32]. An algorithm has been developed to extract an ordered point cloud from a set of raw data (e.g., 3D scan or MRI volume). Therefore, the actual computer's power permits to define a NURBS model from the ordered points cloud in real time, and thus, without a complex approach to define the mathematical representation.
- 3D Rendering. A set of methods has been developed to permit 3D rendering of the models. These modules are based on OpenGL primitives. Furthermore, an interface has been developed to use

NURBS models within virtual environments based on VTK. The implemented interface permits to manage the 3D rendering in a very simple way avoiding involvement on low-level software development.

- Data exchange: This module has been developed using OpenCASCADE in order to obtain data exchange translators for IGES and STL. In fact, both AYAM and NURBS++ do not include this feature that is meaningful for our application field. This part of the suite is entirely developed using C++ language.
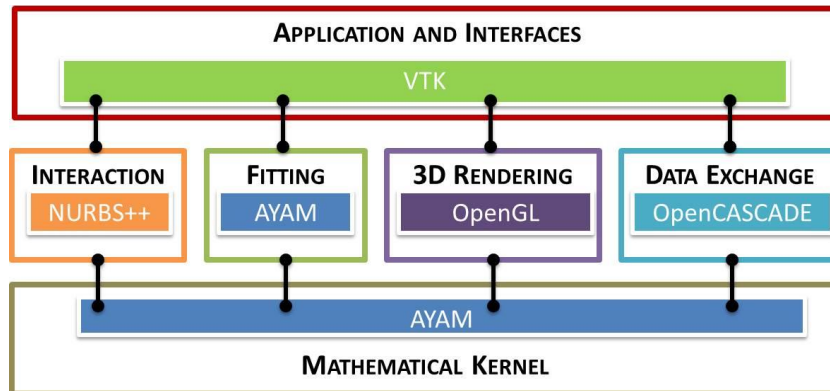
Figure 31: Software Architecture of SimplyNURBS.

3.3.2 How to use SimplyNURBS

The upper layer of SimplyNURBS architecture is based on VTK thus, each mentioned module is called within a set of extended VTK classes, which can be used together with its classic kernel released from Kitware to manage NURBS models. VTK permits to manage each feature related to a 3D environment, such as definition and interaction with both scenes and 3D objects. The mind map [33] in Figure 32 shows the software structure that permits to understand how the instructions have to be used. In particular, there are three main operations embedded into the upper layer:

- Fitting. This operation concerns the creation of NURBS models from a points cloud and so, it is a particular 3D object managed by VTK through the use of single instance. The defined instance automatically provides the generation of parametric model that is also initialized for 3D rendering using OpenGL primitives.
- Interaction. The interaction occurs using an extension of the VTK interactor-style that is based on the use of keyboard and mouse. A new module permits to modify NURBS models within the interactor-style by associating either the typical approach using mouse and keyboard or the tactile interaction. For example, this module permits to select a surface point using two fingers and to modify its position according to fingers position.

- Export 3D NURBS model. VTK makes available a set of classes permitting to export typical 3D models in different formats, such as STL. We have added a new exporting class, which is able to export NURBS models in both STL and IGES.
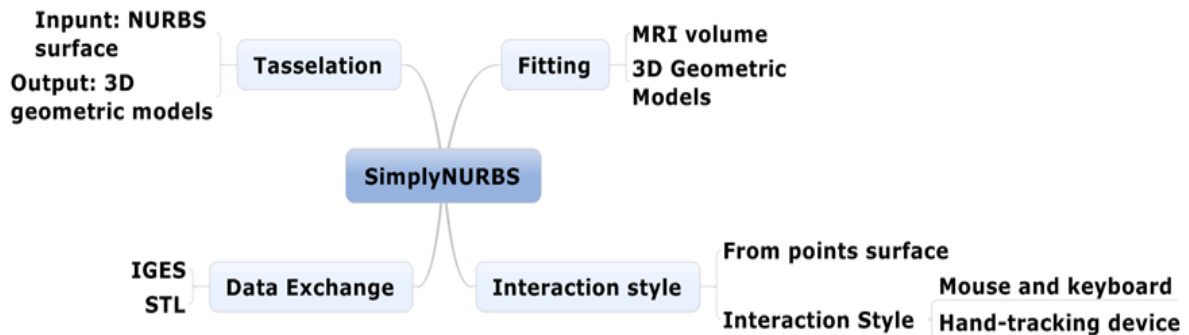
Figure 32: Min map summarizing the features of SimplyNURBS.

A set of tests has been carried out to evaluate functionality of SimplyNURBS with regard our context. Fitting has been tested with different dataset of point clouds. First, we considered sets of MRI images acquired for different amputees and automatically reconstructed the 3D NURBS models. Later, they have been used to generate the NURBS surface, which describe the socket shape. The results reached so far have been compared with those of previous version. It has been highlighted that all NURBS surfaces were properly created and problems mentioned in section 2.5 have been solved. In addition, the correct import of modes in IGES format has been verified. Several ambiguities were present during the fitting of internal bone of below knee amputation. In fact, the detail of the patella was usually lost and the final geometry resulted not similar enough to the real one.

Chapter 4: Residual limb 3D acquisition

This chapter presents an overview of main techniques to acquire human morphological shapes; then, the two approaches adopted for the considered field are described: the first is based MRI images to get the final 3D shape of residual limb, while the second on the use of a low-cost scanning device, i.e., the Microsoft Kinect v.1. Finally, a comparison of mentioned approaches is exposed as well as results reached so far.

4.1 Techniques for residual limb acquisition

Human morphological shapes can be acquired using different technologies. We have grouped them in two main classes:

1. Medical imaging.
2. 3D scanning.

In the following, the most diffused techniques available nowadays on market are introduced [11].

4.1.1   Medical imaging

As presented in [11], in medical sector these techniques are called biomedical imaging or diagnostic imaging, terms used to identify the general process, which allows to observe part of an organism not externally visible. The first imaging technique, the radiography technique, was born at the end of the 19th century, when the German physicist Wilhelm Conrad Röntgen discovered Xrays and it was suddenly understood the possibility to use these rays to obtain human body images (the radiographies). Ever since more sophisticated techniques have been invented, which allow to study anatomy and different biochemical and physiological processes at the base of human organism functioning. In the '70s a big evolution in diagnostic imaging has been represented by introduction of tomographic techniques. These techniques provide images of human organism sections and allow to 3-dimensionally reconstruct and visualize organs and apparatus, by support of modern computers. The principal techniques nowadays available for imaging acquisition of an organism, such as human body, are:

a) Traditional and computerized radiography.

b) Computed tomography.

c) Magnetic resonance.

d) Positron emission tomography (PET).

Traditional radiography is the first invented biomedical imaging technique, and still in use. Radiography is based on X-rays and the barrier effect due to interaction between matter and radiation. It was born at the beginning of the 20th century, and until the '50s it was the only diagnostic imaging technique. X-ray images are obtained generating an X-ray beam and let it go through the patient's body. During the '30s the Italian radiologist Alessandro Vallebona proposed a method to represent a single layer of the human body on a radiographic film: this exam is called stratigraphy. From the middle of the '80s, it has been supplanted by new computerized technologies (see example in

Figure 33).

In medicine **Computerized Radiography** (CR) is the equipment, which allows obtaining digital medical images by X-ray, using specific memory phosphors, which are deleted and reused for many times. This technique was born during the '80s. Absorbing an X-photon, phosphor is excited for a medium/long time, memorizing the photon position. This information is read illuminating phosphor by a red/infrared laser ray. The light, which goes out from the phosphor, is collected by a photo-multiplicator, and a computer collects its information and shows on a monitor the X-ray image, which has generated the signal. This image can be printed, stored in a hard disk, recorded on a CD/DVD or elaborated. In fact this system has not the same resolution as the traditional one, but since computers can improve images thanks to processing image algorithm, this system can provide better images [34].
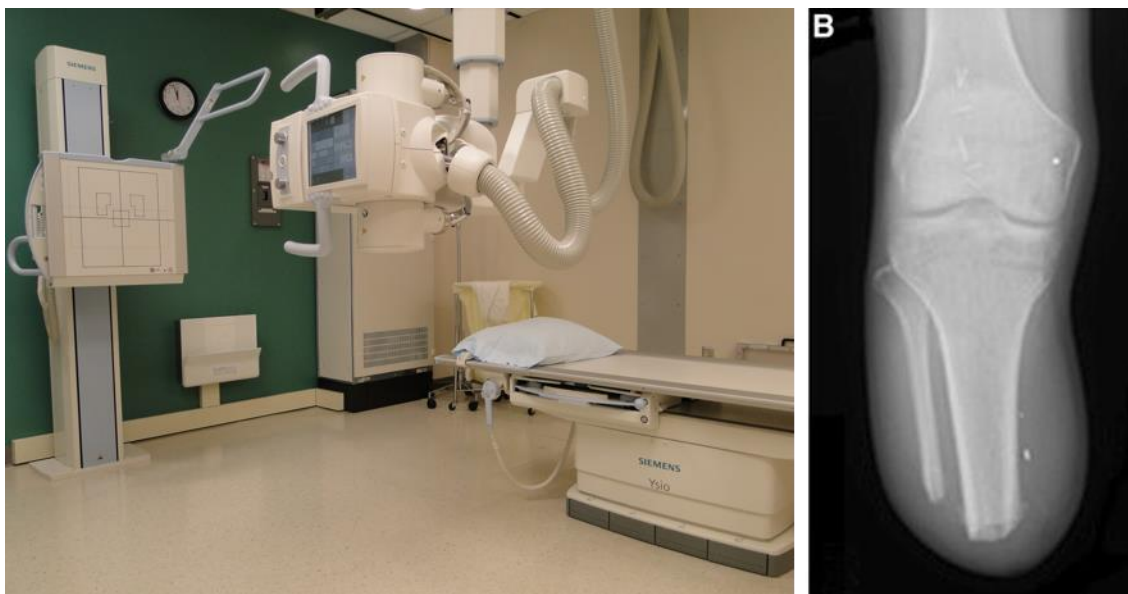
Figure 33: Equipment for traditional radiology (left) and an example of computerized radiology of below knee amputation (right).

Computed Tomography (CT) is a diagnostic imaging method, which uses X-ray, and reproduces patient human body sections and 3-dimensional processing (Figure 34).

45

Figure 34: Example of CT machine (left) and example of CT human body with above knee amputation (right).

It is also well known as Computed Axial Tomography (CAT), but the adjective "axial" is not any more appropriate, since the last generation of this technology doesn't acquire only on one axial plane, transversal, but it has been adopted a spiral acquisition technique, which obtains more images in a single acquisition. The image is created by CT measuring the fading effect of X-ray, which goes through the human body. This varies proportionally to the electronic density of crossed tissues. Since the obtained images are digital, the human body is divided in volume elements (voxel), to each of them it corresponds a single image element (pixel), on the base of a grey scale. A voxel with higher density has a lighter grey (see scale in Figure 35).
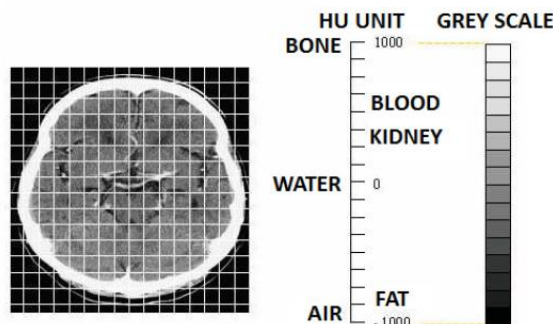
Figure 35: Hounsfield grey scale relative to principal human body organs.

TC methodology obtains better results than traditional radiology regards to diagnostic imaging of soft tissues. Unfortunately radiation dose for the patient is quite high compared to traditional radiology. For this reason it has to be adopted only if strictly necessary, in particular for children. The TC exam can be improved using a contrast liquid by intravenous injection, which better identifies structures with same

density. The emitter of X-ray beam rotates around the patient and the detector, at the other side, collects the image of the patient's body section; the patient is lying on a couch, which slides in precise and determined way inside a tunnel, showing a different body section at each rotation. The sequence and of images and the angle of acquisition are processed by a computer, which shows the results on a monitor. The results are consecutive sequences of sections, with a determined thickness: all the sections can be processed by a 3-dimensional rendering software to obtain tomographic images in any spatial plane or 3-dimensional. These images can be stored in a hard disk, recorded on a CD/DVD or printed on a film. In unidirectional continuous rotation TC machines, emitter and detector are installed on a rotational ring [34].

Magnetic Resonance Imaging (MRI) or Magnetic Resonance Tomography (MRT) or simply Magnetic Resonance (MR) is a diagnostic imaging technique, based on the physical principle of nuclear magnetic resonance. In fact the density signal in MR is given by the atomic core of the examined element. MR is generally considered not noxious for patients. The functioning principle is based on processing patients to a strong static magnetic field. The magnetic field intensity can vary from tenths of Tesla, for small machines, to 3 Tesla for last generation of MR machines (Figure 36). In this static field, the proton spins inside tissues are aligned to the force directions; in this way tissues acquire a small magnetization. The aligned protons show a procession in the magnetic field direction, which has a typical frequency. If to the patient it is applied a rotating frequency and sufficient energy, it is possible to rotate protons magnetization of a determined angle (called flip angle) on the base of the images which are requested. Giving energy to this frequency is the phenomenon, the resonance, which gives the name to the method. After the impulse, protons come back to the magnetic field alignment, and this is called relaxation. This relaxation happens in two determined time constants: the first one, called T1, is the rapidity of protons to come back to the original alignment; the second one, called T2, is the rapidity of destruction of tissue magnetization.

Figure 36: Example of MR machine.

During an MR exam patient cannot wear metal objects; in particular it has to be known if patient has metal chips into his/her body due to previous operations/accidents. This because a strong magnetic field such as MR one can move these metal objects and causes harms to the patient tissues. Until few times ago prosthesis, vascular clips, stent or other chirurgical apparatus were an obstacle to execute an MR exam, anyway after the '90s these objects are made with material compatible with MR. While pacemakers are still dangerous. The patient lies on a couch, which slides through a big ring. An MR image can look similar to a CT image (see example in Figure 37), but in MR case bones are dark. The technical spatial resolution is quite low, but information discovered with this system are different from the others; in fact tissues are differentiated on the base of their biochemical composition, e.g. there is difference between tissue of liver and spleen, or between healthy and diseased tissue. The scanning time is rather long compared to other techniques [34].
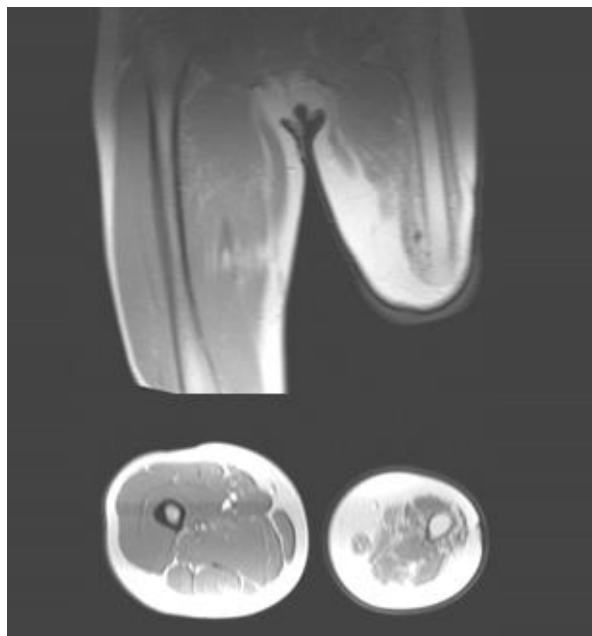
Figure 37: Example of MR images of transfemoral residual limb

Positron emission tomography (PET).

To conduct the scan, a short-lived radioactive tracer isotope is injected into the living subject (usually into blood circulation). Each tracer atom has been chemically incorporated into a biologically active molecule. There is a waiting period while the active molecule becomes concentrated in tissues of interest; then the subject is placed in the imaging scanner. The molecule most commonly used for this purpose is F-18 labeled fluorodeoxyglucose (FDG), a sugar, for which the waiting period is typically an hour. During the scan, a record of tissue concentration is made as the tracer decays. As the radioisotope undergoes positron emission decay (also known as positive beta decay), it emits a positron, an antiparticle of the electron with opposite charge. The emitted positron travels in tissue for a short distance (typically less than 1 mm, but dependent on the isotope [30]), during which time it loses kinetic energy, until it decelerates to a point

where it can interact with an electron.[31] The encounter annihilates both electron and positron, producing a pair of annihilation (gamma) photons moving in approximately opposite directions. These are detected when they reach a scintillator in the scanning device, creating a burst of light which is detected by photomultiplier tubes or silicon avalanche photodiodes (Si APD). The technique depends on simultaneous or coincident detection of the pair of photons moving in approximately opposite directions (they would be exactly opposite in their center of mass frame, but the scanner has no way to know this, and so has a built-in slight direction-error tolerance). Photons that do not arrive in temporal "pairs" (i.e. within a timing-window of a few nanoseconds) are ignored [34].

Among several techniques of medical imaging, the module developed in SMA^2 exploits MRI because this technology is less invasive than Computer Tomography.

4.1.2 3D Scanning

Nowadays, technological progress made possible to introduce the concept of 3D scanning in the industrial world. Scanning means to create a virtual copy of the document or object considered. 3D scanners invaded the global market since they offer great opportunities by allowing users to quickly create a 3D model that can be used in any CAD software. The evolution of 3D scanning systems has determined a large range of commercial solutions available on the market with different costs depending on their performances. The most interesting scanners for the sake of this research rely on structured light optical sensors like Microsoft Kinect v1 sensor, which are extremely low-cost, but they still provide a precision that is valuable for some medical applications, e.g., the scanning of a residual limb.

In the following sections, classifications of mobile scanner technologies according to operating principles and Precision and accuracy.

Operating principles

All scanning technologies are related to only two operating principles [35] based on light projection: laser technology and structured light projection. Both operating principles use the triangulation method to obtain the distances from the sensor and to create the point cloud of the scanned object. This method consists in calculating the distance of a point knowing the projection, and the receiving angle using trigonometric equations.

Laser scanner is formed by a laser emitter and a camera (Figure 38.a). These components are displaced with an angle between them that define the field of view. The emitter projects a laser stripe on the object to scan, it is framed by the camera and using triangulation method, it is possible to calculate the distance from each point of the object. The most important benefits of this system are high precision and accuracy

49

in comparison between real object and scanned model. This operating principle has two disadvantages: the high cost of the hardware components and the scan duration that is high for the small scan area.

Scanners are based on structured light technology (Figure 38.b) consists in the projection of light pattern of IR/laser beams and using triangulation formulas, they permit to evaluate the depth value for each point of the pattern. These scanners include a pattern projector and a light sensor. This technology offers the possibility to capture very large field of view in a single scan acquisition and to increase the precision comparing data with the previous and following frames. Another type of scanners that use a light pattern is based on flash bulb and a camera. The bulb flashes a light pattern onto the object and the camera records this pattern. The distortion in the light pattern, due to the specific curvature of the object is then translated into a 3D image. This type of light is not dangerous for human eyes like laser beams.
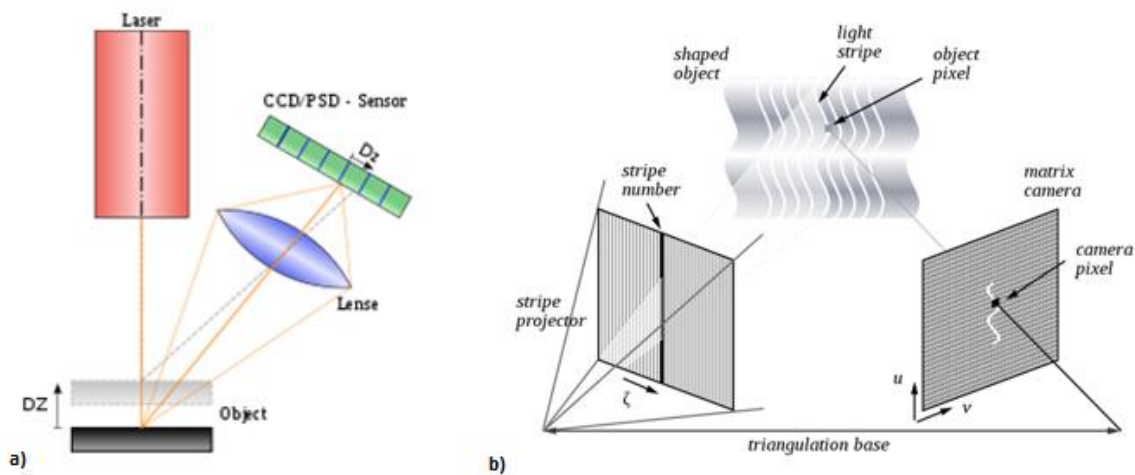
Figure 38: a) Laser triangulation b) Structured light technology.

Precision and accuracy

3D scanner can be classified from an application point of view according to precision and accuracy class. Each scanner ensures a level of precision and accuracy depending on operating principle and structure. Accuracy means the trend to converge of the measurement value towards the average value, while precision consists in having all measurements in a very restricted range around the average value. There is a strength relation between the level of this two features and the price of the scanner. In this way it is possible to associate low price to low precision and accuracy and vice versa.

The first class includes all **low cost scanners**. Most of them come from the video-gaming world and for this reason their price is very low. They cost less than one thousand euros. The precision is related to the precision required in video games; it depends on environment conditions, light and distance from the object. Anyway, the precision value is around 5-10 mm, that is not comparable to laser scanner, but it has many advantages as illustrated in [36]. All devices belonging to this class use structured light projection as operating principles with IR beams. The most representative scanner of this section is the MS Kinect v1

(Figure 39.a) that is the sensor integrated with XBOX 360. It permits to scan objects with a lot of software and to elaborate a triangular mesh model. Its features and its very low cost are suitable for various medical applications such as rehab application, postural control and gait analysis as explained in [37]–[39].

Medium cost class includes 3D scanners from 1.000 to 15.000 euros. Hardware improvement permits an increase of precision with value from 0.1 to 1 mm. This value is also validated by the accuracy index (around 0.1 mm). These scanners are dedicated to artistic and industrial scopes. They uses as operating principle either structured light technology and laser projection. A good example of this group is the FARO CAM2 Freestyle scanner (Figure 39.b).

High cost class includes devices with a cost over 15.000 euros. They are intended for industrial scopes and for this reason they assure a high precision. In fact, guaranteed precision is about 0.005 to 0.01 mm. Obviously, high performances are directly proportional to the price of the scanner: depending on the application, the price could reach hundreds of thousands euros. For example, CREAFORM HandySCAN 700 (Figure 39.c) offers the possibility to scan object with a precision around 0.05 mm and costs up to fifty thousand euros [40]. All these scanners are based on laser projection system and using triangulation they can calculate precise measurements. Some of them mount also a contact hard probe to measure the object in particular points and to integrate non-contact data with the aim to obtain a more precise 3D model.

Figure 39: a) MS Kinect v1 b) Faro CAM2 Freestyle c) CREAFORM HandySCAN 700.

