A Physics-based Virtual Reality Framework for Medical Training and Simulation

by Xiao Xiao

B.S. in Electrical Engineering, May 2011, University of Science and Technology Beijing
M.S. in Computer Science, May 2014, The George Washington University

A Dissertation submitted to

The Faculty of
The School of Engineering and Applied Science
of The George Washington University
in partial satisfaction of the requirements
for the degree of Doctor of Philosophy

May 17, 2020

Dissertation directed by

James K. Hahn
Professor of Computer Science

The School of Engineering and Applied Science of The George Washington University certifies that Xiao Xiao has passed the Final Examination for the degree of Doctor of Philosophy as of April 15, 2020. This is the final and approved form of the dissertation.

A Physics-based Virtual Reality Framework for Medical Training and Simulation

Xiao Xiao

Dissertation Research Committee:

James K. Hahn, Professor of Computer Science,
Dissertation Director

Robert Pless, Professor of Computer Science,
Committee Member

Aylin Caliskan, Assistant Professor of Computer Science,
Committee Member

Xiaoke Zhang, Assistant Professor of Statistics,
Committee Member

Acknowledgments

I would like to thank my advisor, Dr. James Hahn, for giving me the opportunity to be a part of his research team and for guiding me throughout my studies and research at George Washington University. He expanded my knowledge in many areas by providing a productive environment where research ideas were always welcome to be discussed.

I would also like to extend my thanks to Dr. Robert Pless, Dr. Aylin Caliskan, and Dr. Xiaoke Zhang for agreeing to be in my dissertation committee, and for all their help and cooperation with the dissertation process. I especially would like to thank Dr. Xiaoke Zhang who provided valuable inputs and generous support to my research work.

I would also like to thank all of my lab colleagues, especially Shang Zhao and Wei Li for their help with my research work as well as all the neonatologists at National Children's Hospital who participated in the validation studies.

This work has been sponsored by the National Institutes of Health grant R01HD091179.

And last but not least, I would like to extend my warmest thanks to my parents for always believing in me and supporting me through the challenging times that were part of the process of me becoming an experienced researcher.

Abstract

A Physics-based Virtual Reality Framework for Medical Training and Simulation

Medical simulation offers the opportunity to revolutionize the training of medical professionals, from paramedics to physicians and surgeons, allowing early learning to occur in a safe, controllable, and configurable virtual environment, without putting patients at risk. However, the complexity of the problems involved in the development of medical training systems as well as the spectrum of scientific fields that need to be covered have been the major limiting factors for the achievement of realistic simulations. The current state of the field of medical simulation is characterized by scattered laboratories or commercial entities using a variety of models that are neither interoperable nor independently verifiable, resulting in a steep development curve, duplication of efforts, high cost for simulators, and slow adoption of the technology. There have been some attempts to provide open-source standardization. However, they offer a collage of specialized solvers for different substances (bones, tissues, fluids, etc.), which creates redundant work and does not provide stable and efficient two-way interactions between all object types. More importantly, most of the current available simulation systems do not provide a means for automated assessment that can record, visualize, and analyze trainees' performances through quantitative measures, which is an often neglected aspect of medical training.

This dissertation addresses the issues of medical simulation in an attempt to bridge the gap left by previous works. We propose a practical and efficient virtual reality simulation framework that converts the training of medical procedures to a completely immersive virtual environment where both visual and physical realism are achieved. Our generalizable framework embeds independent dynamics models and interaction devices in separate modules while allows them to interact with each other within the same environment, which offers a flexible solution for multi-modal medical simulation scenarios and enables new simulators to be built efficiently. Our framework includes simulation of all human body

constituents, such as bones, soft tissues, and fluids (e.g., blood, secretions), in a unified particle representation using position-based dynamics, which enables different simulated substances to interact with each other seamlessly and allows for efficient modeling of large objects with different properties in real time. Our system supports inputs of patient-specific anatomies from many sources (serial-section, volumetric, or surface scans), which provides realistic anatomical structures and can be parameterized to allow variations in a range of features that affect the level of difficulty. Most importantly, with virtual representation of all the components involved in the procedure, our automated assessment system can capture and visualize a whole set of performance parameters of the instruments in relation to the geometric change of the virtual model for real-time guidance and post-trial assessment. In addition, an interpretable automated scoring algorithm is developed that uses machine learning to mimic the evaluation of human raters. Finally, we demonstrate the utility of the framework by developing a test-bed application for neonatal endotracheal intubation. The clinical realism of the VR simulator and the validity of the automated assessment system were evaluated with a group of neonatologists using qualitative and quantitative measures.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

Virtual Reality (VR) is a way for humans to visualize, manipulate, and interact with computers and complex data. This technology attempts to immerse the user into a computer-generated world representing reality. The user can interact with the virtual world and directly manipulate objects in it. Most of the current VR environments are primarily visual experiences, rendered either on a computer screen or through head-mounted displays (HMD). There are some advanced systems, which include additional sensory modalities, such as tactile information through haptic or force feedback devices.

VR has a wide variety of applications in many different fields, such as video games and flight simulation. VR-based flight simulation, as a classic success story of VR applications, has proven the feasibility to build reliable virtual training systems that are not only virtual but also subjectively real. It artificially re-creates aircrafts and the environment in which it flies and has been extensively used by the aviation industry for the design and development of prototype airplanes and training of pilots. VR is also finding its way into the training of healthcare professionals. VR based medical simulators recreate actual procedures in a virtual environment, which is very useful for the purpose of training, rehearsal, or experiment. Virtual procedures are carried out with the help of advanced techniques and devices in the fields of electronics, robotics, and computer graphics to create human-machine interfaces and to provide visual and force feedback to the user.

Virtual training systems can improve trainees' learning curves with safety, efficiency, flexibility, and without the fear and anxiety of performing live procedures on patients before entering the operating room (OR), thereby improving patient safety. VR simulators have been expected to be an important part of medical training since the early 1990s [74]. A survey shows that 83% of the directors of surgery training programs believe that there is a role for computer-based training systems in surgical training [29][31]. While their initial

cost might seem expensive, VR simulators can in fact be cost-effective when considering the broader economic benefits of better-trained medical professionals and resource optimization [9], including savings on instructor time, error reduction and faster completion times [76][1]. Moreover, VR simulators can be used to explore new ways of performing a procedure, prototyping medical equipment or getting familiar with new surgical techniques or devices [62]. Recent reviews show that, although VR simulation is successfully used in various medical domains, there is still enormous potential for further development [25][14][60].

1.1 Motivation

Various training methods have been used in order to train medical students for live procedures. The closer the model is to the real patient, the greater the benefits. Many of the current methods have some serious drawbacks. In traditional teaching, performing live operations under the supervision of experienced experts has been a practical solution in many scenarios. However, opportunities to practice skills are often limited by availability of instructors and access to willing patients. Practicing of any non-expert on patients exposes these patients to increased risk. In addition, it will inevitably involve a long learning curve for the trainee to gain adequate skills to become a qualified surgeon. This apprenticeship model is based on a high-volume, hands-on training with gradually decaying level of supervision, until the trainee is judged by the mentor to be competent enough to operate on his/her own. Improvements in quality and safety standards in this kind of training yields the apprenticeship model insufficient to create a competent surgeon in a patient-safe way. As a result, the pressure on training outside the OR has increased. The use of human cadavers gives good anatomical accuracy, however, the tissue response is not that close to live tissue. There is also limited availability and high cost associated with storage. When trainee is practicing on these subjects, there is little indication as to the success of the attempt, and the lack of readily available new cadavers restricts their use in everyday training. An alternative that can remedy the issue of the lack of realistic tissue response is to use live