Inside SMA^2, a simply procedure is available to exploit Microsoft Kinect V1 with the cheap application Skanect in order to follow the low cost philosophy of the whole project.

4.2 Automatic 3D reconstruction from MRI Volume inside SMA^2 [13]

The detailed model of the residual limb, including external and internal (soft tissues and bones) parts is built from medical images acquired using MRI technique. Reconstruction from MRI scans is more challenging than from other medical imaging techniques, but we choose MRI since it is the less invasive for the patient and commonly used in considered field. The implemented reconstruction procedure is composed by three different phases: image pre-processing, voxel segmentation, 3D models generation. In the first one, the MRI images are pre-processed in order to reduce noise and digital artifacts. Initially, we

tested a classic Gaussian smoothing but, in our case, this kind of filter did not preserve edge very well, so we decides to implement noise reduction with a 3D anisotropic diffusion filter. This kind of noise reduction fits very well our purpose since it can also work in three dimensions and we can apply the filter to the whole 3D matrix of voxels. The second phase consists in segmenting the voxels belonging to the bone and those belonging to the external surface and in discarding the other voxels. The final output of this phase consists in two clusters of voxels that represent the geometry of bone and of the external surface from which we can derive the soft tissue. In the thirds phase, the procedure, starting from the voxels cluster of bones and soft tissues, creates the 3D geometric models. This operation is carried out using NURBS surfaces. The control points of the NURBS are placed on the external perimeter of the cluster, in order to define the correct shape. Then, the module can export a standard IGES or an STL file after triangulating the NURBS surface. In the following we illustrate the reconstruction of the soft tissue and of the bone, considering as reference a set of MRI images acquired from a transfemoral amputee (i.e., patient with amputation above knee). The automatic procedure has been implemented using C++ languages and SymplyNURBS.

4.2.1   External shape

The external surface of the residual limb is segmented in the following way. After having highly increased the contrast, edge detection by a canny filter is performed (the threshold value is 0.01) and small edges (segments with less than 50 pixels) are discarded (Figure 40). This operation is carried out slice by slice along both axial and coronal axes. The final result is a set of voxels that composes exactly the external surface of soft tissue (Figure 41).
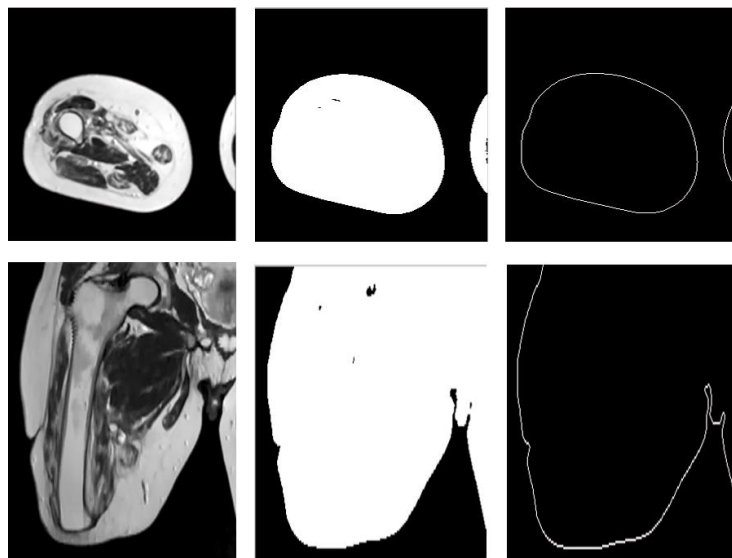
Figure 40: External profiles detection: Axial and coronal slice (left), high contrast image (center), canny edge detection (right).
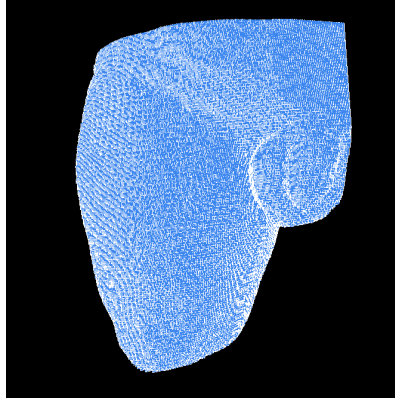
Figure 41: Automatic segmentation of residual limb.

4.2.2 Bone

Inner bone detection from a MRI scan is challenging because we have not a unique intensity level for every kind of tissue. Moreover, since we have to reconstruct the amputee bone, varying in size and shape, we cannot use an approach that exploits a-priori knowledge, such as ATLAS based segmentation or deformable models [41].

The MRI images can be considered as a 3D matrix of voxels, in which every single slice is part of a vertical stack. We decided to use a graph approach as proposed in [41], adapting the algorithm to a three dimensional matrix. Initially, each voxel of the MRI volume belongs to a different cluster that has intensity mean corresponding to the voxel intensity. For performance reason, voxels with value equal to zero (intensity equal to zero means empty space) are discarded.

Then, we create a 3D graph where nodes correspond to clusters and edges (connecting clusters) have weight calculated as difference of the intensity between the two connected clusters. We use a 26 neighbor's structure (all the neighbors' voxels sur-rounding a voxel). The 3D graph is then segmented as described by the following pseudo code: (code is written in C++ language):

```
Sort edge by weights (from the lowest to the highest)
For every edge i...
  Get clusters A e B connected by edge i
  If (mean intensity difference between clusters A e B< threshold K)
    Merge clusters A e B in a new cluster C
    Calculate features of cluster C (mean intensity)
  End
End
Final clusters analysis, discard clusters with size and width/height ratio that
doesn't fit bone shape
```

53

Figure 42 shows an example of the segmented cluster corresponding to the bone of the residual limb for the transfemoral amputee.
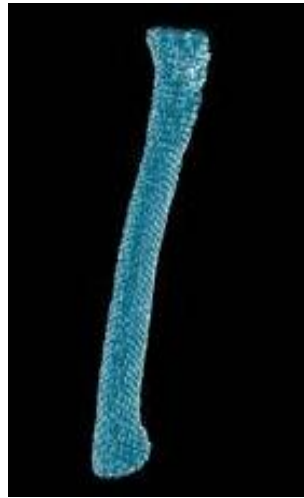
Figure 42: Automatic segmentation of the bone.

4.2.3 Geometric models generation

This phase consists in generating the 3D geometrical model of soft tissues and bone using NURBS surfaces and starting from the clustered voxels. Assuming the z-axis as the vertical axis of the clusters, we initially create a set of x-y planes intersecting the clusters from bottom to the top and with a regular step. For each plane, we create a NURBS circumference whose center corresponds to the center of the voxels belonging the plane. The radius of the circumference is set in such a way that the circle surrounds all the voxel belonging to the plane. Then, a convex hull operation is performed on the voxels belonging to the plane in order to obtain a closed perimeter and include any missing voxels. Afterward, for each NURBS circumference, each control point is moved toward the center of the circumference itself until the point intersects the first voxel, which lies on the external surface of the voxel volume. Figure 43 shows the procedure.
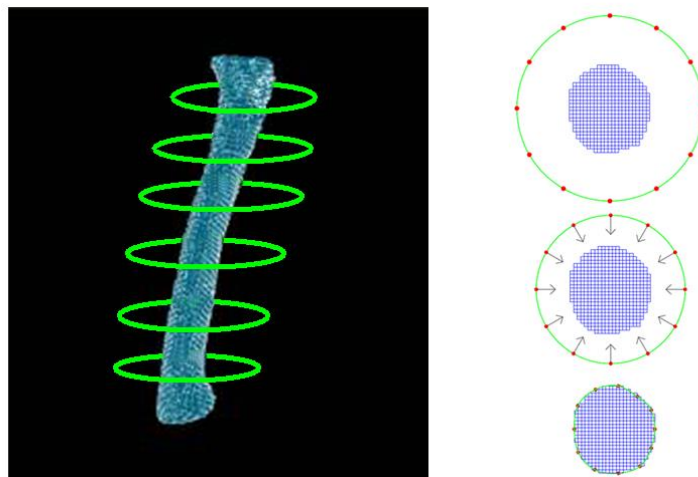
Figure 43: NURBS perimeter voxels fitting

At this point each circumference has the shape of the external perimeter of the voxels group that the circumference surrounds. Finally the NURBS surface is generated by a lofting operation applied to the circumferences. The NURBS surfaces can be exported in IGES format to the SMA[2] through the use of SimplyNURBS. It can be also easily converted by triangulation into a tessellated model and exported in STL format. Figure 44shows the NURBS and tessellated model of external surface while Figure 45 of the bone generated by using SimplyNURBS.

The more U and V NURBS parameters are higher, the more the NURBS surface matches the external voxels position; we figure out that U=20 and V=12 lead to a reasonable level of precision.
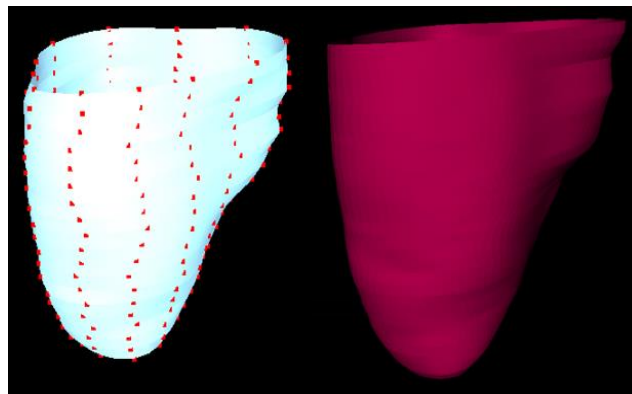
Figure 44: NURBS (left) and tessellated (right) model of the external surface.
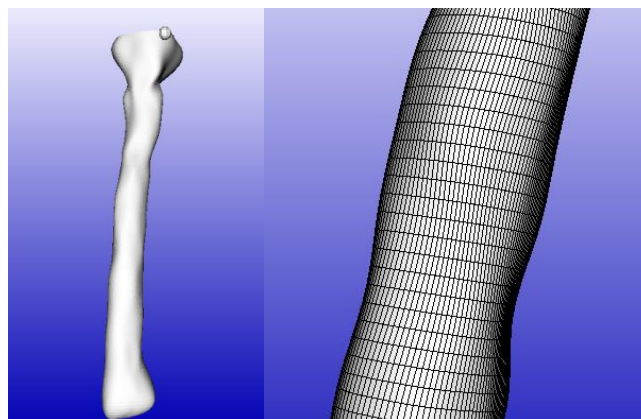
Figure 45: NURBS (left) and tessellated (right) model of the bone.

4.2.4   *Preliminary experimentation and results*

We experimented the automatic procedure with a test case of a transfemoral amputee. We tested the algorithm on different sets of MRI scans of the same patient. External stump surface reconstruction is well performed with every set of MRI; obviously the final model accuracy depends on the resolution of the MRI images. Bone reconstruction, as previously said is more challenging. We found that the best settings of MRI scan for our purpose are the following:

- T2 weighted MRI.

- Slice thickness 2.0 mm.

- Pixel spacing 0.729 mm.

- Resolution 288x288.

In this case, the dimension of the matrix was 288 x 288 x 150 (resolution and number of slices). Based on the scanning parameters, a single voxel had size of 0.729 x 0.729 x 2.0 mm. We performed a normalization operation, so that every voxel had an intensity value between 0.0 and 1.0.

The models obtained have been used to both define a digital model of the transfemoral patient and to design the socket. Figure 7 shows the transfemoral patient's avatar with the reconstructed stump model and wearing the prosthesis.

We compared the results with models reconstructed using a commercial package and following a hand-made procedure. The results have been considered promising, even if the model of the bone is less accurate. Anyway, it meets the requirements of the new prosthesis design framework consisting in an automatic reconstruction without human intervention, and geometrical models suitable for socket design and finite element analysis.

On the basis comparison, we have planned to improve the reconstruction of the bone. The segmentation graph algorithm loses some voxels belonging to the bone, due to high difference of voxel intensity. New features, such as texture characteristics [41], will be introduced during clustering. Furthermore, a shape analysis of the temporary cluster created during the segmentation will be performed to drive the segmentation toward the correct final clusters shape. In fact, a bone has a shape with a high ratio between height and width, and this information could be embedded into the algorithm to make it more effective and fast.

4.3 Procedure with 3D scanning technology inside SMA^2

The whole project follows the low cost philosophy and thus, the choice of 3D scanning systems has been centered on the use of cheap 3D systems on which we can develop our custom software modules for SMA^2

Our aim is to reach the best quality of the 3D virtual residual limb according to precision of measurement and the structure of mesh through the use of low cost IT systems and custom software modules developed within this research activity. To this end the Microsoft Kinect v1 has been selected.

4.3.1 Application

MS Kinect v1 sensor allows scanning the residual lower limb of a patient, exporting a STL file. This tool requires the user to takes special cares to optical occlusions and light conditions. In our tests, the subject is in a comfort and natural standing position and he is not wearing the lower limb prosthesis.

The residual limb must be inclined laterally to allow a complete visual to the sensor without obstructions by the other leg. The subject can lean on tripods placed near him (Figure 46). We consider this position because it emulates the natural standing position and the center of mass of the body is the same. Thus, internal forces are not introduced and soft zones are acquired in relaxed position. In very soft zones (e.g., flaccid part), it is necessary to consider the effect of gravity and this position helps stability and stillness. Actually, an important aspect for 3D scanning of human body concerns imperceptible movements that are impossible to avoid. Even more a natural and stable posture helps to reduce them and, therefore, the scan is more precise. 3D acquisitions with Kinect v1 could be made with commercial software such as Faro Scenect [42], Artec Studio [43] and others, but we preferred Skanect by Occipital [24] for its convenient price-performance ratio. This software, after a preliminary phase of setup of bounding box size, enables to make scan by moving the sensor around the object. In our case, the user has to move the Kinect around patient following, where possible, a circular path. S/He has to pay attention to the zone between legs and eventually around scars. In real time, Skanect software creates a virtual model of the limb scanned. Through colors (green/red) it provides a direct evaluation of which zones are good scanned and vice versa. In this phase it is also possible to apply filters or reduce numbers of points mapped to simplify mesh geometry.

Figure 46: 3D scanning procedure of a lower limb patient with MS Kinect v1.

Anyway, it is important to clean the acquired mesh removing all double points and faces overlapping that could create some trouble. Using this technology, in addiction to geometry, a skin texture is provided that permits to identify scars due to the amputation. This factor helps to creating off-load zones, including scars position.

4.3.2 Comparison and discussion

The presented method could be applied for medical purpose in orthopedic field. Considering lower limb prosthesis, there is no interest to high precision scan of the residual limb. In fact when we do a scan of the

residual limb, we are scanning a soft and deformable component and thus, it does not need a perfect acquisition with a precision of a fraction of millimeter, but it is enough to have a precision of a few millimeters. For this reason we chose MS Kinect v1 (Figure 39.a) that use IR structured light technology to obtain triangular mesh of the scanned object. This choice involves the disadvantage to have a low precision in the reconstruction of the mesh. This means that the reconstruction algorithm of the model have to connect points in several triangular shapes starting from the acquired point cloud. MS Kinect v1 in relation to Skanect software does not provide a homogenous meshed model with the limitation to round the real shape, due to a not precise discretization. A more high-priced scanner could solve this approximation and create a manifold model based on triangles with same areas. This factor simplifies FE analysis, reducing computational time. Anyway, the mesh provided by Kinect is enough clean and accurate only if you pay attention to critical areas to scan, during acquisition phase (i.e. holes, scars, folds, etc.). Figure 47 shows the differences of accuracy between an acquisition with a commercial laser scanner and a Microsoft Kinect v1.

a)   b)
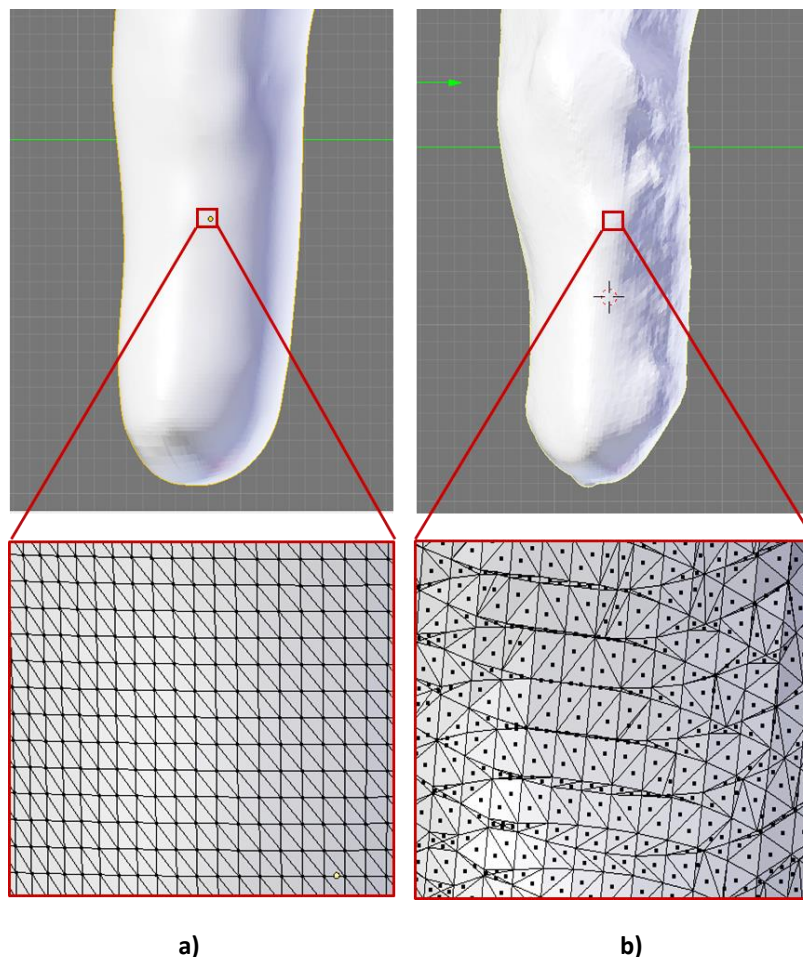
Figure 47: Transtibial lower limb residual model taking with a) commercial laser scanner and with b) MS Kinect v1.

Kinect optical sensor and each other 3D optical scanners have the feature to scan the limb in a relaxed position which does not create or minimizes deformation. This is not possible with MRI exam: it is performed with the patient in a supine position on the examination table for all its duration. Patient limbs are obviously deformed for the contact with the table, but this is not a problem for the aim of the exam

because the internal structure is not modified. MRI external shape of residual limb suffers of a flattening. (Figure 48)

Figure 48: MRI leg cross section: flattening of soft tissues.

Kinect could be also used to evaluate differences between virtual 3D model and real socket. Executing a new scan at the end of the production process it is possible to check the dimension correspondence between the designed and the realized socket. Scanning a socket is not an easy operation because it needs to consider that Kinect captures from a distance from camera lenses of about 50 cm. This limitation requires that user keeps a certain distance from the object and in particular in dark or undercut areas it difficult to scan and create a mesh without any holes. Other problems come from the material of some sockets: they have some areas made of transparent material that creates problems in the scanning phase with Kinect, for this reason it is necessary to temporarily cover them with a dull film.

For example, it may be useful to archive patients' history recording also the shape evolution of the residual limb. In this way, a comparison could be done between old and new socket.

Another useful test is the scan of the 3D printed socket exported from SMA^2 software. This step allows eventually modifying geometry on the 3D printed check socket before realizing the final one.

Chapter 5: Augmented Interaction

This chapter presents the research activities carried out to enhance the interaction between SMA^2 and end-user and make the interaction as much as possible natural. First, a brief overview of low-cost hand-tracking and haptic devices considered for the prosthetic field is introduced. Then, the integration within SMA^2 of selected devices is presented. The main aim has been to experiment the potential and efficacy of emerging interaction devices, mainly used for entertainment, in prosthetic and orthotics domain trying to find a compromise between costs and technical requirements (e.g., precision). Finally, preliminary tests and results reached so far are discussed.

5.1 Hand tracking and haptic devices

In the last decade, many researchers have been involved in researches on how to emulate the five senses through use of computers, [44], [45], in particular, the tactile sense. For example, it is possible to develop surgery simulators to emulate real hand-made operations for practitioners during training processes [46]–[50] or to design customer products [51]. For our purpose we have considered hand-tracking and haptic devices.

5.1.1 Hand-Tracking devices

Several types of 3D motion capture system are able to track the 3D position of hands [52]–[54]. These devices propose an interaction paradigm that allows simulating a real interaction with virtual objects in the virtual scene using the natural motions of hands/fingers. The differences among them consist in how detection of hands/fingers is realized. As already mentioned, we considered low-cost devices

There are different hand-tracking devices on the market, among which Leap Motion device, Intel Gesture Camera, Duo3D and Microsoft Kinect 2.0 (Figure 49).

a)	b)	c)	d)
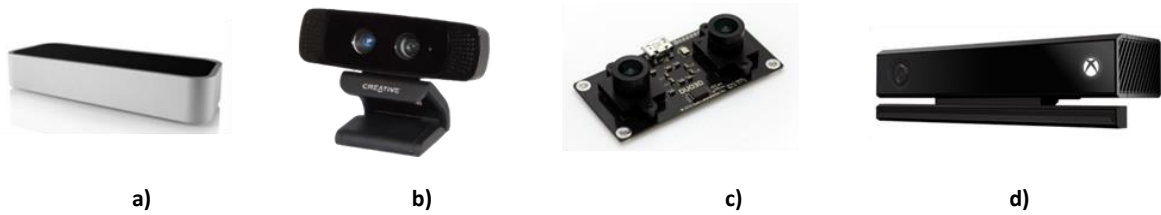
Figure 49: Low cost hand-tracking devices: Leap Motion device (a), Intel Gesture Camera (b), Duo3D de-vice (c) and Microsoft Kinect V2.

The Leap motion device exploits two IR cameras to determine position and orientation of hands/fingers with high precision. Tracking is very accurate and it can be calibrated to map fingertip positions on the screen. Leap motion device has been used in several research works in order to try a new type of interaction with applications [55], [56]. Furthermore, it makes available a Software Development Kit (SDK) to develop new application exploiting augmented interaction. The last version of SDK is already able to recognize a good set of basic gestures (e.g., grabbing, circle made by a single finger and pinch action) [21]. Furthermore, this hand-tracking device can be used with Oculus Rift to increase the quality of user experience into a virtual system. Figure 50 shows how Leap Motion device is used with Oculus Rift device. This solution is able to track hands/fingers according to natural motion of head.

Figure 50: Leap Motion device and Oculus Rift.

Intel Gesture Camera includes RGB and depth camera as well as microphones. It is a good hand-tracking device that is able to track simple gestures in front of the PC screen [57], but doesn't permit to develop user interfaces for complex interaction. Intel gesture camera permits also face analysis and speech recognition.

Duo3D is a device composed by two infrared cameras, an accelerometer and a gyroscope. It is very small and ultra-compact and it's an ideal solution for mobile projects. It is able to track hands/fingers with high precision, but the SDK is available only in pure ANSI C and doesn't offer an initial set of detectable gestures [58].

Microsoft Kinect v2 offers an HD-RGB camera and a powerful IR sensor as well. A large field of view allows this sensor to track object and people closer to the cameras. Just simple gesturers can be tracked by it, such as open and closed hands.

The software development kits (SDKs) made available by mentioned devices permit to create software interfaces with other applications.

The following table describes advantages and disadvantages of the hand-tracking devices considered for an integration within SMA[2].

HAND-TRACKING DEVICES	ADVANTAGES	DISADVANTAGES
Microsoft Kinect v2.0	• Better depth sensing: The new time of flight camera gives a much cleaner depth image and smaller shadows compared to v1 with its structured light approach. • Higher resolution: The RGB camera has an increased field of view and can stream full HD video up from v1's 480p. The depth stream is up to 512×424 at 16 bits per pixel, up from v1's 8 bits. • Better skeleton tracking: The new sensor can now track up to 6 skeletons with 25 joints each (up from v1's 2 skeletons of 20 joints), including joint orientation and finger tracking. • It can detect open and closed hands as well as two finger lasso gesture. • Enhanced IR capabilities: It allows tracking IR reflective objects, while filtering out IR lights. • Enhanced IR capabilities: It allows tracking IR reflective objects, while filtering out IR lights.	• Compared to other sensors the size and extra power adapter are still an issue. • It is only compatible with USB 3.0, and doesn't support most USB over Ethernet extenders, which makes it complicated if the driving computer is away from installation. • Current SDK doesn't support more than one Kinect per PC. • The SDK and driver are only available for Windows 8.
Intel Perceptual Computing	• Smaller and less expensive: The Intel camera (produced by Creative) is smaller than a Microsoft Kinect for Windows sensor, is powered over USB, and is designed to sit on top of most computer displays. • Close-range tracking: It is specifically built for close range tracking, with a range of 0.5ft to 3.25ft (and a diagonal frame of view of 73 degrees). • Hand posture/gesture recognition:	• Getting some of the deeper features (like age and gender detection) to work is a bit tricky. • Due to the close range of the tracking system, hand gestures must be designed such that a user's hand doesn't occlude their own view of the display.

	The SDK allows recognition of hand postures like thumbs up, thumbs down and peace, and gestures like waving, swiping and circling with a hand. Other gestures like grab and release or pan and zoom can be implemented by examining the openness of the palm and fingertip positions. • Facial analysis: The Intel SDK provides capabilities for face tracking, recognition and detection as well as age and gender determination. Expressions like smiling and winking can also be detected. • Speech: Developers can leverage speech recognition by specifying a predefined list of commands, or multiple lists constituting a grammar • Raw data: Developers have access to raw color and depth data from the sensor along with a confidence map to account for distance and light conditions.	
Leap Motion device	• Gesture recognition: Finger tracking is fast and accurate. • Even smaller and less expensive: The device is physically very small and is inexpensive. • Developer friendly: The "Airspace" application store provides a way for developers to market and distribute Leap apps. • Framework support: The Leap supports a number of frameworks, including .NET, Processing, Cinder, etc. • Compatible: Works on both Mac OS and Windows.	• Sensing range is fairly limited. • Only fingers are tracked. There is no skeleton or face tracking. • No access to the raw sensor data.

Table 1: Hand tracking devices used with SMA[2].

In this research work, we pay attention on Leap Motion device and its SDK because it allows us to simply define a set of gestures as well as basic hands/fingers tracking.

5.1.2 Haptic Devices

Tactile perception is related to the sense of touch, which includes the feeling of pressure, vibration, temperature and pain [46]. To emulate part of tactile sensations using computer, many haptic devices have

been designed for different industrial applications [59]–[61]. Most of them incorporate only force feedback component of touch and, thus, providing mainly kinesthetic and proprioceptive sensations [62]. Some examples are the well-known PHANTOM device from Sensable Technologies and CyberGrasp, from CyberGlove System LLC [63].

Analyzing the traditional socket design process, the prosthetists use their hands to manipulate the socket shape and often rely on the sense of touch. They use their hands to evaluate residual limb tonicity and manipulate/modify accordingly the socket shape. To replicate this operation using sense of touch, haptic device for managing the interaction with the 3D model of the residual limb could be adopted. In fact, haptic interaction could help users to regulate better the entity of the deformation: the more the user presses on the haptic device, the deeper is the deformation. In the prosthesis design field, on the market we can find several prosthetic CAD/CAM systems (see §2.3), but only Rodin (www.rodin4d.com) proposed a CAD system (Rodin4Design software) to provide the user force-feedback during virtual carving and sculpting of the 3D models.

Haptic gloves, such as the CyberGrasp, could be suitable since they permit to replicate the behavior of the hands interacting with a real object, but, on the other hand, they are complex, expensive and not affordable for small orthopedic labs. Therefore, among the different available solutions, we have considered low cost haptic devices since costs could be a key issue for small orthopedic labs. Some examples are Novint Falcon and low-cost PHANTOM Omni® device. For our purpose we selected the with Novint Falcon device (http://www.novint.com/). The Novint Falcon lets you control a 3D environment in three dimensions, and also lets you feel high-fidelity three-dimensional force feedback. The Falcon controller moves right and left, forwards and backwards, like a mouse, but also moves up and down.

In addition, a haptic mouse specifically conceived and tested for our application, specifically a haptic mouse, has been realized.

5.2 Hand-tracking devices for socket design

In this section, we focus the attention on the development of a Natural User Interface (NUI) for hand-tracking devices. NUI is an emerging concept in Human/Computer Interaction that refers to an inter-face that becomes invisible to its user with successive learned interactions related to natural human behavior. As stated in [64] "The word natural is used because most computer interfaces use artificial control devices whose operation has to be learned. A NUI relies on a user being able to carry out relatively natural motions, movements or gestures that they quickly discover control the computer application or manipulate the digital content". In other words, designed NUI has to be able to make user experience very simple and comfortable. Preliminary research works have been carried out in order to understand and consequently solve matters related to NUIs for hand-tracking devices [65]–[68].

NUI design for hand-tracking devices has to be based on definition of rules that are useful to design an intuitive interaction for a generic application. Leap motion developers subdivided these rules as follows [21]:

- Ergonomics, Posture and Environment. We have to consider how much fatigue the interaction creates at arms and shoulders.

- Virtual Space Mapping and Interaction Resolution. Every gesture has to be mapped according to 2D screen resolution and 3D virtual environment of the application.

- Dynamic Feedback and Gestalt Intermediates. The interface has to give user the best feedback in order to understand what happened using a particular gesture.

- Interaction Space: Separation, States and Transitions. User has to be able to understand which type of action is activated during interaction.

In particular, we paid attention to different aspects mainly related to ergonomics and ease of use, which are the most important features to recreate a better user's experience with augmented interaction.
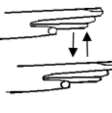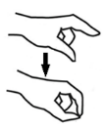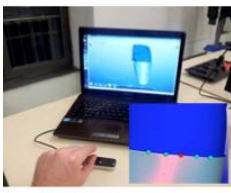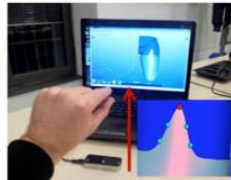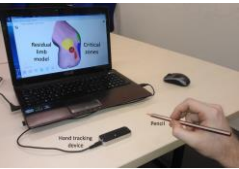
5.2.1 Gestures definition

Since the release of low-cost hand tracking devices, various researches have been carried out to identify the best set of gestures [68], [69]. The main aim has been the creation of natural user interfaces (NUI. In our case, the terms "natural" refers to the use of computer devices that are able to track natural gestures of hands/fingers. When developing a gesture interface, the goal is to implement a gesture interface that is very efficient for the specific context. We started from the traditional operations performed by orthopedic technicians in order to identify a set of gestures adequate for socket design. In fact, as described in previous Chapters, the technicians continually use their hands to carry out different actions to shape the socket positive cast around which the physical prototype is thermoformed.

From the analysis of preliminary results and according, we have identified the set of gestures of interest for our application. In particular, two types of gesture have been identified according to both modeling operations and actions for basic interaction with a generic 3D virtual environment (Table 2).

Other gestures have been added in order to give a better feedback to the user during interaction with SMA. In particular, there are two available interaction styles; Camera and Modification. Camera style rotates 3D model following the orientation of a palm. In this modality, other gestures are disabled. Modification style permits to execute actions relative to socket design according to the gestures previously described.

To switch between the two modalities, one has to put a hand above the sensor with thumb and index finger extended and, then trace a circle in the air, as depicted in Figure 51a.

ACTION	GESTURES	NOTES
BASIC INTERACTION		
Moving from one modeling tool to another		Horizontal movements of single finger or vocal commands like the words "previous" and "next"
Moving selection cursor		Free movement. The palm is parallel at screen
Rotation of 3D models		The palm is parallel to the desk
Zoom		Using both hands. The palms is parallel at desk
Switch between basic interaction and modelling operations		Neutral Gesture. The hand stays motionless for three second with palm parallel at desk.
MODELLING OPERATIONS		
Selecting a single point of 3D shape		For example, in the prosthetic field, the technician sketchs the trim-line along upper part of negative model of the socket. S/he can modify the curve of virtual trim-line by pinching its points that can be moved to the correct position along the residuum.
Marking parts of a 3D model		For example, in the prosthetic field, the technician usually marks critical zones using a pencil.

Table 2: Set of defined gestures.

To move the hand without interacting, the user has to face the palm to the screen. In this way you can find a more comfortable position or rest your hand without com-promising the current state of the application (Figure 51b).

a)   b)

Figure 51: Gestures used to switch between the Camera and Modification modalities.

The user can always check the confidence of data coming from the Leap Motion device. This percentage is displayed at the bottom of the software interface. This percentage permits to understand which the state of the detected action is. Indeed, if percentage is under 20% the detected gestures is considered NOT_DETECTED; while percentage is between 20% and 65% a warning icon appears on left lower corner of screen. If confidence is over 65% user can interact with high precision.

In order to provide additional feedback there is an icon at the right bottom corner of the screen showing which hand gesture has been detected (Figure 52).
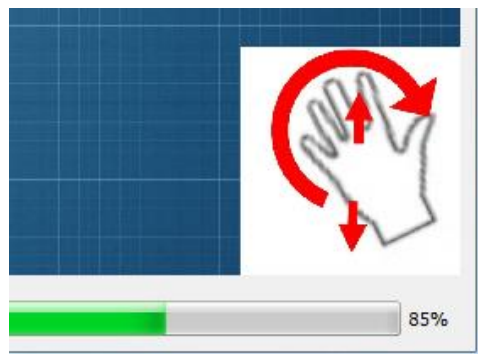
Figure 52: Icon and progress bar into SMA to show state of interaction using Leap Motion device.

The gestures are few because the user has to learn how to interact with SMA in a very simple way. The difference among gestures allows user to distinguish each action from the other ones and thus, the use of augmented interaction appears simple and intuitive during user's experience.

5.2.2 Software development

The modularity of the software architecture of SMA[2] permitted to easily add a new module, which exploits Leap Motion device in order to interact with 3D virtual environment using hands/fingers.

SMA[2] communicates with Leap Motion device through its SDK and applies the appropriate action, which has to be executed according to retrieved data. In order to use Leap Motion device into SMA, we exploited a set of VTK classes, which makes available methods to interact with the 3D scene of application. The interaction handler is a subclass of the existing VTK class `vtkInteractorStyle`, i.e., `vtkInteractorStyleTrackballcamera`; while `SensorInteractorStyle` is an abstract class for

augmented interaction. It is a subclass of `vtkInteractorStyleTrackballcamera`, which represents the interactions by moving the camera. In our class some methods are over-ridden and some abstract methods are introduced (Figure 53).
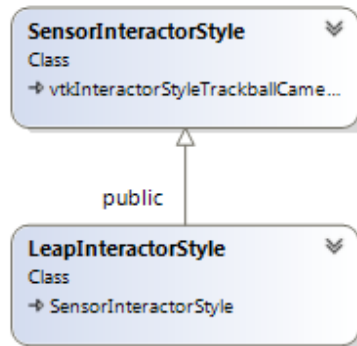
Figure 53: UML class diagram of software interface for exploiting Leap Motion device within SMA.

Some of the most important fields and methods are described in the following:

- `enableSensor()` and `disableSensor()`: abstract methods that enable/disable the interaction when user is using a hand-tracking device.
- `transformVector(double* vector)`: method for transforming a vector from the sensor coordinate system to the VTK scene coordinate system.
- `OnTimer()`: abstract method redefined in this class and implemented in the specialized class. This method is executed every 50 millisecond in order to query hand-tracking device and execute the correct action according to detected gesture.

This class has been extended with another class in order to use Leap Motion device. `LeapInteractorStyle` contains a set of methods and instructions to exploit Leap motion SDK. Some of the most important fields and methods are described in the following:

- State: its value may be either `CAMERA` or `MODIFICATION`, which depends by the current interaction mode.
- `NextMode()`: method that switches between interaction modes.
- `leapController`: Leap Motion instance that communicates with the device. From this object we get all information about hands and movements through software instructions of Leap Motion SDK.

In these methods, there are the instructions, which permit to give information to the user in order to understand what happens during the interaction, such as quality of interaction, interaction mode and state of each action.

5.2.3 Test and Results

The system has been tested with ten volunteers to evaluate system performances especially with regards ergonomics, ease of use and precision. The testers were six male and four female with different levels of experience (i.e., beginners and experts) regarding the use of hand-tracking device. First a demo was carried out to show how to execute modeling operations by means of the Leap Motion device. Then, each tester performed the three main basic operations until s/he was able to interact in natural way without the help of technical staff. The modeling operations were: basic interactions with 3D environment (e.g., zoom in and out), marking critical zones (Figure 54.a) and trim-line definition (Figure 54.b and Figure 54.c).

a) **b)** **c)**
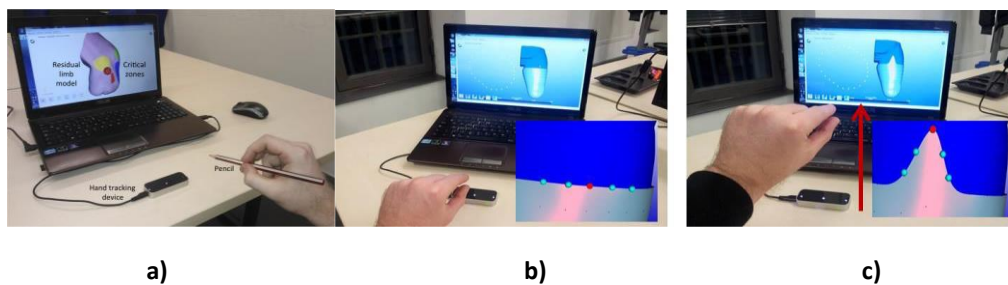
Figure 54: Marking critical zones of residual limb (a), pinch action to model socket shape (b and c).

Testers with a low level of experience found very useful the icons showing the detected gestures but the gestures should be repeated from 5 to 8 times to be properly executed by the user and detected by the system especially for tasks involving 3D geometry modifications. Experts were able to correctly perform the modeling tasks with no more than 2-3 trials.

Regarding ergonomics issues, the new version of the NUI and related gestures al-lows an adequate and long lasting interaction without fatigue at the arms.

The icons permit to simply learn how to interact with SMA[2] and have been considered helpful mainly by beginners. On the other hand, experts have appreciated the progress bar while beginners were more concentrated on gesture execution and detection.

5.3 Tracking plug-in [70]

From the analysis of tests carried out to understand both potentialities and limits of hand tracking devices, a set of common features have been identified. This means that it is possible to define a design pattern that could permit a fast implementation of interface for hand tracking interaction. Therefore, a software tool to automatically generate a basic software interface for mid-air interaction starting from a standard software setting for the considered hand-tracking device. Once developed the target application, the plug-in, named tracking plug-in, permits to easily implement the software interface for different hand-tracking devices and compare their potential and performance according to the considered applicative context.

The above-described set of gestures has constituted the starting point for the development of the tracking plug-in

5.3.1 Plug-in development

The solution is based on open-source software tools that allow the automatic code generation in a simple way after having defined the target application using a meta-model language, in our case the UML class diagram. These tools are totally based on the Eclipse platform [71], a free integrated development environment (IDE), which contains a base workspace and an extensible plug-in system to customize the environment. Written mostly in Java, by means of various plug-ins, Eclipse can be also used to develop applications in other programming languages, among which C, C++, Fortran, JavaScript, Python, and R. In particular, it has been used the plug-in Acceleo [72] that includes tools for automatic and incremental code generation from Eclipse Modeling Framework (EMF) based models. Incremental generation gives the ability to generate a piece of source code and, then, modify the automatically generated code according to the specific needs of a particular context. Acceleo also allows:

- Code generation from any kind of UML 2.0 meta-models and even custom meta-models.
- Customization with user defined templates.
- Generation of any textual language (e.g., C++, Python and Java).

The module, named tracking plug-in, has been defined through the UML class diagram, which has been translated in an Acceleo module. For the development, we focused the attention on some technical aspects (e.g., interaction events and 3D object management) meaningful to automatically generate the software interface. Figure 55 shows a simplified version of the defined UML class diagram and how considered technical aspects are correlated:

- *HandTrackingDevice* is the reference to the SDK of considered hand tracking device, i.e., Leap Motion, Intel Gesture Camera or Kinect v2.
- An *EventsHandler* to interact with 3D objects. The auto-generated source code has to include a set of instructions to allow the management of interaction events through the definition of a method, such as *OnTimer()*. This method continuously controls if hand-tracking device detects a *Gesture*. Usually, 3D environment is available through a *SceneManager* object that is based on the use of an SDK, such as VTK, pure OpenGL or Qt.
- *Gesture class* for the technical features of each gesture listed in Table 1. They have to be implemented as a set of attributes and operations (e.g., activation, execute and release). Each instance of gesture class has to assume a value within a set of constants defined by the enumerative *GestureType,* which includes all gestures listed in Table 2.
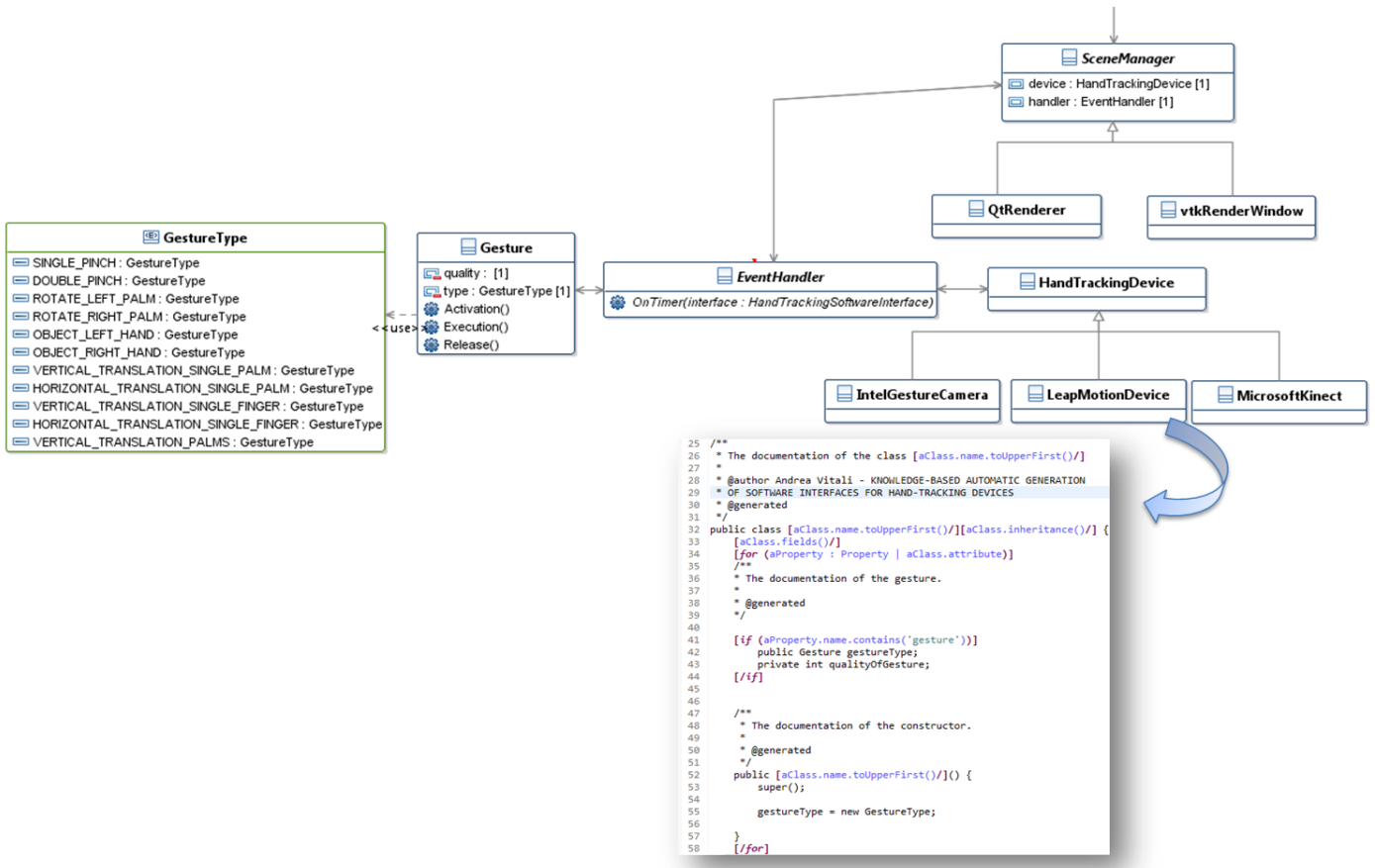
Figure 55: UML class diagram of tracking plug-in.

Figure 55 also portrays a shortcut of the tracking plug-in code developed using Acceleo modules, while Figure 56 how the plug-in works.

Starting from the UML class diagram of the target application, the tracking plug-in generates the software interface (`HandTrackingSoftwareInterface` class), which is directly linked to the target application. The developer has only to specify the hand tracking s/he wants to use and the programming language for the software interface.

The developed plug-in automatically generates functions and methods according to chosen gestures (i.e., each function automatically defines a set of parameters used for detecting a specific gestures), but the developer has to write specific source code related to a specific applicative context, such as socket design.

Three hand-tracking devices have been considered as well as three programming languages: C++, Python and Java. Anyway, the plug-in could be easily extended to new devices and other programming languages.

UML DIAGRAM OF THE TARGET APPLICATION

```
25  /**
26   * The documentation of the class [aClass.name.toUpperFirst()/]
27   *
28   * @author Andrea Vitali - KNOWLEDGE-BASED AUTOMATIC GENERATION
29   * OF SOFTWARE INTERFACES FOR HAND-TRACKING DEVICES
30   * @generated
31   */
32  public class [aClass.name.toUpperFirst()/][aClass.inheritance()/] {
33      [aClass.fields()/]
34      [for (aProperty : Property | aClass.attribute)]
35      /**
36       * The documentation of the gesture.
37       *
38       * @generated
39       */
40
41      [if (aProperty.name.contains('gesture'))]
42          public Gesture gestureType;
43          private int qualityOfGesture;
44      [/if]
45
46
47      /**
48       * The documentation of the constructor.
49       *
50       * @generated
51       */
52      public [aClass.name.toUpperFirst()/]() {
53          super();
54
55          gestureType = new Ges
```

HandTrackingSoftwareInterface

Figure 56: Tracking plug-in.