animals. Unfortunately, this comes with additional ethical issues, and the anatomy may be quite different from humans'. Using inexpensive, low-fidelity physical task trainers or manikins can alleviate ethical concerns and allow learners to achieve some level of competence with greater repeatability, however, silicon, foam or plastic parts used in task trainers, lack physiological behaviour and different biomechanical properties compared to human tissue and provide little variation in anatomy or difficulty level. Hence, these methods do not provide sufficient realism. Consequently, learners who train on one manikin may develop model-specific techniques that may not transfer to patients or even other manikins. These manikins might also be limited in the types of procedure that can be practiced on them. Lastly, assessment of trainees' performance under artificially ideal conditions likely overestimate their skill level since they do not mimic the stressors and distractions that are inherent in the real clinical environment. Thus, there is a pressing need for innovative training modalities that can bridge the gap left by traditional training methods and thereby allow rapid skill acquisition.

The solution presented to remedy the concerns of previous training methods is to create a virtual training system with realistic visual/haptic feedback. High-fidelity VR-based training systems offer an elegant solution to the current need for better training in the medical field. They provide a realistic, controllable, and configurable training environment free from ethical issues in which trainees can repetitively practice without putting patients at risk. In addition, some of the rare pathological conditions and emergency procedures can only be recreated and trained on a simulator. These tasks can be scheduled without additional cost concerns. Virtual trainers initiate scenarios with each starting point pertinent to individual progress of each trainee, instead of restarting from the beginning each time. Once created, a virtual trainer can be replicated to the desired degree. A virtual training can be converted into a rehearsal or simulation planning by an update of patient-specific input. Virtual trainers have completely control over their virtual environments to adjust variations or change levels of difficulty, such that every simulated object can be selectively presented, modified or

concealed, for expected training configurations that are not available in the real world. Finally, simulators make training and skills acquisition more efficient through the use of quantifiable measures. The simulation can provide quantitative feedback as to the trainees' skill and areas requiring improvement. It is a great learning tool where the trainee is allowed to go back and see what went wrong, and how to improve, with an objective evaluation of the trainees' dexterity. It brings more engagement and realism to the process. After training with these methods, the next step is to move on to supervised live practice trials. In this case, it is important that the trainee comes into the OR having been well practiced and with confidence in order to reduce patient harm.

1.2 Problem Statement

The design and development of a VR medical simulator is a complex process requiring the integration, within a single environment, of leading-edge solutions from multiple disciplines, such as computing, engineering, physics, medicine, and human factors. As a result, it can be laborious and time consuming for developers with expertise in even a few of these disciplines to build a simulator that can pass validation and be ready for clinical deployment. Despite the fact that many of the requisite technologies have matured to a level that supports the degree of realism required, the implementation of simulation software for the realistic and compelling recreation of medical procedures is still a challenging and open problem.

As the capability of computing resources has increased over the years, the complexities of medical procedures being simulated have also increased. A typical medical scene now involves much more than just tool-tissue interactions. Simulations of multiple tissues and organs, medical instruments, and devices, coupled fluid flows due to bleeding and modelling of the physiological consequences of procedures are becoming common. To achieve a certain degree of realism in such complex interaction scenarios, the simulation environments need to support the following:

- Modeling heterogeneous scenes composed of different substances (bones, tissues, and

fluids) with complex geometry and material properties;

- Dynamic and real-time interactions (palpation, probing, grasping, cutting, cauterizing, etc.) between virtual objects and various instruments physically manipulated by the user;
- Robust collision detection/resolution for frequent tool-tissue interactions;
- Multi-modal (visual, haptic, and auditory) rendering of the results to the user.

Among these components, deformable modeling and collision detection are two bottlenecks of real-time computation. Besides the above important components, an effective medical training system should also include an automated assessment component which can provide automated feedback during/after each training session and quantitatively evaluate and measure trainees' performances.

Currently, VR medical simulators tend to be constructed independently of any universally accepted systematic framework. Each laboratory or commercial entity tends to use its own library of proprietary software, resulting in an *ad-hoc* collection of simulation engines, application program interfaces (API), and software development kits (SDK). This has resulted in a steep development curve, duplication of efforts, high cost for simulators, and slow adoption of the technology. This is a major impediment to wide acceptance and use of VR simulators, since each new medical domain for simulators tends to result in a new stand-alone system.

There have been some attempts to provide open-source standardization. However, they offer a collage of specialized solvers for different substances, such as rigid, deformable, and fluid objects, which creates redundant work and does not provide stable and efficient two-way interactions between all object types. They also do not provide a means for automated assessment that can record, analyze, and visualize trainees' performances, which is a neglected aspect of medical simulators.

In response to this need, we have developed a practical and efficient simulation and training framework that includes a stable and efficient modeling method for seamless simulation of all object types as well as the capability of incorporating an automated assessment system. Our generalizable framework will enable new simulators to be built efficiently. This work will significantly advance the practical use of VR simulators in a variety of procedures. We will demonstrate the validity of the framework by developing an test-bed application for neonatal endotracheal intubation (ETI) simulation, which captures the most important aspects of the aforementioned general problem we are trying to address.

1.3 Proposed Solution

Our work aims at developing and evaluating a physics-based VR simulation framework for medical training and simulation. This involves designing and implementing software solutions combining realistic simulation, visualization and interactions. The software will be integrated with third-party displays, tracking, and haptic devices in order to deliver a complete VR training experience. The simulation framework presented in this dissertation improves the effectiveness of VR medical simulation and training by addressing the following aims:

- Build a fully interactive VR simulation system that integrates, in the same environment, different dynamics models and interaction devices;
- Offer a completely immersive virtual environment where both visual and physical realism are achieved;
- Support realistic anatomical models from many sources of patient-specific anatomies that can parametrically adjust features that vary the levels of difficulty;
- Simulate heterogeneous human body constituents, including bones, soft tissues, and fluids, in a unified solver based on position-based dynamics (PBD);

- Evaluate the clinical realism of the VR simulator by conducting a validation study using quantitative and qualitative measures;
- Develop an automated assessment that can captures all motions and gives a complete visualization of the procedure for real-time guidance and post-trial assessment;
  - Our automated visualization tools can visualize a whole set of performance parameters that our experienced instructors consider important in their subjective assessment of performance.
  - More importantly, our automated scoring system provides an interpretable automated prediction model that uses a machine learning algorithm to mimic the evaluation of human raters.

1.4 Organization of Dissertation

The dissertation is structured as follows. Chapter 2 critically reviews and summarizes the current state-of-the-art of VR medical simulations and the relevant existing frameworks that are available for VR medical simulation development. Chapter 3 presents the main contributions of our work. The design of the generalizable simulation framework and details of the implementation are explained. Chapter 4 describes the core simulation modeling method of our framework, which is based on PBD to simulate hybrid scenarios, such as rigid body, soft body, and fluids, and their interactions in a unified particle representation. Chapter 5 describes an application that is implemented using the proposed framework, which is the simulation of neonatal ETI, and presents the concurrent validation study of the ETI simulation system compared to traditional manikin-based simulator. Chapter 6 describes the automated assessment algorithms for ETI performance and the results of the experiment. Finally, Chapter 7 concludes the dissertation and proposes possible future work.

Chapter 2: Related Work

This chapter explores the benefits of VR simulation-based medical training and reviews the current state-of-the-art of VR medical simulators and the relevant existing frameworks that are available for VR medical simulation development.