5.3.2 Tests and results

The plug-in has been primarily tested with SMA^2 and the software interface has been generated for the Leap Motion device, Intel Gesture Camera and Microsoft Kinect v2. In fact, these devices make available basic SDKs to simplify the software development of custom applications. We decided to export the source code for both C++ and Java development under Microsoft Windows.

Figure 57 shows the high-level UML class diagram of SMA^2 modeling tools used to automatically generate the software interface between SMA^2 and the hand-tracking devices.

Since, SMA^2 is defined as a set of virtual tools to simulate operations traditionally performed by the prosthetists to shape the socket, the abstract class *Tool* has been implemented. It makes available properties and operations as follows:

- *3DObject*: this property manages 3D object that has to be modified to model the socket.
- *Gesture*: this property allows associating a particular gesture to interact with 3D object using the considered hand-tracking device.
- *SceneManager*: This property manages 3D rendering and makes available an instance of the event handler class. It is directly linked to either VTK or Qt.

- *DoWork()*: this operation executes actions on *3Dmodel* when the associated *Gesture* is detected by events handler of *SceneManager* object. This is possible since the event handler is directly connected to software interface in order to communicate with the hand-tracking device.
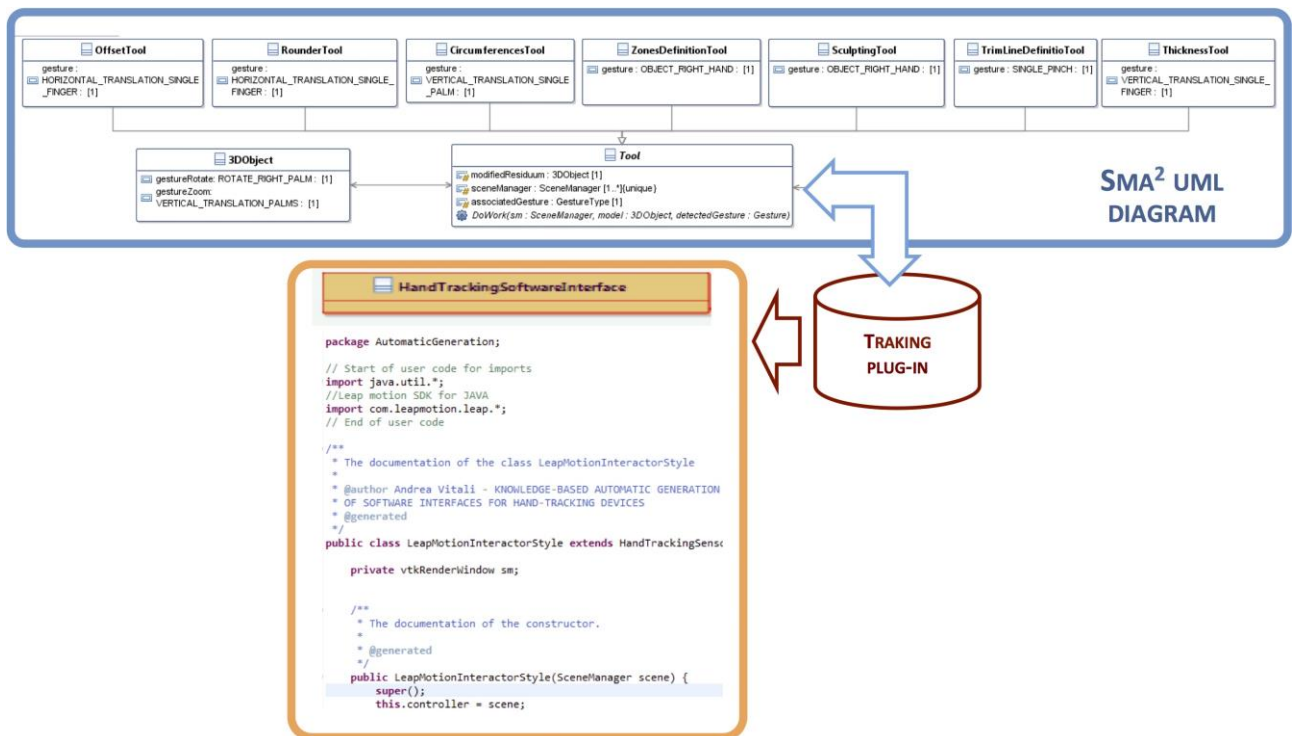
Figure 57: UML class diagram of SMA2 virtual tools.

An instance of *Gesture* is associated to each modeling tool. For example, the tool named *ZonesDefintionTool* allows the user to mark critical zones of socket using a thin object held in hand, such us a pen or a pencil. Therefore, it has a *Gesture* instance to which is associated the constant of *GestureType* that could be OBJECT_LEFT_HAND (mark a residuum/socket area with left hand) or OBJECT_RIGHT_HAND (mark a residuum/socket area with right hand). Moreover, there is the *3DObjec*t class that has two gestures for zoom and rotate operations.

Tracking plug-in analyzes the SMA² UML model, detects the presence of a gesture instance and automatically generates the source code in C++ to manage the interaction with considered gesture. With regard to the Leap Motion device, the auto-generated software interface and the customized one previously described have been compared and tested. The software structure is very similar, even if the customized interface contains instructions to improve the quality of the interaction with the socket 3D model. All gestures have been tested and no significant differences have been detected. The most relevant difference between the two software interfaces has been the development time. In fact, the auto generated software interface required five days of software development for adapting modeling tasks of SMA² for mid-air interaction. Custom interface required eight days since the developer had to set basic parameters of Leap motion device (i.e., field of view of device as well as the correlation between the

coordinate systems of the Leap motion device and the 3D environment). It is very important highlight that the parameters' setting has to be done for each hand-tracking device used for interacting with the application. Using tracking plug-in, these parameters are automatically set for each mentioned device.

Figure 58 shows two steps of residual limb rotation using rotate gesture (ROTATE_RIGHT_PALM *GestureType*) by using both Leap Motion device and Intel Gesture Camera.

| a | b | c |
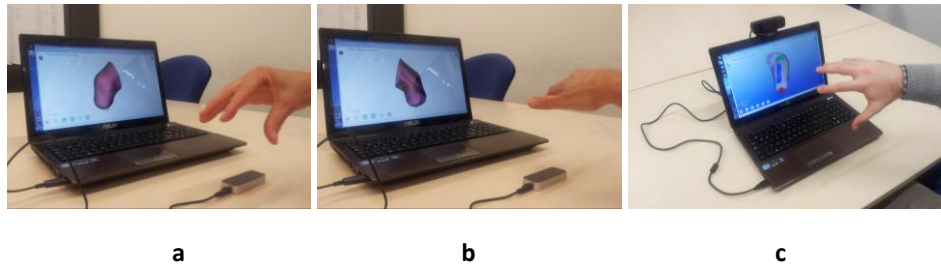
Figure 58: Rotate gesture with both Leap Motion device (a,b) and Intel Gesture Camera (c).

The software interface for Leap Motion device is better than one for Intel Gesture Camera. This because the modular structure of Leap Motion SDK is better than that one of Intel Gesture SDK, which requires many instructions in pure C.

The Kinect v2 has been mainly considered to verify that the tracking plug-in can be used for different devices; in fact, its hand tracking capabilities are less precise compared with the other two devices.

A preliminary test for an application in the textile/clothing context has been also carried out to verify the applicability in an industrial context. Two simple tests have been considered to emulate tailor's operations when s/he takes customer's measurements and manipulate 2D patterns composing a cloth using only the Leap Motion device and few gestures. A simplified and preliminary UML model of the target application for cloth modeling and simulation has been realized. The tracking plug-in has been used, as done for SMA, to successfully generate the software interface to manage the interaction with the Leap Motion.

5.4 Haptic devices for socket design

5.4.1 Novint Falcon for socket design

Regarding the haptic interaction, to test the feasibility of this kind of interaction we developed a software module that exploits the Novint Falcon device and a real time finite element model (Figure 59). For this last one we made reference to the model described in [73] developed for surgical simulation applications.
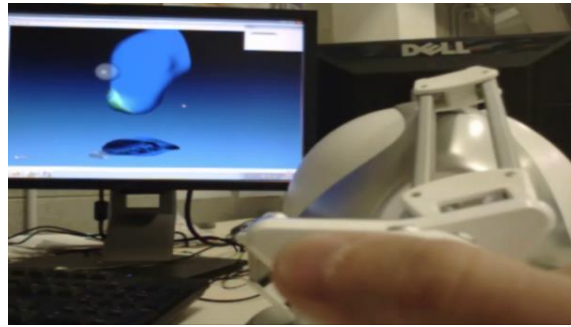
Figure 59: Novint Falcon used to emulate adding/removing chalk along the positive model of the residual lower limb.

The mathematical model consists of a mesh of tetrahedral elements, created in automatic way through a software module that drives a commercial FE analysis tool, in our case Abaqus. The model obeys the equation:

f=Mu''+Du'+Ku

where:

D   is the damping matrix.

M   is the mass matrix.

K   is the global stiffness matrix.

f   is the vector of the external forces applied to the model.

u   is the vector of the nodes displacements.

K is computed by assembling the stiffness matrix of each tetrahedral element, considering the mesh connectivity.

To solve the equation we use an explicit approach [73]. The result of the equation drives the haptic device, so the equation has to be resolved in real time. To provide a realistic simulated force-feedback, the haptic loop has to be rendered at least with a frequency of 1000 Hz (instead of the video rendering that usually requires a minimum frequency rendering of 30 Hz).

We assumed lumped masses at nodes, so M and D can be diagonal matrices. Some terms of the equation can be pre-computed to save simulation time. Furthermore, in the pre-processing phase we perform matrix condensation on the K matrix that consists in eliminating rows and columns that correspond to the fixed nodes (e.g. those near to the inner bone).

After pre-processing optimization, the only term of the equation that has elevate computational cost is the multiplication Ku. In order to keep the multiplication times low we exploited GPU computation, in particular CUDA. K is a sparse matrix, so we used the CUSPARSE library (a sub-library of CUDA) that provides fast and optimized matrix-vector multiplication through GPU.

Finally, in order to provide realistic visual feedback to the user, the tetrahedral mesh is not visualized during the video rendering, but it drives a high-definition mesh. Each vertex of the high-definition mesh is connected to the nodes of the tetrahedral mesh with a given weight; therefore, the tetrahedral mesh can be considered as a skeleton that influences the deformation of the high-resolution mesh.

The haptic force feedback helps the technician to manipulate the geometric model as s/he usually acts using hand. Prosthetist considered interesting these first tests, even if the haptic device permits to emulate only the interaction of one finger. The optimal solution should be a haptic glove, but, at this stage of the research, our main goal has been to test with the orthopaedic technicians the potential and feasibility of such an approach.

5.4.2 Low-cost haptic mouse for socket design

To build low-cost and easy to use device, we considered, at this stage, sufficient the possibility to interact with one finger, by which the user can feel the softness/hardness of the residuum surface and perceive better the virtual object.

The haptic mouse prototype is basically a traditional mouse device enhanced with a servomotor and a pressure sensor pad (Figure 60).The servomotor is the Hitec HS-225BB high speed model, with 0.14 sec/60 deg. transit time; while, the pressure sensor is the Flexiforce piezoresistive force sensor built by Tekscan. If pressed, its resistance changes from infinite to about 50.000 Ohm. Figure 3 shows the haptic prototype.
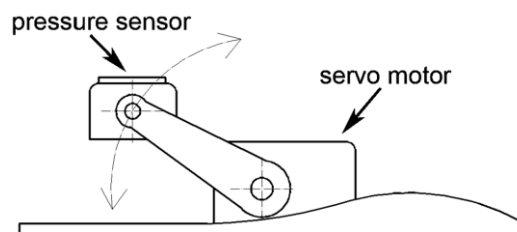
Figure 60: Haptic mouse operating principle.
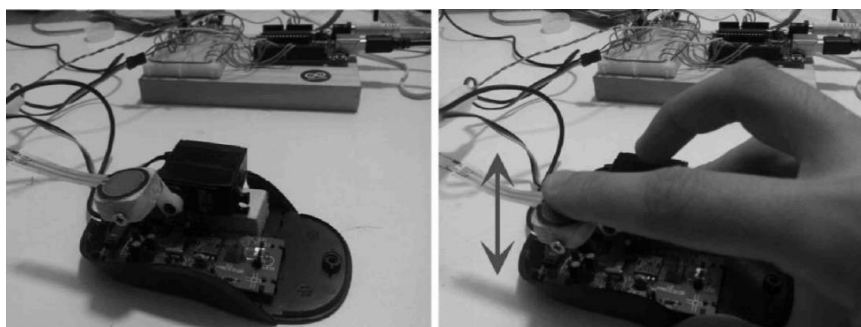
Figure 61: Haptic mouse prototype.

The pressure sensor pad works as input device, measuring the force the user applies with her/his finger. The servomotor works as output actuator setting its position according the physic-based model

computations. To get data from the pressure sensor pad and to drive the servomotor, we exploited the Arduino board linked to the computer running SMA^2 (Figure 62). The communication between Arduino board and the computer is through USB port, with serial protocol.
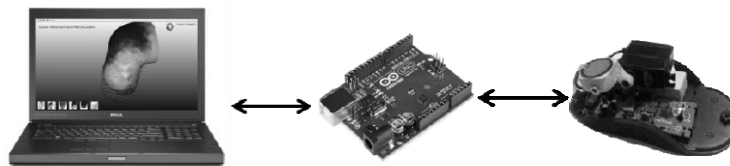
Figure 62: Computer - Arduino - Haptic mouse workflow.

The user can move the sphere cursor of the Sculpt Tool by moving the mouse, as s/he would do with a traditional mouse. Putting one finger on the pressure pad, the user can push the pressure sensor and the pressure value is read by Arduino and transmitted to SMA. The embedded physic-based model computes the deformation created by the applied force and calculates the new servo motor position, which is moved by Arduino board. The more the user pushes the pressure pad, the deeper is the deformation of the surface and the servo motor position is lowered.

The input data from the pressure sensor pad are smoothed by keeping a FIFO (first in first out) buffer with the last collected samples and averaging them.

In order to compute the deformation of the surface according to the user input, we exploited the physics model used with Novint Falcon.

5.4.3 Preliminary Test and Results

We experimented the haptic devices to create the shape of the socket starting from the residual limb model and to emulate the operation of adding and removing material from the plaster cast. The 3D model of the residual limb of a transtibial (amputation below knee) patient has been acquired by MRI (Magnetic Resonance Imaging). For materials characterisation (bones and soft tissues) we considered data found in literature [74].

Figure 63 shows the shape modelling using the Novint Falcon.
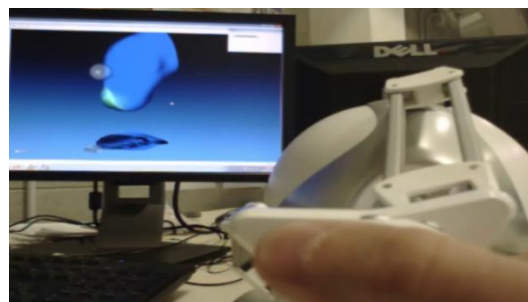
Figure 63: Use of Novint FALCON to interact with the 3D model of the socket.

Regarding the haptic mouse we first tested it only with the servomotor and without the pressure sensor to evaluate how the haptic mouse replicates the curves and slopes of the residuum virtual surface. By moving the mouse on the table plane, the servomotor sets its height position according to the height of the surface of the virtual model. In case of fast movements, we noted a small delay to update the servomotor position due its speed limits. However, in case of normal movements in a limited area, the servomotor follows the virtual surface with good approximation. Afterwards, we tested the complete system including the pressure sensor pad. We found some fake peaks in the detection of applied forces; however the filtering software routine limited the problem. Figure 64 shows an example of shape modeling using the haptic mouse.
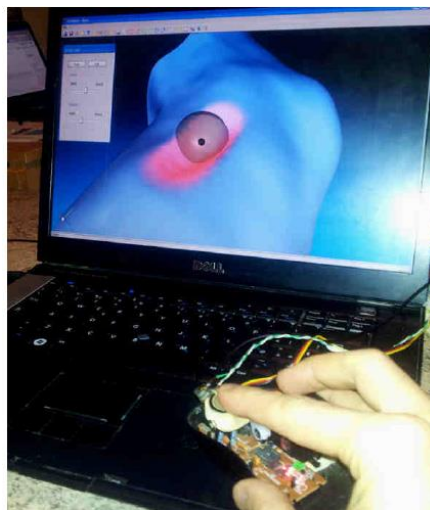
Figure 64: Use of the haptic device to interact with the 3D model of the residuum or of the socket.

The haptic force feedback helps the technician to manipulate the geometric model as s/he usually acts using hand. Prosthetist considered interesting these first tests, even if the haptic device permits to emulate only the interaction of one finger. However, further enhancements are required in order to develop a haptic mouse suitable for the considered context and able to satisfy both economic and technical requirements. For example, the haptic mouse can be improved add more couples of servomotors and pressure sensors in order to provide haptic force-feedback with multiple fingers. This should permit to perceive better curves and slopes of the surface.

The development of a new haptic device targeted for the prosthetic environment represents a possible future trend of development of this Ph.D thesis.

Chapter 6: Socket Manufacturing

This chapter presents the use of additive manufacturing (AM) technique for the realization of the physical socket. First, a brief description of the adopted AM technology is introduced, then the state of the art of infill techniques. Finally, how mentioned technique has been applied and new SMA[2] features to exploit AM are reported.

6.1 FDM and State of Art of Infill methods

Additive manufacturing technologies are changing the traditional approach to design and manufacture products. The global market is full of 3D printing machines, materials, and other variants. All contexts are interested by AM such as industrial, design, medical, scientific or daily scopes. As a consequence to this large diffusion, a large variety of materials is available on the market: plastic filaments, bio-materials, ceramic, clay, cork are just a few filaments that can be printed nowadays. We are interested, in particular, to FDM (Fuse Deposition Modeling), a 3D printing technology that permits to create an object, extruding fused material on the printing plane by layers from the bottom up. It is created by overlapping a lot of layers. The typical structure of a FDM printer is illustrated in Figure 65. In the pre-processing of the 3D printing phase, the model exported from a CAD system or a scan program needs to be manifold: it means that the meshed model must not have holes, degenerated triangles and overlapped faces. After that, a slicing algorithm subdivides the object in the sequence of layers to overlap. PLA (Polylactic acid or polylactide), biodegradable thermoplastic polyester derived from renewable resources and ABS, are the most used materials.

There are also some special filaments for particular uses such as flexible filaments made by a rubber material permitting great deformation, or water-soluble filaments useful to realize supports to be easily removed. An interesting challenge is the possibility to use and integrate multiple materials in the same object; in this way, it is possible to realize products with different composition areas.

Exploiting the potential of flexible materials makes possible to integrate them with classical materials using a 3D printer with multiple extruders. This allows creating models with soft and hard parts.
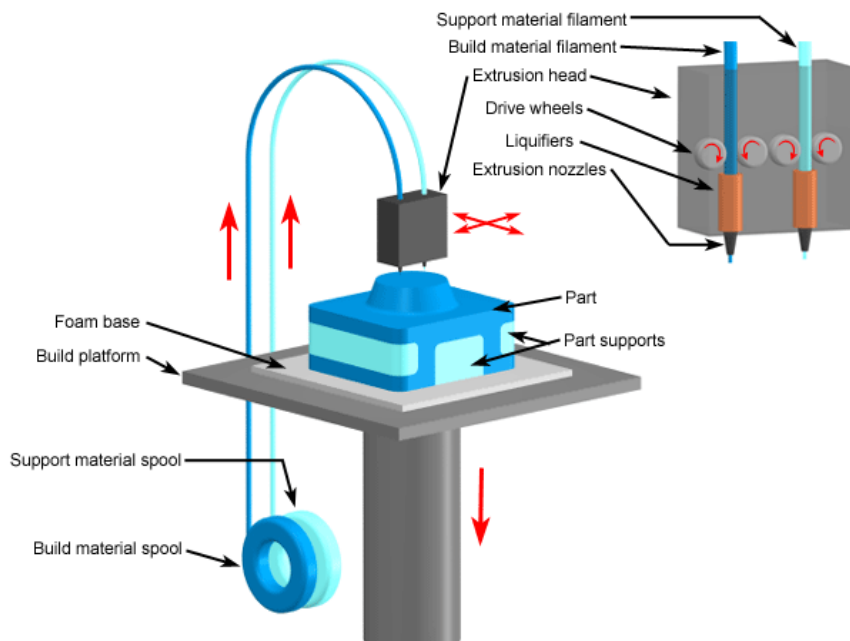
Figure 65: FDM operating principle.

The infill is another fundamental element for 3D printing; in fact, it could offer some improvements with regards to different aspects, such as mechanical properties, filament saving, objects lightening and internal structure optimization. In particular, internal structure confers to the printed element rigidity and deformability that are strongly related to external stress distribution. Rectangular shape, honeycomb or concentric path are only a few of internal infill available with common 3D printers (Figure 66). In addition to classical infill patterns, there are some research works that study algorithms to optimize and customize infill.

FDM printing could seem an easy way to print, but there is a large set of parameters that influences the result such as heated bed temperature, infill ratio, printing temperature, layer height and printing speed. One of the most representative phenomenon is the warping that appears and damages the printed object. We can see it in the detachment of the first layer from the printing plane caused by a bad control of fan speed and printing temperatures. Therefore, it is important to take care to its cause or rather the fast cooling during the deposition process, using some expedients to avoid it like a brim, or a raft. These two elements reduce the warping introducing material or reducing the contact area of first layer.
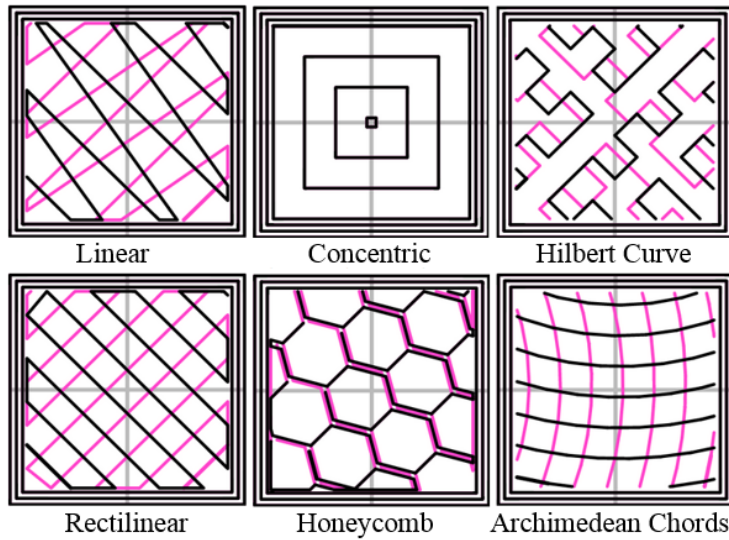
Figure 66: Typical infill patterns for 3D printing.

6.2 Optimizations methods for additive manufacturing

Among several studies, we proposed the most interesting literature contribution concerning infill creation to characterize the properties of an object. In the following subsections, the selected methods have been subdivided according to the techniques used.

6.2.1 Cross sectional analysis

Li et al. [75] developed an innovative method whose aim is to optimize the internal structure of 3D object according to their real usage. They started from a meshed geometry derived from a CAD system or other modeling software and they execute a cross-sectional stress analysis to identify zones in which fracture could occur (Figure 67). This process is based on the Euler-Bernoulli assumption, thus, avoiding the use of Finite Element analysis. In this project, they introduced a method to obtain an approximated neutral axis perpendicular to each section that allows calculating cantilever structural stress.
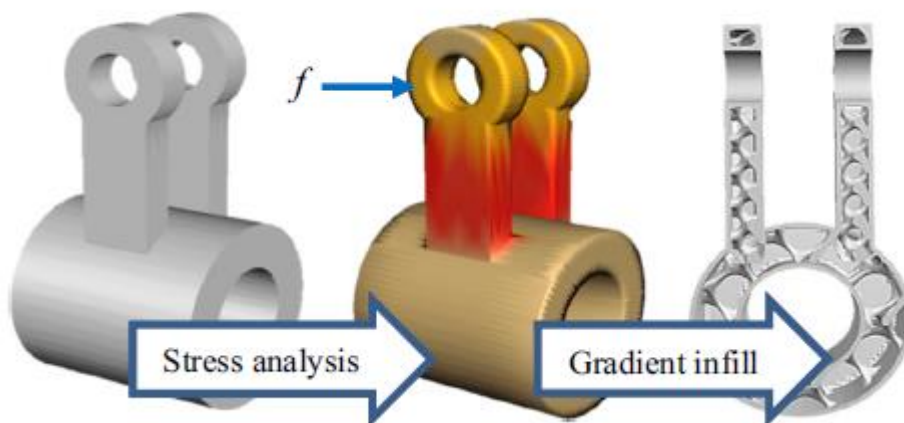
Figure 67: Process steps of optimization infill with cross sectional method [75].

After that, they estimated the density by evaluating the moment of inertia: they constructed the relationship between geometry shape parameter of the cross section and the moment of inertia. Internal structure will be made of porous infill and its density is regulated according to stress applied that is inversely proportional to the moment of inertial of cross section. Infill ratio changes between different volumes of the same object, divided according to cross sections. A local gradational infilling made by a mathematical algorithm and Boolean operations to integrate the shell model and the porous structure are executed. The output file is a STL file that is usable with commercial 3D printers.

Umetani and Schmidt [76] in an Autodesk research illustrated a similar method based on weak sections of the object. They introduced a method to give a real time feedback to the designer about critical zones and also to optimize the creation of 3D printed objects. In this way, they offer a way to identify the best infill direction that guarantees the highest resistance to the applied loads. If there are several weak sections, maybe it is not possible to find only a single orientation of the model so it required privileging the weaker section. Therefore, this solution does not modify infill ratio and, thus, the quantity of material, but it controls the growing direction of the model, as shown in Figure 68.

Both methods have the most important advantage in their rapid execution and this is because they analyze only the weak sections of the object. By the way, this solution is not yet addressing the important approximation of the structure, e.g., improving of other properties such as tensile stress, torsional stress and center of mass. These and other research groups are already working on such issues and improvements are expected in a near future.
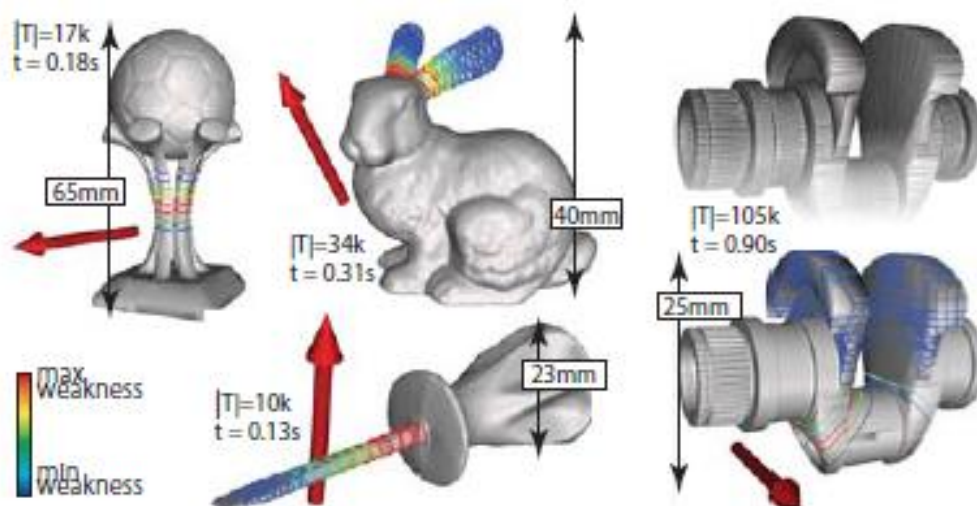
Figure 68: Examples of optimized 3D printing direction [76].

6.2.2 Medial axis tree method

Zhang et al. [77] defined an interesting approach to internal supporting structure for 3D printing. Their work is based on the use of medial axis concept: biologically, it is the skeletal structure that absorbs and disperses loads applied to the object; mechanically, it is the axis of the volume that represents the support of the structure. The 3D printed object has a structure made by three different parts: the medial structure, the boundary framework and the group of connecting bars. The algorithm used to reach the aim is at first the scale axis transform developed by Giesen et al. [78] to extract the medial axis of the structure; then, the Restricted Centroid Voronoi Tessellation allows creating the external hexagonal frame structure. The last process consists in linking all vertices of the hexagonal structure to the corresponding points on the medial axis. After that, they compute the optimization process of the medial axis tree according to external loads applied. The process is divided into three parts: convex hull analysis and load combination, mechanical analysis and radii classification and, topology and volume optimization (Figure 69).
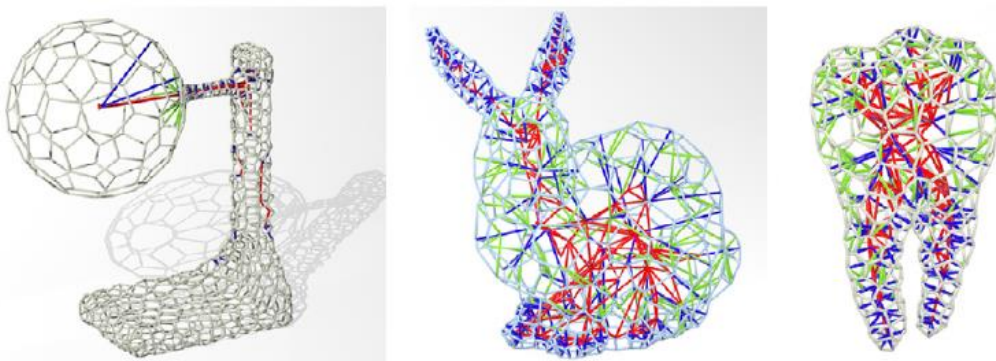
Figure 69: Structures obtained with medial axis method [76].

The aim of this procedure is to reach the desired printable shape able to support pre-determined or estimated loads. Following the steps of this method and, thus, saving material and improving strength does not affect 3D printability in general, but it could be challenging with FDM technology because of the limited quantity of material composing any layer. Using dissolvable supporting material could face the problem.

6.2.3 Hollowing and rotation method

Christiansen et al. [79] present an automatic balancing of 3D models focused on low cost FDM printers. They provide two steps to optimize mesh geometry and internal structure. First, the model optimization is made creating cavities filled of lighter or heavier material in a process called hollowing. Using triangular meshed model, it is possible to discretize it into tetrahedral elements obtaining a simplicial complex (Tetgen algorithm). Thus, using DSC method (Deformable Simplicial Complex), which is a Lagrangian method, cavities are created in the tetrahedral structure without modifying the external shape. After that,

83

for complex models, they introduce another type of optimization that recommends the rotation of model during the 3D printing process. The rotation gives stability to the object that, after the hollowing process, may result unbalanced. This could cause a little deformation in the contact zone with the printing board, but the phenomenon is neglectable, because most of the cases a rotation of few degrees from the original position is enough.

This contribution shows a simple but effective methodology to obtain a particular infilled model. By the way, it has a limitation with FDM technology; actually, internal hollows are not always self-sustained and in these cases, they need a support that changes the internal structure and is not removable.

6.2.4 Voronoi tessellation method

Lu et al. [80] demonstrated how a 100% filled object is not the best solution in any condition and they proposed to create an ad-hoc structure that support loads applied to the object (Figure 70). The aim is to obtain a model that supports loads while minimizing weight and cost.
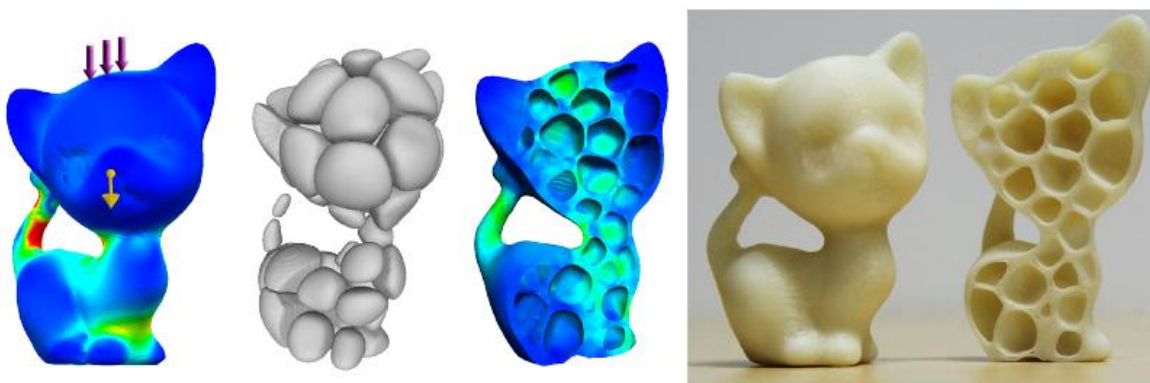
Figure 70: Process flow from object with applied loads to 3D printed object with porous structure [80].

They reached it in some steps. At first, the triangular mesh is converted in a tetrahedral system through Tetgen library. This allows analyzing the model with FE software to evaluate stress distribution. The stress map admits to place some sites in the object volume according to stress values; thus, they applied a Centroidal Voronoi Tessellation (CVT) to subdivide the volume into cells. In order to hollow each Voronoi cell, they compute a harmonic distance field in the cell that allows generating porous surfaces in the best way. The last step is the strength-to-weight optimization that is a Monte Carlo stochastic sampling of the site distribution density and the hollowing value in each cell. This process refines and extracts optimal value for lighter volume and maximum stress. To validate their method the research group executed a test campaign based on compression tests and strains evaluation. The optimal models extracted with this process are generally supporting higher stress than models infilled in standard way.

Some other challenges remain to be solved, in particular those caused by the approximation of the stress acting on the model. For example, secondary order effects due to outside temperature or material fatigue

are not investigated. Furthermore, the Monte Carlo sampling does not guarantee a global minimum for the optimization. Surely, despite it, they developed a method that includes all aspects needed for infill customization according to loads.

6.2.5 Elastic textures method

Panetta et al. [81] give an important contribution to the infill optimization research. Actually, they propose an ad-hoc internal structure according to the object usage. Even if they focus on Stereo Lithography and not on Fused Deposition Modeling printers, this work is noticeable for particular internal structure modelling process. Thus, the process should be handled with different 3D printing technologies. The model is subdivided in a number of cells and for each of them Young modulus and the Poisson coefficient are calculated to identify the correct and desired elasticity. In this way, all pattern cells have a different structure that must be connected to the others to obtain a single object with defined deformation features. Researchers found and solved all problems related to these structures and they presented objects capable of anisotropic deformations.

The microstructures developed are surely an interesting subdivision of the space and offer a particular anisotropic behavior, but they discretize the space in a pattern reducing the continuous properties. This method creates thin structures or layers with singular points, which are not easy to be printed with a FDM machine, but some precautions or proper supporting structures may be designed to make this possible.

6.2.6 Biomimetic method

The biomedical world inspired a lot of industrial and mechanical applications. An example is the work presented by Murphy [82] for generating a foam structure to be used as an infill pattern for 3D printing applications. He started the project with a detailed research on the internal structure of human bones, from which he was inspired. Actually, 3D object can be infilled with a porous pattern like bones. The method to construct this geometry starts with the analysis from a file in OBJ format that contain all vertexes coordinates and all faces features. Protosphere algorithm solves the problem of creating porous pattern by subdividing the geometry in the desired foam structure. It randomly inserts the desired number of spheres in the volume until the algorithm converges. The major advantage is that it is executable with parallel computing mode that consists multiplying the calculation power of a single laptop/desktop computer using NVIDIA GPU with CUDA framework. In this case, the object is divided in parts and they are parallel computed reducing the calculation time. The output of this process is a model ready to be 3D printed, as shown in Figure 71. The author also made some tests validating the possibility to print the porous structure without internal supports.

Figure 71: Example of 3D printed rabbit with porous internal structure [82].

The disadvantage of this process is that there is no attention to load distribution, but it is only an alternative 3D infill method. Future works could relate this geometrical method to a stress map.

6.2.7 Multi-material approach

Plastic materials, biomaterials, ceramic, clay are just a few that can be printed nowadays. FDM process is used especially with plastic filaments of different composition like PLA, ABS or HIPS. These are the main materials used in 3D plastic printing, but there are some special filaments for particular usages, like flexible filaments, which are made with a rubber material permitting great deformation, or water dissolvable filaments useful to realize supports easy to be removed. This variety of printing filaments allows the integration of multiple materials in the same object. In this way, it is possible to realize object with different composition areas. Taking advantage of flexible material and multi-material print possibility, new 3D printers permit to create object with soft and hard parts. Another important parameter is the infill density and shape. The internal structure conferred to the printed component contributes substantially to confer it rigidity and deformability. The internal grid can follow a twisted rectangular shape, honeycomb or round path. The relationship between internal grid shape and the features of the part is very tight: the internal structure is manually defined in relation to the stiffness required by the component. The transformation from virtual prototype to the printing geometry is the base to obtain multi-material object. The different characteristics of the volumes, obtained by varying the density through the infill and the different materials, are the most important element that gives to the object the local mechanical characteristics necessary to find a good compromise between comfort and structural strength.

In the next sections, we take into account each features about multi-material approach in order to create the software module for managing 3D printing with SMA[2].

6.3 Multi-material 3D printing for lower limb prosthesis [15]

After trial tests, we decided to reach a compromise solution between two approaches: multi-material method and infill customization. Actually, integrating the two solutions, the internal structure of the model can be optimized according to applied loads. Multi-material methodology offers the possibility to use materials of different density. This aspect allows integrating in a single object some areas of different chemical composition that implicated different behavior with the same infill. In addition to it, optimization algorithms will improve the internal structure creating a 3D pattern. An important aspect to consider is the printability with a specific technology: for instance, FDM requires a good overlapping of consecutive layers and does not work properly with models having only a few points or much reduced areas on a layer.

The application in the specific case of lower limb socket is meant to exploit FE analysis or real measurement to gather loads amount and distribution on the internal surface. After that, the thickness of the shell, its material mix and the infill ratio of the wall are designed. Beyond the basic requirement of integrity under the highest dynamic load that may occur during a walk, the socket is designed to be comfortable. This brought us to consider the overall deformation of the socket-limb system. Given the contact pressure map, the deformation in each area must be considered separately. Actually, in the off-load zones a high deformation of the socket must be guaranteed not to create pain, while in the load zones the weight of the body have to be transferred to the artificial leg with minimal displacement. Hence, we need the socket to assume both soft and hard behavior in different zones. At present, a software module has been developed inside SMA[2], which is able to export two different STL files, one for the hard zones of the socket and the other one for soft zones. Both files are loaded in 3D printer software to permit multi-material printing. In this way, the socket can be printed by exploiting a multi-material 3D printer (Figure 72).

| a) | b) | c) |
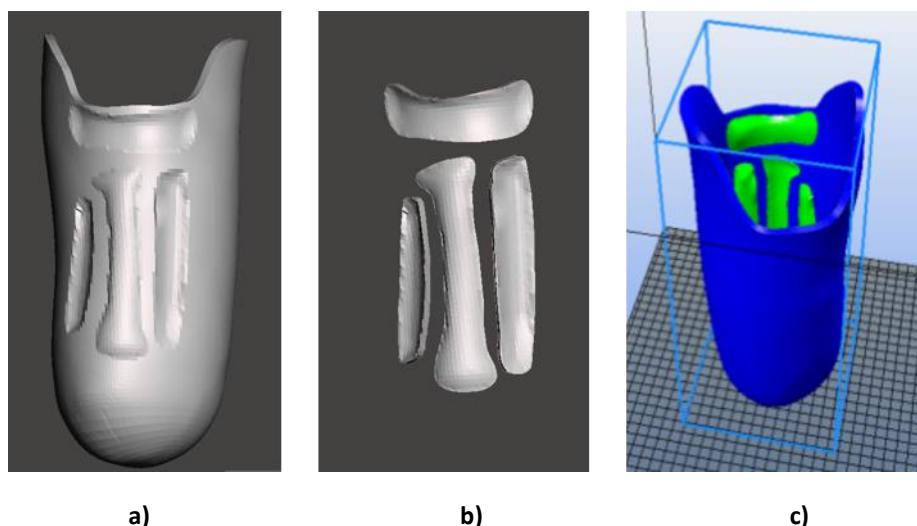
Figure 72: (a) The shell structure to be printed with hard material and (b) the critical zones with soft material.

Figure 73 shows an example of two-material socket for a below knee prosthesis printed with FDM technology in which we used a red soft material and standard grey PLA. Figure 73 shown an example of socket for an above knee prosthesis printed with FDM technology realized with one material.

Working with shell geometry is a challenging condition to experiment with infill shape and ratio optimization because some other parameters, such as wall thickness, shell thickness or extrusion diameter, highly impact on the final result.
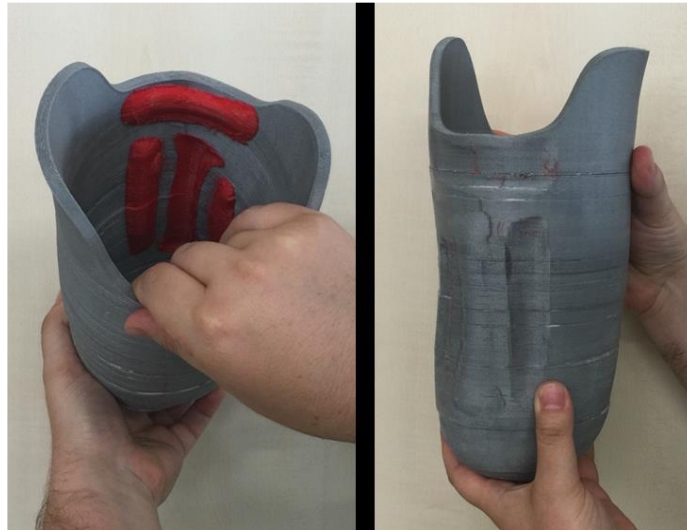
Figure 73: Two-material 3D printed socket for a below knee prosthesis.

6.4 Results and Future Software development

The experimentation is providing satisfactory results in terms of patient comfort with both two material and single material sockets in which the infill density has been varied according to the design requirements. Figure 74 shows the socket printed for an above knee prosthesis.
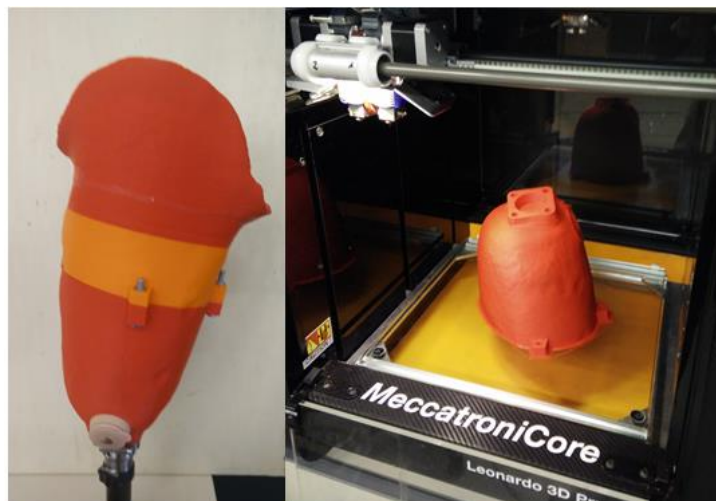
Figure 74: 3D printed socket and assembly of an above knee prosthesis.

The real challenge of additive manufacturing is to be able to properly design goods to exploit the new ways of realizing them [83]. Flexibility and quick production is only a part of the potential benefits new production solutions are introducing. The most difficult but promising aspect relies on the chance to create non-homogeneous and multi-material objects. The ratio and the way an object is filled out may depend on

known stress applied, on how it is supposed to be used in its lifecycle, or on the way it must behave to any environmental condition. Different algorithms exist to discretize and create automatic infill to address some of this condition. By the way, the complexity and interdependency of geometric and material parameters do not allow a simple approach to predict the outcome of design.

The case of printing lower limb prostheses components, which have a shell type organic shape shows how the use of soft material together with a proper infill can be successful, but further study are necessary to create a reliable method and proper software tools. In a near future, we are going to assist to a change in designing and these issues will become part of an ordinary step in product development.

Our future work will be oriented around the software development of a set of modules that will give the possibility to select the most appropriate infill and multi material approach depending on the component to be printed. After first functional analysis of the new software, we have decided to take advantage of two most important basic software development kits, which are CUDA framework and Visualization Tool Kit (VTK). CUDA allows us to exploit the high performance computing of NVidia Graphic Cards to apply algorithm of topology reconstruction as mentioned in [8]. CUDA permits to execute a set of computational operation in parallel way and usually, the topology reconstruction algorithms on a set of computational operation that can be parallelized in easy way. Visualization Tool Kit-VTK has been already used for 3D rendering inside SMA[2].

Chapter 7: Case study

In this chapter the case study adopted to test virtual platform is described. First, the procedure of the trial test is described with particular attention to the patient's involvement in order to obtain the best result at each step. In fact, each activity of the trial test has been thought to make the experience simple from the patient' point of view. The modeling procedure to define the patient's socket is described as well as socket 3D printing. Furthermore, additional trial tests have been performed to acquire patient's gait and contact pressure. Finally, the results reached so far are discussed according to the patient's feedback.