2.1 Virtual Reality Medical Simulators

VR-based medical simulator simulates an environment for the purpose of surgery or medical procedure training. It provides a safe, real-time interactive deformation of the organs and haptic feedback for different kinds of tool-tissue interactions. It helps medical professionals get familiar with the anatomy and practice the skills of the procedure prior to clinic exposure. VR simulation has come a long way in the past decade and has now reached a point where it has been demonstrated to effectively improve learning outcomes in clinical settings [76]. Depending on how much tissue deformation is involved in the procedure, the simulation complexity can be categorized into classes from no deformation, such as dental and bone surgery, to small deformation, such as bronchoscopy, colonoscopy, and endoscopy, and to large and complex deformation, such as laryngoscopy and laparoscopy. This section details a selection of the most relevant VR medical simulations currently available.

2.1.1 Dental and Bone Surgery Simulators

The main interaction for both dental and bone surgery is the manipulation and modelling of rigid structures (bones or teeth) with surgical tools. Therefore, the methods developed for these simulators are very similar. These simulators commonly represent rigid structures as volumetric models, which is referred to as voxel-based surgical simulation.

Several interactive VR medical training simulations with haptic feedback have been developed, including Temporal Bone Surgery [53], Craniofacial Surgery [86], and Dental

Surgery [81]. The type of interaction supported by each of the simulators is very similar: volumetric tissue removal of rigid bone (or tooth) tissue. The haptic feedback is computed in real time from the interaction of the tool with the voxel-based representation of the bone or tooth. Moreover, additional fidelity and effect is added to simulate the forces resulting from the motion of the abrading tool tip. However, this approach does not support volumetric tissue removal from deformable tissues.

2.1.2 Endotracheal Intubation Simulators

ETI is a time sensative and life-saving procedure that is commonly performed on patients in order to maintain a clear airway and for administering a general anaesthetic [30]. The procedure can be briefly summarized as using a laryngoscope to manipulate the tongue and epiglottis, and insert a tube into the trachea.

The interaction of the laryngoscope with the tongue and the epiglottis are the keys to the success of the simulator. The tongue and the epiglottis must deform realistically in real time in response to contact with the laryngoscope. Realistic visco-elastic modelling of the deforming tongue is especially challenging. The tongue is a large muscle with internal variability and dynamic behaviour even when the patient is unconscious. ETI simulation is often associated with physical manikin simulators rather than computer-based simulators. A few VR ETI simulation systems have been developed; however, most of them allow limited interactions and/or do not simulate the physics of the virtual models.

Rodrigues et al. were the first to create a real-time interactive mechanical model of the tongue [69][70][71]. Their biomechanical model of the upper airway included the tongue, ligaments, larynx, vocal cords, and bony landmarks at relatively low resolution in order to achieve real-time performance [71]. The finite element method (FEM) was employed to simulate the behaviour of the tongue, from simple linear elastic material to complex non-linear visco-elastic material [71]. The described configuration of the mechanical system is intricate, which includes specialised hidden mechanical interconnections to impart the

desired properties [69]. Empirical validation of the model was performed, which showed that the non-linear model behaves most closely to the experimental studies [70]. The visual quality of their simulator is poor by today's standards; models consist of low numbers of flat-shaded polygons with no textures or realistic surface details. The simulation and insertion of the endotracheal tube (ETT) were not implemented. Mayrose et al. developed an intubation simulator that uses mass-spring model for soft tissue deformation [48][49]. In order to achieve real-time performance, the system only allows the interactions of the laryngoscope blade and the ETT with relevant soft tissues of the virtual model, which is not fully interactive. The laryngoscope and the ETT are manipulated through a single haptic device, which does not allow bimanual operation. The graphics are projected through a semi-transparent mirror to allow the user to view through 3D glasses, which does not provide a fully immersive experience in a virtual environment. Demirel et al. [18] developed a virtual airway skills trainer in which only two tasks prior to the ETI procedure were simulated: mallampati scoring and placing the patient in an optimal intubation position. Although their system provides an immersive experience with HMD, it does not include simulation of the actual intubation, which is more challenging and critical for ETI training.