7.1 Patient's Data acquisition

The patient is a male with an above knee amputation and is 53 years old. The first step is to fill the form relative to anthropometric data and his life style. The form is also used to compile the digital form of SMA^2, which permits to design the socket starting from the 3D model of patient's residual limb. Several sections compose it; each of them contains information useful for a particular step of the design procedure. As shown in Figure 75, the modules on the lower part of the form will be used by SMA^2 for socket design as well as the final assembly of the whole lower limb prosthesis. In particular, the circumferences along the tight of the residual limb and the anthropometric data are exploited into SMA^2 to permit the automatic and semi-automatic operations for each virtual tool. The other data are used during acquisition of the patient's gait.

PATIENT DATA FORM

NAME:
SORNAME:
AGE:
SEX: ☐ Male   ☐ Female
HEIGHT: cm
PROSTHESIS: ☐ TF
             ☐ TT
WEIGHT (WITHOUT PROSTHESIS):   Kg
WEIGHT (WITH PROSTHESIS):   Kg

BODY STRUCTURE

1:minimal | 2:low | 3:middle | 4:tall | 5:maximum

INDEX

BODY
MASS

WIDTH OF
PELVIS

MASS OF
ABDOMEN

TRANSFEMORALI   TRANSTIBIALI

Arm length:_____ cm
(T-Pose):

Shoulder width:
_____ cm

COSCIA
PROSSIMALE
COSCIA
MEDIANA
COSCIA DISTALE
POLPACCIO

H Bacino
H Min

H Min
H Ginocchio
H Bacino

NOTES:

CIRCUMFERENCES
C PROXIMAL THIGH: cm
C MEDIAN THIGH: cm
C MEDIAL THIGH: cm
C CALF: cm

HEIGHTS
H pelvis: cm
H minimum: cm
H knee: cm

Figure 75: Patient data form used during tests of digital platform.

7.2 3D Acquisition of the residual limb

The acquisition and reconstruction of the residual limb from MRI images was already available [11]; therefore, the 3D scanning acquisition using Microsoft Kinect v1 and the Skanect application has been tested as shown in section 4.3. The residual limb has been acquired by keeping the muscle of the residual limb in relaxed condition. This is very important because the shape of the residual limb can suffer from a big deformation that makes impossible the socket design with the 3D virtual model. The 3D scanning of the residual limb has been done twice, the first one without stand under the residual limb (Figure 76) and the second acquisition using the stand (Figure 77).

Figure 76: 3D acquisition using Microsoft Kinect and Skanect without support under the residual lower limb.

Figure 77: 3D acquisition using Microsoft Kinect v1 and Skanect and with a stand for residual limb.

The crucial difference is the motion of the lower part of the residual limb. In fact, the procedure without stand allows 3D scanning of the complete model but small deformations occurred because there are small motions of the patients to maintain body balance. During the acquisition with stand, the 3D model has been acquired without motion, but the presence of stand did not permit a good acquisition of the lower part of the residual limb. The 3D model of the residual limb has been created inside Skanect and the STL file is exported in order to starting the socket design with SMA^2 (Figure 78).

7.3 Socket modelling

Starting from the acquired patient's data, the virtual tools of SMA^2 are used for socket design as presented in section 3.2. The triangular mesh of the residual limb is loaded inside SMA^2 and the scaling tool is used starting from the measurements of the circumferences along the tight obtained by the form previously introduced. Then, the marker tool allows the identification of load and off-load zones and thus, the deformation tool has been used to deform marked zones (Figure 78).
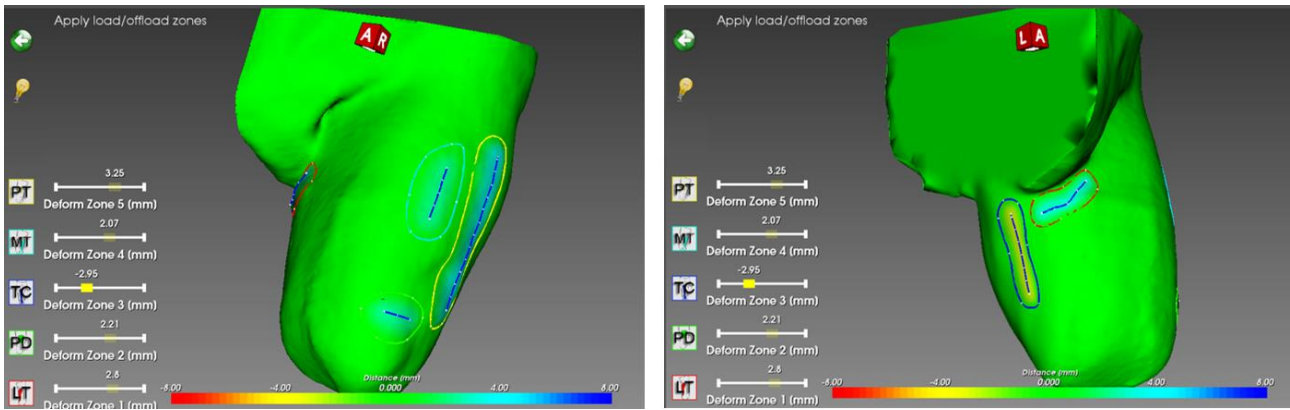
Figure 78: Deformed critical zones for the residual limb of the patient.

The lower part of the socket is generated (Figure 79.a) and mesh modelling is done to remove some useless details of the initial 3D triangular mesh through the use of the sculpt tool (Figure 28). The trim-line of upper part of the socket is automatically created and the thickness applied (Figure 79.b). Finally the, the plug is merged with the 3D socket model, the hole for the valve added and the final STL file of the 3D socket generated. The STL is used in the next step in order to permit a single material 3D printing (Figure 79.c).

a)   b)   c)
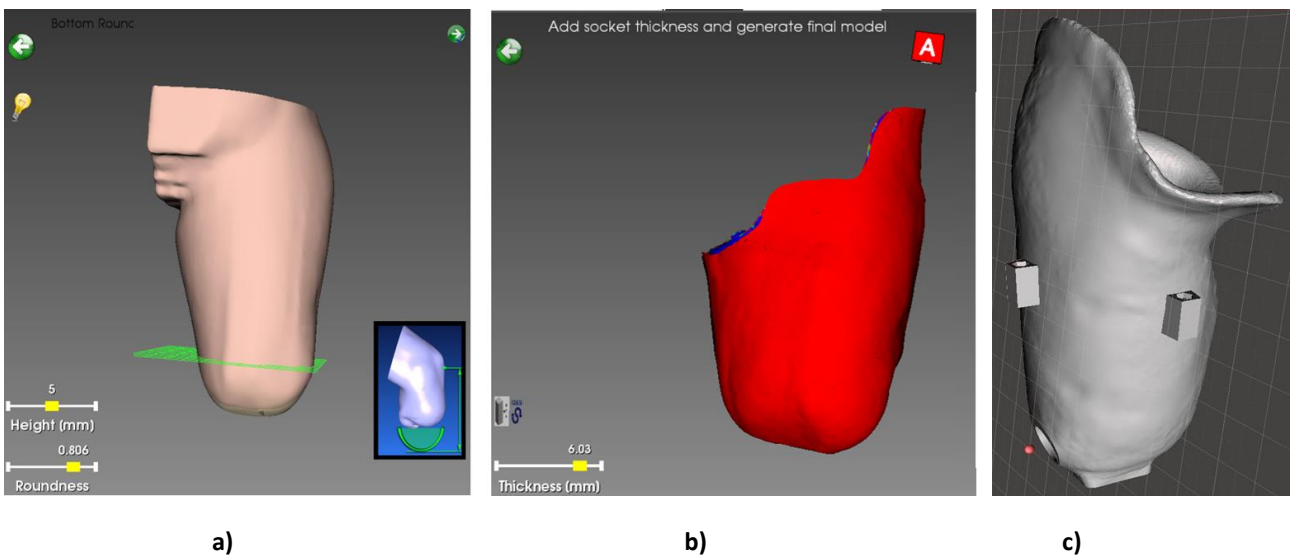
Figure 79: Lower zone definition (a) and socket with applied both trim-line and thickness (b). Application of socket plug and create of the hole for the valve in the lower zone of the socket (c).

7.4 Socket 3D printing

The socket has been realized with a FDM printer (i.e., Meccatronicore Leonardo) and, then assembled with the valve (Figure 74) and connectors with the whole prosthesis. The patient wore it and patient's feedback was good enough to complete the donning of the socket (Figure 80).

Figure 80: Donning of the 3D printed socket.

Once assembled the complete prosthesis, the patient worn it and walked along a straight line. He reported that there was a tight-fitting zone in the ischial area and this caused pain after few steps. Even if the designed socket was not good for a normal gait and modifications are required, the patient commented that the 3D printed socket can potentially substitute the thermoformed complex traditionally realized at the orthopaedic lab.

7.5 Gait analysis and pressure acquisition

In addition to the previous test, other trials have been carried out to acquire the patient's gait and pressure at the socket-residual limb interface, even if they are only partially related to SMA^2 implementation but mainly refer to the whole Prosthesis Design Platform (Figure 14). Acquired data will be used both for the gait analysis and validate results obtained when running the FE analysis within SMA^2.

7.5.1 Gait analysis

Motion acquisition and gait analysis have been performed with the commercial application iPiSoft. IPiSoft allows markerless acquisitions through the use of a multiple Microsoft Kinect v1 and v2. The layout of the system is showed in Figure 81. It is based on the use of two Kinect v2, which are plugged to two different computers connected by an Ethernet connection. Patient was asked to walk along a straight line of 5 meters twice after the start of the acquisition with iPiSoft, which is able to recognize the several human body segments (e.g., head, leg, feet and backbone) and reconstruct the virtual skeleton with associated animation of the gait.

The system has been used both to study how much the lower limb prosthesis affects the correct gait of the patient and to acquire pressure distribution using the F-Socket Teskan.
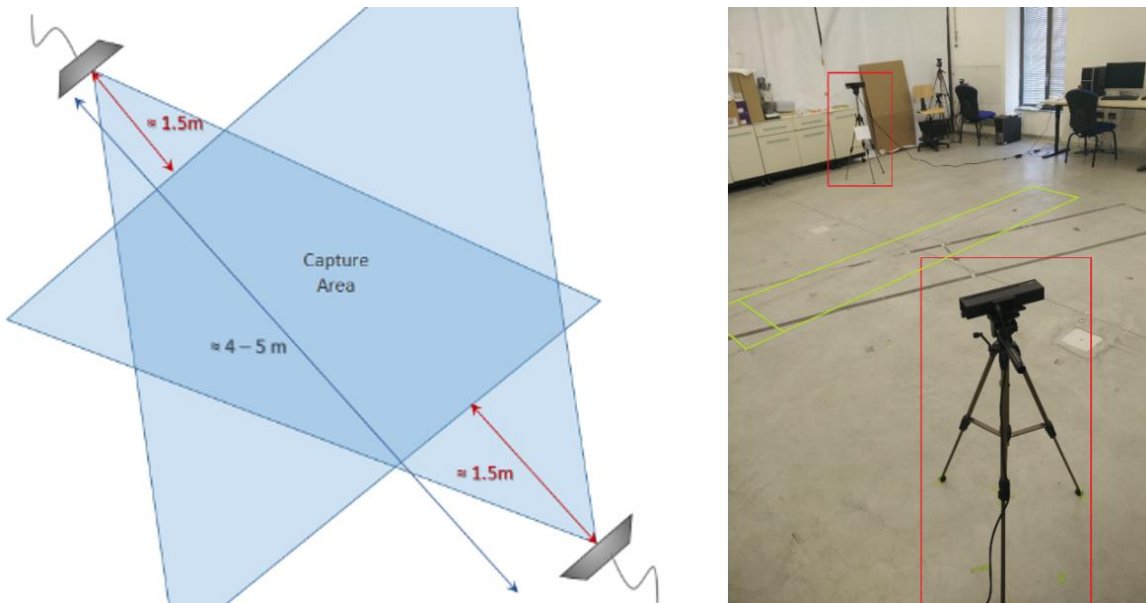
Figure 81. Layout of the MOCAP system based on the use of iPiSoft and multiple Microsoft Kinect v2.

The acquisition has been done with the patients wearing two different prostheses: the first one with the mechanical knee Octobock 3R45, the second one with the electronic knee RHEO Knee. The comparison is useful to understand how the electronical knee improves the gait respect to the traditional mechanical knee. Three acquisitions have been executed for each type of prosthesis. The patient during all tests walked in the designed area with a normal gait, step frequency and speed. For each recording, 5-6 steps on average were acquired (Figure 82); the Matlab tool is based on the analysis of two steps.

Figure 82: Gait acquisition with iPiSoft system.

Table 3 reports the gait abnormalities highlighted during the test and according to prosthesis type.

Analyzing data acquired, we can say that the most relevant gait abnormality, according to the Atlas of Limb Prosthetics [84], is the "lateral trunk bending" that in some cases is over the threshold level of severity. The parameter referred to this movement is the rotation around the frontal axis of the lower spine link. The amputee leans toward the amputated side of more than ten degrees when the prosthesis is in the stance phase. Apparently, this is not a high rotation value, but since the center of rotation is positioned in the lower back, the lateral shift of the upper trunk can be up to 10 cm, and this imply a critical bending.

95

We noted also two other repeated abnormalities that are "no plantarflexion" and "insufficient toe raising". The first one is connected to the rigidity of the prosthetic foot that do not reproduce properly the ankle articulation. The second one is referred to a non-homogenous distance between ground level and foot toe on the two sides of the body. Both these defects have low level of severity.

Abnormalities	3R45		RHEO KNEE	
	Right	Left (pros)	Right	Left (pros)
Lateral trunk bending	High		Medium	
Wide walking base	-	-	-	-
Circumduction	-	-	Low	-
Medial/lateral torsion	-	-	-	-
Foot rotation at the contact	-	-	-	-
Jumped walk	-	-	-	-
Asymmetrical gait	-	-	-	-
Insuf/exces toe raising	Low insuf		Low insuf	
No plantarflexion	-	Low	-	Low
Speedy plantiflexion	-	-	-	-
Lordosis	-	-	-	-

Table 3: Gait abnormalities resulted on the same patient with different knees.

There is also an acquisition in which we can identify a minimum "vaulting" problem that means the patient executed a jumped gait varying the distance between ground and the top head point.

We expected that the electronic active knee could improve some gait issues compared to hydraulic passive one. Surely, the outcomes confirm that there is a benefit, but some abnormalities still affect the gait. In particular, the major improvement is the reduction of lateral trunk bending that is the most serious defect found causing muscle-skeletal disease [3]. The absence of plantarflexion is due to prosthetic foot that is the same in both artificial legs.

7.5.2 Pressure acquisition

The Tekscan F-Socket™ System has been used for pressure acquisition. F-Socket Pressure System, portrayed in Figure 83, consists of scanning electronics, software and patented thin-film sensors. The sensors are thin

paper with high-resolution sensor placed within the socket that can be trimmed into freely floating fingers to closely approximate the curvature of the socket interface.
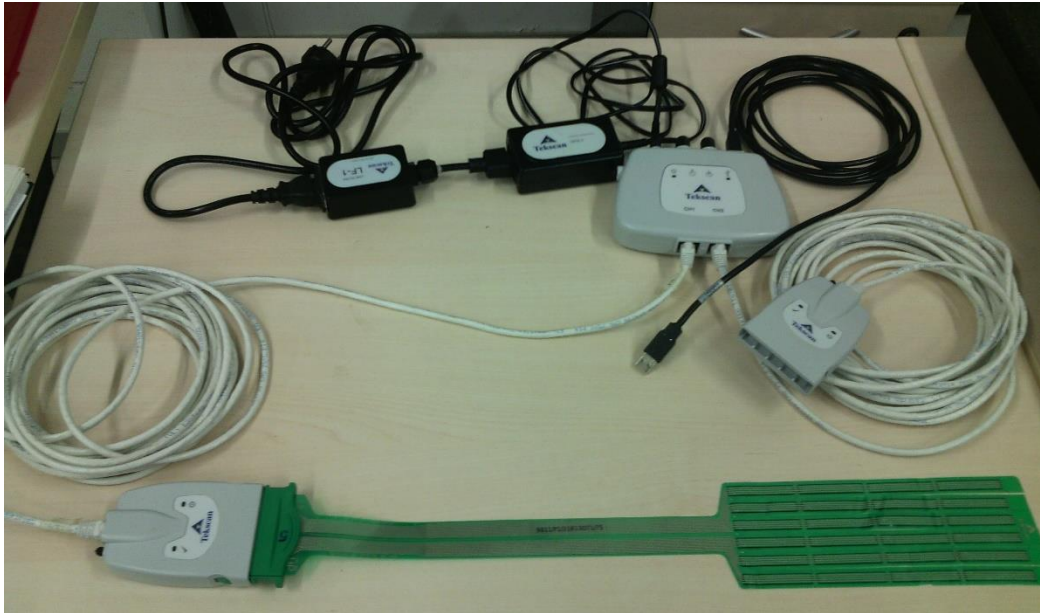
Figure 83: Tekscan systems exploited for pressure acquisition.

The F-Socket system is available with the following hardware choices: Tethered, Wireless, and Datalogger. Tethered device has been used in these research activities in order to connect the sensors and scanning electronics on the subject to the computer via USB port.

F-Socket Software characteristics are: 2-D and 3-D display, both real-time and recorded data; contact area, average and peak pressures; Center of Pressure and its trajectory; frame by frame data view.

A full line of equilibration devices are sold as optional by Tekscan. These devices apply a uniform pressure load across the sensor surface that is placed on a flat plain. This process electronically compensates for any variation or uneven output across individual sensing elements (sensels) caused by manufacturing or repeated use of the sensor. Equilibration devices are useful to perform quality assurance checks on the sensor and confirm uniform output by the sensor.

Tekscan F-Socket System has been chosen according to the technical specifications of the sensors and the need to have a system ready to use. In fact, the system permits to determine contact area and dynamic stress due to the interaction of the residuum with the socket both during single stance and in various phases of gait. Moreover, the experience of the company on the pressure acquisition in the prosthetic socket field and the extended literature that analyses the technical performances and adopts this sensors system were crucial in this choice.

For the test, four sensors stripes have been exploited to cover the whole shape of the residual limb (Figure 84). An ID has been associated for each sensor and they are positioned in the following way:

- Sensor ID0: anterior-lateral position

- Sensor ID2: anterior-medial position

- Sensor ID4: lateral position

- Sensor ID5: posterior position

After the positioning of sensors, the test can begin. There are two types of load test, 'static' versus a 'dynamic load test'.

Figure 84: Tekscan sensors applied on the residual lower limb.

During static load test, the patient was upright and the sound lower limb was on a scale in order to permit the reading of the patient's weight every time, while the residual limb with sensor and a transparent trial socket were placed on an ad-hoc support developed to prevent falls of the patient during load tests.

Since the system allows using just two sensor per time, three acquisitions have been executed for each load phase by maintaining sensor 0 as reference.

The patient's load was progressively shifted from the sound leg to the prosthetic leg during the load acquisition. The quantity of load was monitored through the scale and two different percentages have been acquired for each combination of sensors, such as from 0% to 50% and from 0% to 100% of the load of the patient (Figure 85).

Figure 85: Static test of pressure acquisition.

The dynamic load test was based on the use of the lower limb prosthesis with mechanical knee. Patient executed the same gait of MOCAP test but, in this case, during the gait analysis also the pressure data had been acquired by Tekscan system. The sensors have been connected to the system with the same combination previously described.

The innovative contribution of this trial test relies on the combined acquisition of the gait of the patient while s/he wearing an instrumented socket. In fact, we succeeded in acquiring at the same time pressure values in crucial zones of the residual limb and joints dynamics during a straight walk. Thus, exported data are both positions and angles of each joint, and pressure data on the socket in the same time frame. In order to synchronize, visualize and evaluate data coming from these different sources, a proper 3D virtual environment has been developed. The main benefit obtained is that it creates a visual representation which is simple and effective to be interpreted also by non-skilled personnel with ICT tools. The application permits to map the sensors along the 3D virtual model of the residual limb (Figure 87) and then, the acquired pressure are automatically visualized according to the position of each sensor (Figure 88). The mapped pressure data could be compared with those ones from FE analysis and validate the defined FE model [9].

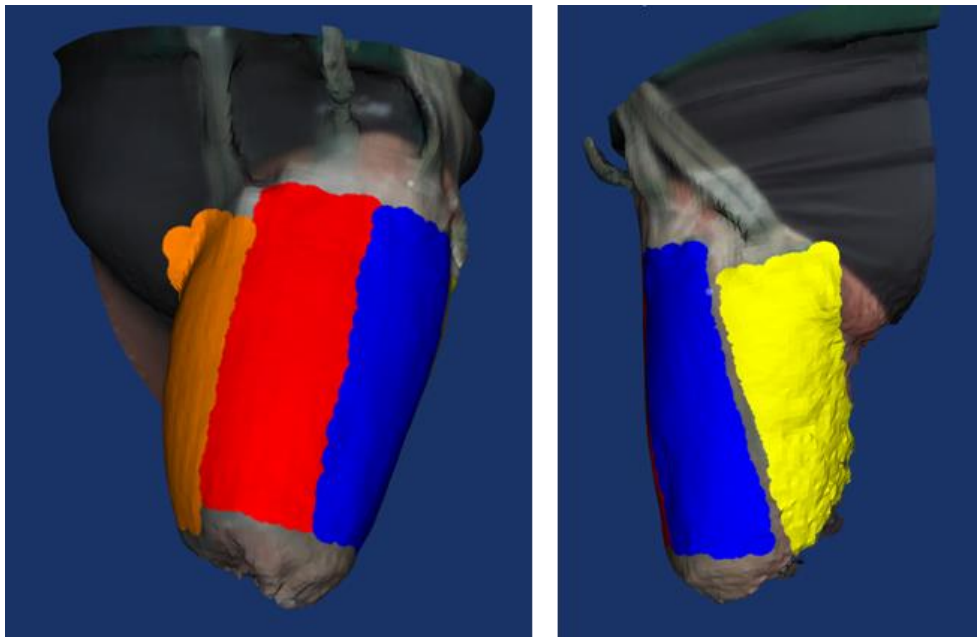**Figure 86: Pressure acquisition during MOCAP for gait analysis.**

Figure 87: Pressure mapping on the residual lower limb. Each colored surface represents the position of a sensor.
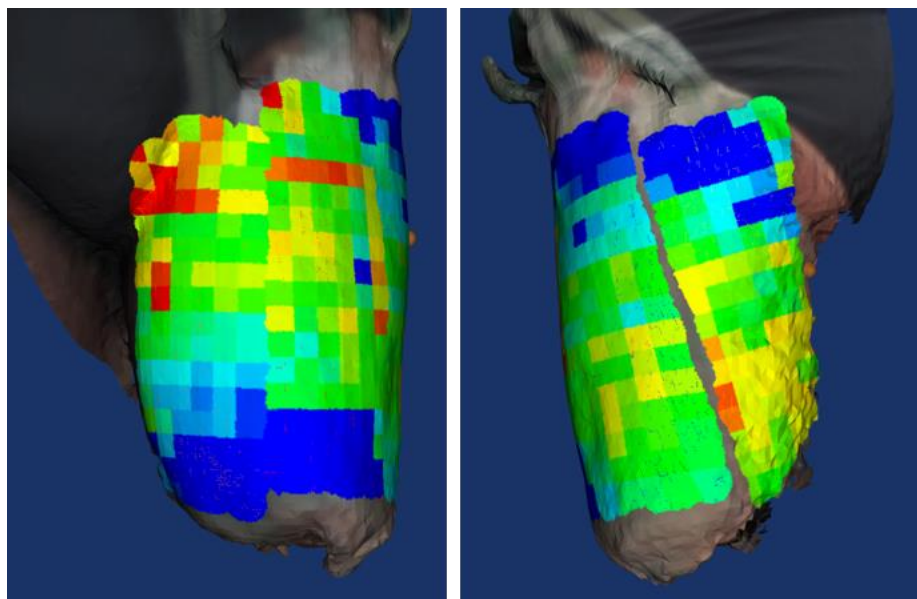

Figure 88: Pressure data applied on the 3D model of the residual lower limb.

7.6 Discussions and future developments

SMA[2] has permitted to successfully generate the 3D socket model and realize the physical prototype with the 3D printer. At this stage, we didn't carry out the FE analysis, since the main goal was to validate the modelling procedures and tools of SMA[2] and potential of 3D printing. A more comfortable socket could be easily obtained modifying the designed socket according to the patient's comments and exploiting the FEA module embedded within SMA[2]. The modelling procedure has been considered adequate but new modelling tools can be added to make easier the socket shape modifications. An example could be a module that automatically identifies the critical areas of the residuum, adequately modifies the socket model to reach the optimal shape. Once the critical areas have been identified, the system can apply proper geometry modifications, in order to create the load and off-load zones for a good pressure distribution over the residual limb. To this end, preliminary tests have been carried out adopting a supervised learning approach. Another example could be a virtual tool that replicates the brushing operation performed by the technicians during positive cast shaping.

The patient has positively evaluated the 3D printed socket and he really envisaged the possibility to substitute the traditional thermoformed socket with the 3D printed one. As mentioned in chapter 6, we have developed a solution based on multi-material and infill customisation to print the socket. Up to now we have demonstrated the feasibility of the approach and SMA[2] permits to generate two STL files, one for the shell structure to be realised with hard materials and infill customization and another one for critical load zones with soft materials. The preliminary tests demonstrated that the use of soft material together with a proper infill can be successful, but further study are necessary to create a reliable method and proper software tools. Further developments will be to automate the process defining a strategy and a software module that automatically selects the material and the infill percentage according to structure of the object and behaviour it should fulfil.

Data derived from gait analysis and pressure acquisition can be combined to obtain information useful to validate the FE model defined for the automatic simulation of the socket-residuum interaction and to define the optimal socket shape. The coloured mapping of acquired pressure values over the residual limb model will permit to identify the critical areas that can cause pain or discomfort during gait and thus perform proper adjustments to the socket model. Future step will concerns the development of a module to automatically correlate acquired pressure values with those obtained through the FE analysis.

Chapter 8: VOLAB: Towards an Augmented Reality Environment

This chapter presents a Mixed Reality environment, named Virtual Orthopedic LABoratory (VOLAB), which represents the future evolution of SMA[2]. The proposed solution is always based on low cost devices and open source libraries. First, a brief overview of main low-cost devices we considered for the development of VOLAB is described. Then hardware and software solutions adopted for VOLAB implementation are described. Finally, preliminary tests are illustrated as well as results reached so far and future development.

8.1 Low cost devices for VOLAB

For our application we considered hand tracking devices and 3D vision systems.

As described in Chapter 4, there are devices that allow the tracking of hands/fingers and gestures recognition, such as Leap Motion device (Figure 89.a), Duo3D (Figure 89.b) and Intel Gesture Camera (Figure 89.c) as well as other solutions that are able to track both whole human body and 3D objects, such as Microsoft Kinect v1 and v2 (Figure 89.d and Figure 89.e).

(a)　　　　(b)　　　　(c)　　　　(d)
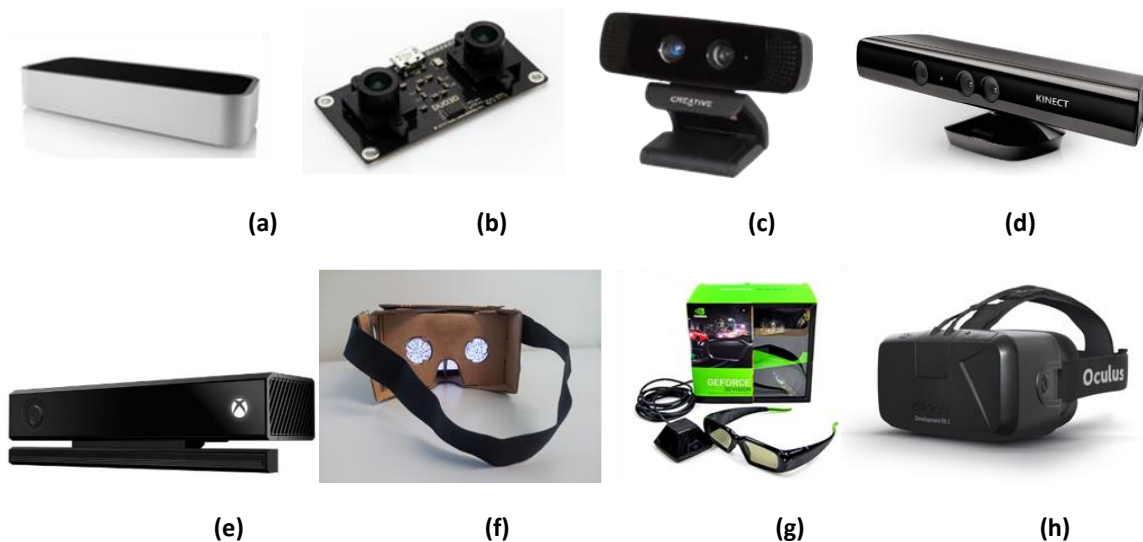
(e)　　　　(f)　　　　(g)　　　　(h)

Figure 89: (a) Leap Motion, (b) Duo3D, (c) Intel Gesture Camera, (d) Microsoft Kinect v1, (e) Microsoft Kinect v2, (f) Google Cardboard, (g) NVidia 3D vision –system, (h) Oculus Rift.

The Leap Motion, Duo3D and the Intel Gesture Camera have been already described in Chapter 5. Microsoft Kinect v1 permits to track objects and human beings in large spaces and have been used for various applications [85]–[87], such as gait analysis and ergonomics studies [88]. Microsoft Kinect v2 has been totally redesigned in order to exploits high definition sensor and improve data acquisition [89]. Windows Kinect SDK offers a large set of features, such as face detection, human body tracking and 3D reconstruction [90], [91]. The Windows SDK does not permit to manage multiple Kinects on the same computer and to overcome this limit it is necessary to use open source SDKs, such as libfreenect2 (https://github.com/OpenKinect/ libfreenect2).

Regarding 3D vision systems, they have been developed to recreate depth sense during the visualization of 3D environments. These systems can offer either partial or total immersive user experience. The first one is based on PC screen using LCD shutter glasses, such as NVIDIA 3D Vision System (Figure 89 .g). This solution has been used into several research works, such as emulation of amblyopia treatments and measurements of stereo acuity of sight [9]–[11].

Totally immersive 3D systems are based on the use of wearable head-mounted displays, such as Google Cardboard and Oculus Rift. The Google Cardboard is a simple box with two lenses that, used in combination with a smartphone (Figure 89.f), constitutes a simple and powerful virtual reality viewer. The smartphone must be inserted in the box and the user looks inside in order to see the images displayed by the phone. Oculus Rift is a new solution to create a real user experience into virtual reality (Figure 89.h). Indeed, older head mounted display for virtual reality had problem of latency on rendered images and thus, user could present some diseases, such as loss of balance and sickness. Oculus Rift technology solved this problem and the HD displays behind the two lenses offer a realistic experience into 3D environment in which the user can interact with it. This device has been used in research works related to different research field, such robotics and medicine [92], [93]. Oculus Rift makes also available a powerful SDK to develop new applications with several programming languages (e.g., C++ and C#).

8.2 Mixed Reality for prosthesis design

In order to create the virtual orthopedic lab, we need to interact with virtual models (i.e., the residuum and socket models) as the prosthetists typically do during the traditional process. The proposed solution is based on the following devices:

- Oculus Rift to render the scene (i.e., the orthopedic lab) with depth perception.
- Microsoft Kinects v2.0 to acquire high definition images rendered through the HD display of Oculus Rift. The multiple images are used to recreate real environment into a 3D scene. Both orthopedic technician and patient are tracked using Kinects. In fact, the real world has to be

visible to the user and the interested part (i.e., patient's residuum) has to be detected into the scene.

- Leap Motion device to track hand/finger and allow the prosthetist to interact by hands with the socket and residual limb models.

As mentioned, SMA offers a set of virtual tools to emulate the most important tasks executed by orthopedic technicians. Therefore, thanks to its modularity, the virtual modeling tools can be used to manipulate socket shape tracking hands/fingers through the Leap Motion device.

Regarding the last issue, we have defined a set of gestures according to different actions that are executed to design the socket with SMA, such as marking of critical zones. Table 2 (see Chapter 5) shows the set of gestures exploited inside VOLAB to interact with both 3D residual limb and socket shape. These gestures are heavily linked to the socket design and they have to be changed for other virtual environments according to the custom fit product designed. Accordingly a Natural User Interface (NUI) based on the use of Leap Motion device has been implemented to interact with the 3D models of the residual limb and the socket using hands/fingers inside a complete 3D virtual environment.

Figure 90 portrays the final state machine that describes the NUI specifically developed for VOLAB. The legend explain how VOLAB changes interaction mode according to the selected gesture.
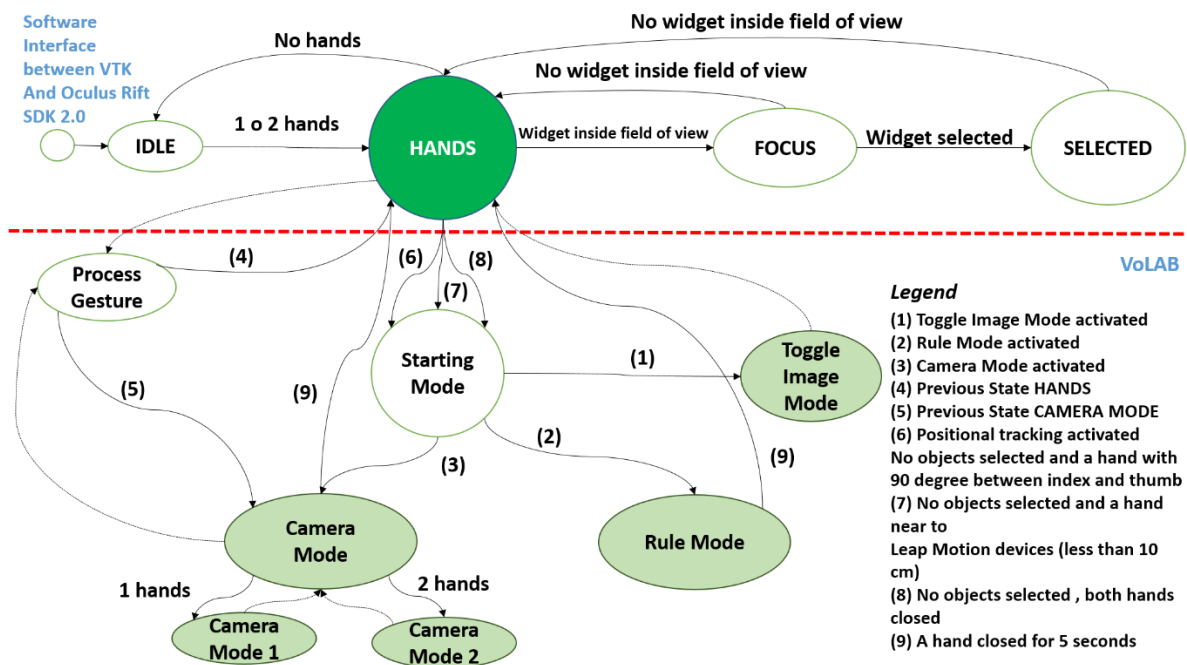

Figure 90: Final state machine for VOLAB NUI.

To manage the various interaction modes and the gesture, the NUI of VOLAB is constituted by a limited number of states, represented by a finite state machine (FSM) contained in the software interfaces developed inside VOLAB. The FSM is based on the use of developed software interfaces to synchronize

Oculus Rift and Leap Motion device with the 3D environment. Following the state diagrams showed in Figure 90, the top is the software interfaces to interact with Leap Motion device and Oculus Rift, the bottom part of the FSM describe the several modes developed in order to use VOLAB.

Hands state permits the link between the basic software interface of Leap Motion device and VOLAB. Hands state controls the following events if there are no selected virtual widgets (e.g. virtual sliders and 3D buttons):

1. Hand with 90 degree between index and thumb and positional tracker of Oculus Rift deactivated. This event activates the *Camera Mode.*

2. Hand with 90 degree between index and thumb and positional tracker of Oculus Rift activated. This event activates the *Object Mode.*

3. Hand near to Leap motion (i.e., less than 10 cm). This event activates the *Toggle Image Mode.*

4. Both hands closed. This event activates the *Rule Mode.*

The state *Starting Mode* permits to monitor activation gestures, update the indicators of the actual interaction mode (i.e., icons and images). Dashed arrows describe particular states in which the executed operation cause exiting from the mode without other gestures. The *Process Gesture* state monitors the execution of basic gestures (e.g., circle, swipe and pinch) and permits to execute the actions associated to a particular gesture in the activated mode.

8.3 VOLAB implementation

The prosthetic mixed reality, named Virtual Orthopedic LABoratory (VOLAB), has been based on the modular structure of SMA and new modules have been developed to guarantee the communication among the devices and performance. In the next sections, the hardware and software solutions are described.

8.3.1 Hardware architecture

Figure 91 shows the hardware architecture of VOLAB, which consists of:

- **Three Kinect v2** connected to a desktop computer. They are positioned to create an equilateral triangle with side length of 4.0 meters. The use of multiple Kinects v2 guarantees an adequate human body detection and tracking of residuum as well as a good 3D reconstruction by acquired infrareds images. The RGB camera of each Kinect is able to capture the scene video with high definition (1920 x 1080 pixels), which are visualized through the use of Oculus Rift. Furthermore, Kinect v2 has a frame rate of 30 fps, which is quite good for our final purpose.

- **Oculus Rift v2.0** connected to a personal computer both through the HDMI and USB port. HDMI port is used to send HD RGB images to HD displays and the connection to USB port permits to switch on the

device. Oculus Rift v2.0 is able to reach a refresh rate of 75 fps. Even if Oculus Rift permits 75 fps, we have to use just 30 fps as maximum refresh rate according to frame rate of Kinect v2.

- **Leap Motion device** placed on the front side of Oculus Rift connected to a USB port of the computer. Hands/fingers of the technicians are detected to interact with virtual interface that SMA makes available. The depth cameras of Leap Motion device are able to reach a refresh rate of 100 fps, but we exploited just 30 fps for hand tracking into VOLAB. In that way, we have the same refresh rate among exploited devices. This feature is very important due to get excellent synchronization between what the user see and what the user make during the socket design.

- **Personal Computer** that runs VOLAB and manages the synchronization of devices through the middleware. PC must make available at least three USB 3.0 ports for the three Kinect v2 as well as two USB 2.0 ports for Leap motion device and Oculus Rift v2.0. Furthermore, an NVidia graphic card has been mounted on PC in order to permits excellent rendering of VOLAB. According to these technical features, we are using an HP Pavilion 500-333nl personal computer, which is a normal computer desktop simply available on the market.
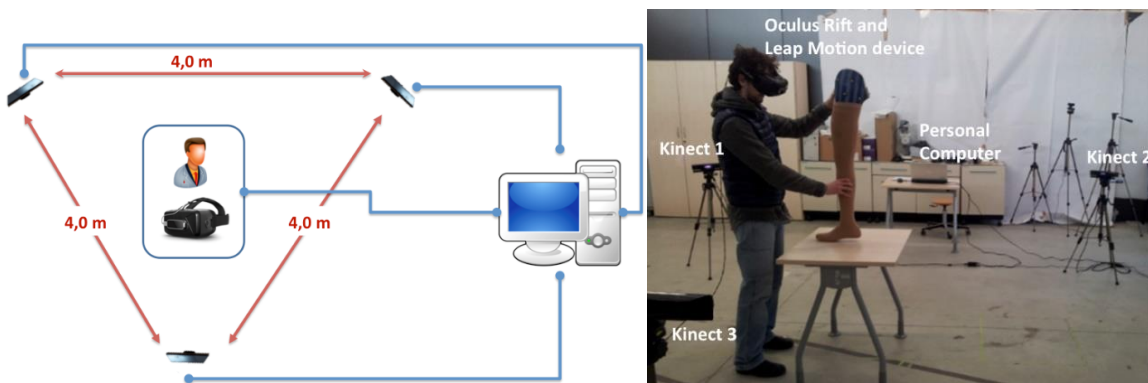
Figure 91: VOLAB Hardware architecture.

Optionally, a Microsoft Kinect v1 or v2 can be used to scan the 3D model of the residual limb.

Two main features are required:

- **Device synchronization**. In fact, the system needs synchronizing different signals emitted by devices (e.g., frames and orientation of objects) that have to be read by SMA to perform modeling operations and manipulate socket shape.

- **Devices Connections**. All devices are connected to a unique personal computer through different USB ports. In particular, the system requires at least three USB 3.0 ports for Microsoft Kinect v2, two USB 2.0 ports for Oculus Rift and another USB 2.0 port for Leap Motion device. If, we consider and additional Kinect v1 to scan the residual limb, another USB 2.0 is required.

The real world is rendered on the viewer of Oculus Rift by exploiting the HD-RGB cameras of the three Microsoft Kinect v2 that detect human body parts and manipulate images used for creating the mixed

reality environment. When the residuum is detected, the user can shape the socket model using virtual tools of SMA and interacting with hands/fingers or object held in the hand.

8.3.2 Software architecture

Figure 92 shows the software architecture of the mixed reality environment.
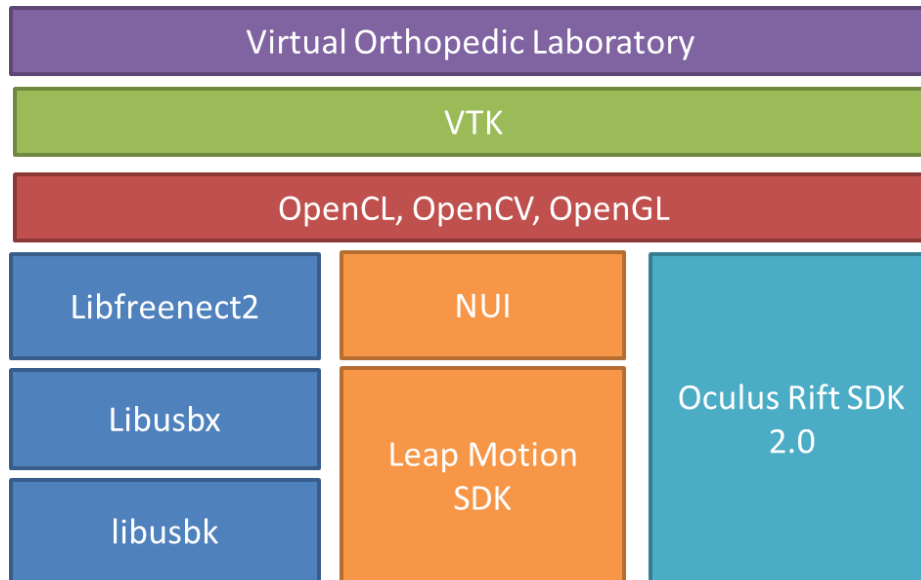
Figure 92: Software Architecture.

As previously mentioned, each device makes available a SDK and among the data made available by different SDKs, we considered:

- Depth maps and high definition images by Microsoft Kinect SDK.
- Position and orientation of hands/fingers by Leap Motion SDK.
- Position and orientation of head by Oculus Rift SDK 2.0.

The official Microsoft Kinect SDK v2 does not make available a driver to read multi Kinects by a single computer, so we adapted the open-source library named libfreenect2. Libfreenect2 is able to detect multiple Kinect v2 through libusbx as well as data acquisition of RGB camera, depth and infrared sensors. Libusbx uses libusbk, which is an USB driver substituting the original driver used for Kinect. In addition, we used VTK for SMA modeling tools, OpenCL for parallel computing, OpenGL for scene rendering, and OpenCV for image processing.

New modules have been developed for:

- *Real Time performance*. The system needs high computing performance to have no delays between human interactions and rendering of virtual scene. In order to obtain this performance, VOLAB has to offer at least a frame rate of 30 fps with a full HD resolution that is defined in pixel as 1920 x 1080. Both Oculus Rift and Microsoft Kinect v2 are able to achieve these performances.

- *Data synchronization*. The personal Computer executes VOLAB and manages synchronization of devices through the middleware. As shown in Figure 90, a software interface has been developed that permits to synchronize both Oculus Rift SDK v2 and Leap motion SDK inside VOLAB by following the schema of the designed NUI. The personal computer makes available an NVidia Graphic Card, which is CUDA and OpenCL compatible. The graphic card has to make available a HDMI port for correct video rendering of Oculus Rift.

Finally, SMA modeling tools have been adapted for the prosthetic mixed reality. They have been implemented using VTK and each is composed of a set of widgets (e.g., sliders and buttons) to execute a particular modification and reach the final socket shape. Virtual widgets are automatically visualized in the user's field of view of the Oculus Rift when residuum is detected and thus, s/he can start to shape the socket using hands/fingers detected by the Leap Motion device. Each virtual object is rendered through the use of Oculus Rift SDK.

A software interface has been developed between VTK and the SDK of Oculus Rift in order to exploit several rendering features of VTK. This interface has totally written in pure OpenGL because Oculus Rift renders the 3D scene using frame buffer objects (FBOs) and textures. At present, VTK doesn't make available a standard viewport that uses FBOs for 3D scene rendering. FBOs permits to directly route the rendering of a 3D environment on the two displays of Oculus Rift for stereo vision. 3D scene has rendered through two textures, which are used as two images to which an image-distortion is applied for creating the depth sense. Depth sense is possible because there are two lenses inside Oculus Rift, which magnify the screen so it fills field of view of the user.

The NUI (Natural User Interface) uses the basic SDK of Leap Motion and it is part of an ad-hoc developed module that extends the C++ class of VTK to interact with the virtual tools of SMA. In particular, we implemented an events handler made available by VTK. It uses a timer that starts when VOLAB application is initialized. The events handler generates a timer event each 5ms; if a gesture is detected when time event happened, the associated operations are executed.

The actual development state of VOLAB permits to start first trial tests, but some mentioned modules need to be completed with further software development. In particular, the data synchronization module has to improve the image processing during data acquisition using multiple Microsoft Kinects v2. At this aim, a new software module is under development in order to exploit the Point Cloud Library (PCL, http://www.pointclouds.org/), which makes available many features suitable for our aim. Furthermore, the interface between VTK and Oculus SDK has been quickly developed and thus, some parts of this module have to be completed and improved in order to totally take advantage of both SDK. The other parts of system have already ready to be safely used, even if other features will be added in order to obtain complete software architecture useful for new systems based on the use of mixed reality.

8.4 Preliminary Tests and discussions

At present, VOLAB is not fully integrated and we carried out two preliminary tests: one to verify if the proposed solution offers a virtual environment with real-time performance and another one to verify the augmented interaction with hands using SMA modeling tools.

In the first test, we considered two case studies related to the design of the socket for above knee and a below knee amputees where the patient stands in the center of the tracking area in upright position. This test aimed to recognize different parts of residuum by technicians, who marks different zone using his/her hands. With this choice of interaction, we define a procedure to validate the efficiency and adequacy of our solution in terms of real-time performances and residuum detection. The real-time performance was good but we identified a problem related to the quality of residuum detection. In fact, above knee residuum is detected in real time with good precision, but below knee residuum had some problems for the areas close to the knee. Therefore, further investigation is needed.

For the second test, we considered a trial socket and we applied the modeling tools on the detected socket model (Figure 93). In this test, three main basic operations were tested using hands: moving from one tool to another one, marking critical zones (Figure 93.) and trim-line definition. Both gestures and modeling tasks have been properly detected and executed. The system permits to interact with the virtual environment as done during traditional process manufacturing. At this stage, the results reached so far have been considered valuable; however we need further development and improvement to make the whole system usable by prosthetists.

Figure 93: Interaction test using a residual limb with below knee amputation..

In fact, even if VOLAB has been initially targeted for application in the prosthetic domain, it could be used for other applications, which require product designed around the human body. In fact, it has been planned to use the system to develop application for 3D clothing design and simulations.

Conclusions

The performance of a prosthesis for lower limb amputation heavily depends on the experience and skills of the prosthetist. Most of its component are standard, such as the knee or the foot, and can be selected from commercial catalogues. Instead, the socket, and sometimes also the liner, is the customized component and totally hand made. It is the most critical component and is the interface with the human body. The final comfort and function of the whole prosthesis mostly depends on its quality. Available commercial prosthetic tools can support some specific steps of the process, but they still rely on a traditional design paradigm and don't offer any kind of assistance or suggestions to the user, and the technician knowledge and experience are still required.

This thesis work presents SMA^2 an innovative prosthetic CAD system specifically conceived to design the socket. It has been designed and implemented following a low cost philosophy and open source libraries to provide a computer-aided environment affordable also by small orthopedic labs

Two procedures have been developed to reconstruct the 3D models of the residual limb around which the socket model is designed. The first one is based on a low cost scanner 3D, i.e. the Microsoft Kinect; the second one permits to automatically reconstruct the virtual model from MRI images.

Starting from the virtual model of the residual limb, SMA^2 makes available a set of virtual modeling tools that emulate the operations performed by the technicians during the traditional hand-made manufacturing process. A new interaction style has been implemented to allow the user to interact by hands using hand-tracking and haptic devices. A data driven multi-material 3D printing approach has been developed to realize the physical socket using the FDM technology.

Finally, a mixed reality environment, named VOLAB, has been implemented as evolution of SMA2. VOLAB is always based on low cost devices that permit to create a virtual environment where the technician can model virtual socket shape starting from the real model of the detected residual limb.

The whole digital platform has been tested with a transfemoral amputee. Once modeled with SMA^2, the socket has been realized using the FDM technology. The patient wore the physical prototype that has been considered comfortable. The technicians appreciated the whole system and the possibility to use the

framework to simplify the work of experienced technicians and train junior designers who can learn more quickly about lower limb prosthesis design.

Main results can be summarized as follows:

- A **module to automatically reconstruct the virtual model** of the residual limb **from MRI images**. It permits to get the 3D model with a good approximation for above knee amputation; while the reconstruction for below knee amputation requires improvements around the zone of the knee. A low cost scanner can be used for acquire the 3D model of the residual limb and the accuracy of the model is resulted well enough for socket design. It can be considered an alternative both to MRI and to 3D scanners used in commercial prosthetic CAD system that are costly.

- A **new prosthetic CAD system, SMA²**, based on a new design paradigm and augmented interaction that permits to replicate/emulate using hands manual operations usually performed by the prosthetist during the traditional development process. The system is now ready to be effectively used by technicians in an orthopedic laboratory.

- A **3D printing** solution based on infill customization and multi-material to realize the physical prototype of the socket. To this end, SMA^2 permits to export data necessary for multi-material printing.

- A **mixed reality environment, VOLAB**, which will be the future evolution of SMA^2 that will permit to recreate virtually the orthopedic laboratory.

Two additional results are:

- **SimplyNURBS,** a free software development kit useful to manage NURBS models developed upon open source libraries, such as AYAM, OpenCASCADE and NURBS++.

- **Tracking plug-in** to automatically develop software interface to manage the interaction with hand tracking devices, such as Leap Motion device, Intel Gesture Camera and Duo3D.

Even if, these last two applications have been initially targeted for application in the prosthetic domain, the developed solutions are general purpose and can be used in other applications related custom-fit products, such as clothing design.

References

[1]   G. Colombo, G. Facoetti, and C. Rizzi, "A digital patient for computer-aided prosthesis design," *Interface Focus*, vol. 3, no. 2, 2013.

[2]   Biosculptor. Available: http://www.biosculptor.com.

[3]   Vorum. Available: http://www.vorum.com.

[4]   Rodin4D. Available: http://www.rodin4d.com.

[5]   Inifinity CAD Systems. Available: http://infinitycadsystems.com/.

[6]   Orten.Available: http://www.orten.fr/.

[7]   Ossur. Available: http://www.ossur.com/.

[8]   G. Facoetti, S. Gabbiadini, G. Colombo, and C. Rizzi, "Knowledge-based system for guided modeling of sockets for lower limb prostheses," *Comput. Aided Des. Appl*, vol. 7, no. 5, pp. 723–737, 2010.

[9]   G. Colombo, G. Facoetti, R. Morotti, and C. Rizzi, "Physically based modelling and simulation to innovate socket design," *Comput. Aided. Des. Appl.*, vol. 8, no. 4, pp. 617–631, 2011.

[10]   G. Colombo, D. Regazzoni, and C. Rizzi, "Ergonomic design through virtual Humans," *Comput. Aided. Des. Appl.*, vol. 10, no. 5, pp. 745–755, 2013.

[11]   S. Gabbiadini, "Knowledge-based design of lower limb prosthesis," Università degli studi di Padova, 2011.

[12]   G. Colombo, G. Facoetti, C. Rizzi, and A. Vitali, "Socket Virtual Design Based on Low Cost Hand Tracking and Haptic Devices," in *Proceedings of the 12th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, 2013, pp. 63–70.

[13]   G. Colombo, G. Facoetti, C. Rizzi, A. Vitali, and A. Zanello, "Automatic 3D Reconstruction of Transfemoral Residual Limb from MRI Images," in *Digital Human Modeling and Applications in Health, Safety, Ergonomics, and Risk Management. Human Body Modeling and Ergonomics*, Springer, 2013, pp. 324–332.

[14]   G. Colombo, G. Facoetti, C. Rizzi, and A. Vitali, "Virtual, Augmented and Mixed Reality: 7th International Conference, VAMR 2015, Held as Part of HCI International 2015, Los Angeles, CA, USA, August 2-7, 2015, Proceedings," R. Shumaker and S. Lackey, Eds. Cham: Springer International Publishing, 2015, pp. 351–360.

[15]   C. Comotti, D. Regazzoni, C. Rizzi, and A. Vitali, "Multi-material Design and 3D Printing Method of Lower Limb Prosthetic Sockets," in *Proceedings of the 3rd 2015 Workshop on ICTs for Improving Patients Rehabilitation Research Techniques*, 2015, pp. 42–45.

[16] G. Colombo, C. Rizzi, G. Facoetti, and A. Vitali, "SimplyNURBS: A Software Library to simplify the use of NURBS Models and its Application for Medical Devices," in *Proceedings of CAD'14*, 2014, pp. 215–218.

[17] Comotti Claudio, Colombo Giorgio, Regazzoni Daniele, Rizzi Caterina, Andrea Vitali "Low Cost 3D Scanners Along the Design of Lower Limb Prosthesis," in *6th Int. Conf. on 3D Body Scanning Technologies*, 2015, pp. 147–154.

[18] R. Morotti, "DEVELOPMENT OF A VIRTUAL TESTING LABORATORY FOR LOWER LIMB PROSTHESIS," Università degli studi di Padova, 2013.

[19] VTK. Available: http://www.vtk.org/.

[20] Qt Project. Available: http://qt-project.org/.

[21] Leap Motion. Avaialable: *https//www. leapmotion. com/*.

[22] L. Stãnculescu, R. Chaine, and M.-P. Cani, "Freestyle: Sculpting meshes with self-adaptive topology," *Comput. Graph.*, vol. 35, no. 3, pp. 614–622, 2011.

[23] S. Eginier, SculptGL.. Available: http://stephaneginier.com/sculptgl/ .

[24] Skanect.. Available: http://skanect.occipital.com/.

[25] SMS NLib. Avaialable: http://www.smlib.com/ .

[26] L. a. Piegl, W. Tiller, and K. Rajab, "It is time to drop the 'R' from NURBS," *Eng. Comput.*, pp. 1–16, Apr. 2013.

[27] L. Piegl and W. Tiller, *The NURBS Book (2Nd Ed.)*. New York, NY, USA: Springer-Verlag New York, Inc., 1997.

[28] L. A. Piegl, "Knowledge-guided computation for robust CAD," *Comput. Aided. Des. Appl.*, vol. 2, no. 5, pp. 685–695, 2005.

[29] NURBS++. Available: http://libnurbs.sourceforge.net/old/.

[30] OpenCASCADE. Avaialable: http://www.opencascade.com/

[31] AYAM. Available: http://ayam.sourceforge.net/ayam.html.

[32] D. Brujic, I. Ainsworth, and M. Ristic, "Fast and accurate NURBS fitting for reverse engineering," *Int. J. Adv. Manuf. Technol.*, vol. 54, no. 5–8, pp. 691–700, 2011.

[33] T. Buzan, *How to mind map*. HarperCollins UK, 2002.

[34] J. T. Bushberg and J. M. Boone, *The essential physics of medical imaging*. Lippincott Williams & Wilkins, 2011.

[35] N. D'Apuzzo, "State of the art of the methods for static 3D scanning of partial or full human body," in *Proceedings of Conference on 3D Modeling, Paris, France*, 2006.

[36] C.-S. Park, S.-W. Kim, S.-W. Kim, and S.-R. Oh, "Comparison of plane extraction performance using laser scanner and Kinect," in *Ubiquitous Robots and Ambient Intelligence (URAI), 2011 8th International Conference on*, 2011, pp. 153–155.

[37] Y.-J. Chang, S.-F. Chen, and J.-D. Huang, "A Kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities," *Res. Dev. Disabil.*, vol. 32, no. 6, pp. 2566–2570, 2011.

[38] R. A. Clark, Y.-H. Pua, K. Fortin, C. Ritchie, K. E. Webster, L. Denehy, and A. L. Bryant, "Validity of the Microsoft Kinect for assessment of postural control," *Gait Posture*, vol. 36, no. 3, pp. 372–377, 2012.

[39] D. Regazzoni, C. Rizzi, C. Comotti, and F. Massa, "Towards Automatic Gait Assessment by Means of RGB-D Mocap," in *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2015, pp. V01AT02A057–V01AT02A057.

[40] HandyScan. Available: http://3d-scanners.ireviews.com/ .

[41] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient Graph-Based Image Segmentation," *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181.

[42] "SCENECT by Faro." Available: http://www.faro.com/it-it/scenect/scenect.

[43] "ARTEC STUDIO by Artec." .

[44] M. Bordegoni, F. Ferrise, M. Covarrubias, and M. Antolini, "Haptic and sound interface for shape rendering," *Presence Teleoperators Virtual Environ.*, vol. 19, no. 4, pp. 341–363, 2010.

[45] T. Narumi, T. Kajinami, T. Tanikawa, and M. Hirose, "Meta cookie," in *ACM SIGGRAPH 2010 Posters*, 2010, p. 143.

[46] R. Riener and M. Harders, *Virtual Reality in Medicine*. London, 2012.

[47] M. Hutchins, M. Adcock, D. Stevenson, C. Gunn, and A. Krumpholz, "The design of perceptual representations for practical networked multimodal virtual training environments," in *Proceedings of the 11th International Conference on Human--Computer Interaction*, 2005, pp. 22–27.

[48] G. Sela, J. Subag, A. Lindblad, D. Albocher, S. Schein, and G. Elber, "Real-time haptic incision simulation using FEM-based discontinuous free-form deformation," *Comput. Des.*, vol. 39, no. 8, pp. 685–693, 2007.

[49] B. Perez-Gutierrez, D. M. Martinez, and O. E. Rojas, "Endoscopic endonasal haptic surgery simulator prototype: A rigid endoscope model," in *Virtual Reality Conference (VR), 2010 IEEE*, 2010, pp. 297–298.

[50] S. Katsura, W. Iida, and K. Ohnishi, "Medical mechatronics—An application to haptic forceps," *Annu. Rev. Control*, vol. 29, no. 2, pp. 237–245, 2005.

[51] M. Bordegoni and F. Ferrise, "Designing interaction with consumer products in a multisensory VR environment," *VIRTUAL AND PHYSICAL PROTOTYPING*, vol. 8. pp. 1–21, 2013.

[52] R. Y. Wang and J. Popović, "Real-time hand-tracking with a color glove," in *ACM Transactions on Graphics (TOG)*, 2009, vol. 28, no. 3, p. 63.

[53] S. Melax, L. Keselman, and S. Orsten, "Dynamics based 3D skeletal hand tracking," in *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 2013, p. 184.

[54] V. A. Prisacariu and I. Reid, "3D hand tracking for human computer interaction," *Image Vis. Comput.*, vol. 30, no. 3, pp. 236–250, 2012.

[55] L. E. Potter, J. Araullo, and L. Carter, "The Leap Motion Controller: A View on Sign Language," in *Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration*, 2013, pp. 175–178.

[56] F. Weichert, D. Bachmann, B. Rudak, and D. Fisseler, "Analysis of the Accuracy and Robustness of the Leap Motion Controller," *Sensors*, vol. 13, no. 5, pp. 6380–6393, 2013.

[57] S. Sridhar, A. Oulasvirta, and C. Theobalt, "Fast Tracking of Hand and Finger Articulations Using a Single Depth Camera," 2015.

[58] Duo3D. Available: https://duo3d.com/.

115

[59] G. Colombo, F. De Angelis, and L. Formentini, "Integration of virtual reality and haptics to carry out ergonomic tests on virtual control boards," *Int. J. Prod. Dev.*, vol. 11, no. 1, pp. 47–61, 2010.

[60] K. Ohnishi, S. Katsura, and T. Shimono, "Motion control for real-world haptics," *Ind. Electron. Mag. IEEE*, vol. 4, no. 2, pp. 16–19, 2010.

[61] A. Ninu, S. Dosen, D. Farina, F. Rattay, and H. Dietl, "A novel wearable vibro-tactile haptic device," in *Consumer Electronics (ICCE), 2013 IEEE International Conference on*, 2013, pp. 51–52.

[62] T. R. Coles, D. Meglan, and N. W. John, "The role of haptics in medical training simulators: a survey of the state of the art," *Haptics, IEEE Trans.*, vol. 4, no. 1, pp. 51–66, 2011.

[63] M. Abu-Tair and A. Marshall, "An empirical model for multi-contact point haptic network traffic," in *Proceedings of the 2nd International Conference on Immersive Telecommunications*, 2009, p. 15.

[64] NuiGroup. Available: http://nuigroup.com/go/lite.

[65] D. Wigdor and D. Wixon, *Brave NUI world: designing natural user interfaces for touch and gesture*. Elsevier, 2011.

[66] D. Kaushik, R. Jain, and others, "Natural User Interfaces: Trend in Virtual Interaction," *arXiv Prepr. arXiv1405.0101*, 2014.

[67] D. Delimarschi, G. Swartzendruber, and H. Kagdi, "Enabling Integrated Development Environments with Natural User Interface Interactions," 2014.

[68] G. Colombo, G. Facoetti, C. Rizzi, and A. Vitali, "A preliminary study of new interaction devices to enhance virtual socket design," 2014, pp. 1–8.

[69] U. Lee and J. Tanaka, "Finger identification and hand gesture recognition techniques for natural user interface," in *Proceedings of the 11th Asia Pacific Conference on Computer Human Interaction*, 2013, pp. 274–279.

[70] G. Colombo, C. Rizzi, G. Facoetti, and A. Vitali, "Automatic Generation of Software Interfaces for Hand-Tracking Devices," in *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2015, pp. V01BT02A057–V01BT02A057.

[71] I. D. E. Eclipse, "The Eclipse Foundation." 2007.

[72] J. Musset, É. Juliot, S. Lacrampe, W. Piers, C. Brun, L. Goubet, Y. Lussaud, and F. Allilaire, "Acceleo user guide," 2006.

[73] U. Meier, O. López, C. Monserrat, M. C. Juan, and M. Alcaniz, "Real-time deformable models for surgery simulation: a survey," *Comput. Methods Programs Biomed.*, vol. 77, no. 3, pp. 183–197, 2005.

[74] J. Schmid and N. Magnenat-Thalmann, "MRI bone segmentation using deformable models and shape priors," in *Medical Image Computing and Computer-Assisted Intervention--MICCAI 2008*, Springer, 2008, pp. 119–126.

[75] D. Li, N. Dai, X. Jiang, and X. Chen, "Interior structural optimization based on the density-variable shape modeling of 3D printed objects," *Int. J. Adv. Manuf. Technol.*, pp. 1–9, 2015.

[76] N. Umetani and R. Schmidt, "Cross-sectional structural analysis for 3d printing optimization," *SIGGRAPH Asia*, p. 5, 2013.

[77] X. Zhang, Y. Xia, J. Wang, Z. Yang, C. Tu, and W. Wang, "Medial axis tree—an internal supporting structure for 3D printing," *Comput. Aided Geom. Des.*, vol. 35–36, pp. 149–162, 2015.

[78] J. Giesen, B. Miklos, M. Pauly, and C. Wormser, "The Scale Axis Transform," in *Proceedings of the*

Twenty-fifth Annual Symposium on Computational Geometry, 2009, pp. 106–115.

[79] A. N. Christiansen, R. Schmidt, and J. A. Bærentzen, "Automatic balancing of 3D models," *Comput. Des.*, vol. 58, pp. 236–241, 2015.

[80] L. Lu, A. Sharf, H. Zhao, Y. Wei, Q. Fan, X. Chen, Y. Savoye, C. Tu, D. Cohen-Or, and B. Chen, "Build-to-last: Strength to Weight 3D Printed Objects," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 97:1–97:10, 2014.

[81] J. Panetta, Q. Zhou, L. Malomo, N. Pietroni, P. Cignoni, and D. Zorin, "Elastic Textures for Additive Fabrication," *ACM Trans. Graph.*, vol. 34, no. 4, pp. 135:1–135:12, 2015.

[82] C. Murphy, "3D Printing infill using Biomimetics," 2015. .

[83] W. Gao, Y. Zhang, D. Ramanujan, K. Ramani, Y. Chen, C. B. Williams, C. C. L. Wang, Y. C. Shin, S. Zhang, and P. D. Zavattieri, "The status, challenges, and future of additive manufacturing in engineering," *Comput. Des.*, vol. 69, pp. 65–89, 2015.

[84] D. A. Gibson, "Atlas of Limb Prosthetics," *Can. Med. Assoc. J.*, vol. 127, no. 4, p. 324, 1982.

[85] T.-H. Kwok and C. C. L. Wang, "Shape optimization for human-centric products with standardized components," *Comput. Des.*, vol. 52, no. 0, pp. 40–50, 2014.

[86] I. Oikonomidis, N. Kyriazis, and A. A. Argyros, "Efficient model-based 3D tracking of hand articulations using Kinect.," in *BMVC*, 2011, vol. 1, no. 2, p. 3.

[87] K.-Y. Yeung, T.-H. Kwok, and C. C. L. Wang, "Improved Skeleton Tracking by Duplex Kinects: A Practical Approach for Real-Time Applications," *J. Comput. Inf. Sci. Eng.*, vol. 13, no. 4, p. 41007, 2013.

[88] D. Regazzoni, G. de Vecchi, and C. Rizzi, "RGB cams vs RGB-D sensors: Low cost motion capture technologies performances and limitations," *J. Manuf. Syst.*, vol. 33, no. 4, pp. 719–728, 2014.

[89] X. Xu and R. W. McGorry, "The validity of the first and second generation Microsoft Kinect™ for identifying joint center locations during static postures," *Appl. Ergon.*, vol. 49, no. 0, pp. 47–54, 2015.

[90] J. Smisek, M. Jancosek, and T. Pajdla, "3D with Kinect," in *Consumer Depth Cameras for Computer Vision*, A. Fossati, J. Gall, H. Grabner, X. Ren, and K. Konolige, Eds. Springer London, 2013, pp. 3–25.

[91] Z. Zhang, "Microsoft Kinect Sensor and Its Effect," *MultiMedia, IEEE*, vol. 19, no. 2, pp. 4–10, Feb. 2012.

[92] H. G. Hoffman, W. J. Meyer III, M. Ramirez, L. Roberts, E. J. Seibel, B. Atzori, S. R. Sharar, and D. R. Patterson, "Feasibility of Articulated Arm Mounted Oculus Rift Virtual Reality Goggles for Adjunctive Pain Control During Occupational Therapy in Pediatric Burn Patients," *Cyberpsychology, Behav. Soc. Netw.*, vol. 17, no. 6, pp. 397–401, 2014.

[93] T. Kot and P. Novák, "Utilization of the Oculus Rift HMD in Mobile Robot Teleoperation," in *Applied Mechanics and Materials*, 2014, vol. 555, pp. 199–208.

Aknowledgments

First of all I have to thank my family, for their constant support during all moments of these last three years. Without them, it would have been almost impossible to realize my PhD.

Thanks to all my past and present colleagues of VK Group at University of Bergamo, for their friendship and patience during these last years together. A special thanks to Prof. Daniele Regazzoni, Giancarlo Facoetti and Claudio Comotti. I shared with them many hours of hard work, but also much satisfaction for the scientific results reached so far.

An immense thanks to Prof. Caterina Rizzi, who has made possible this thesis with her support and the great professionality and experience. Thanks for everything!

Finally a special thanks to all the staff of Ortopedia Panini that allowed me to partake their great experience and knowledge and to Fondazione Cariplo, which has partially funded my research activity within the framework of I4BIO project.