An commerical airway lab VR simulator has been developed by Arch Virtual®, in which high-quality visual realism was achieved. However, their system only allows limited interactions, such as the insertion of the endotracheal tube. Soft tissue deformation and collision detections between the laryngoscope and the soft tissue/tube were not implemented. Xia et al. [87] developed an interactive virtual ETI simulator, in which both visual and physical realism were achieved. The laryngoscope and the endotracheal tube were each attached to haptic devices, which allow for bimanual interactions during the procedure. However, their system only simulated the deformation of the tongue and collisions between the laryngoscope and tongue, which is not fully interactive. Rajeswaran et al. [65] developed a VR-based immersive trainer consisting of two training modes: the animation mode provides essential knowledge of the ETI procedure and visualization of the process of ETI

in 3D space while the tutorial mode allows the user to carry out proper intubation in a virtual environment by following step-by step on-screen instructions. However, their virtual model only allows limited interactions, such as head elevation and jaw opening. Neither collision detection of the tools with the virtual model nor soft tissue deformation was implemented, both of which are essential for realistic simulation. The laryngoscope and the ETT are controlled by two hand controllers, which is not intuitive and does not allow force feedback.

2.1.3 Minimally Invasive Surgery Simulators

Minimally invasive surgical (MIS) procedures (also known as keyhole surgery), such as laparoscopy, is a modern surgical procedure in which operations in the abdomen are performed through small incisions (usually 2-3 cm) as compared to open body surgery. A tiny laparoscope, a telescopic rod lens system, with a video camera and a halogen or xenon light source, is inserted through the incision to view the operative field. The surgery is performed by observing the image received from the laparoscope on a computer monitor.

MIS procedures are well suited to VR simulation mostly because the user interface is relatively easy to replace with equivalent devices interfaced to a computer. When performing a real MIS procedure, the operator is guided by visual feedback from an fiber optic camera displayed on a monitor, and haptic feedback via the handpieces of the surgical instruments. Likewise, endoscopic procedures use visual feedback via the endoscope (presented on a monitor) and haptic feedback via the endoscope handle.

Most of the current research on MIS simulation focuses on very specific topics, such as soft tissue deformation for surgical simulation [78], surgical cutting/dissection [52] etc., rather than the practical framework. The available laparocopic surgery simulators in the market, such as LapSim®, aim only for training basic skills following strict and relatively simplistic routines. They are not capable of simulating the whole surgery procedure. More recently, the state-of-the-art Simbionix® LAP Mentor™ simulator provides a structured curriculum with different difficulty levels for training and monitoring basic laparoscopic

skills as well as advanced full laparoscopic procedures. The system has the capabilities of simulating a set of specially tailored and designed techniques, ranging from modeling of complex anatomy, different tissue types and dissection planes, and realistic tool-tissue interaction.

Aside from the simulators developed commercially, several MIS simulators have been developed by research groups. Several arthroscopy simulators have been developed [27][92]. These simulators are similar to those bone surgeries as described previously because they typically make minimal use of deformable structures and all interactions are via specialised (keyhole) instruments. Hellier et al. [36][72] have developed a multi-threaded colonoscopy simulation framework that provides robust deformation simulation of the colon with high visual fidelity and a specially built haptic interface that enables the user to manipulate a real endoscope with tactile feedback. Their work provides an excellent solution for cavity simulators. However, surgical simulators and VR medical simulations more generally cannot be developed using this framework without additional technology. Harders et al. [8][32] have developed an impressive hysteroscopy simulator that renders the uterine cavity with high visual fidelity and allows the user to cut deformable polyps and myomas. Ablation is also supported. Their approach to surgical cutting is optimized for their specific domain and is relatively complex, requiring further work before arbitrary cut paths in 3D are supported.

2.2 Simulation Frameworks

The development of medical VR simulators is complex and combines numerous technologies and techniques. Unlike previously reviewed VR medical simulators that tend to be constructed independently of any universally accepted systematic framework, there have been some attempts to provide open-source standardization. This section provides an overview of the most significant simulation frameworks available for VR medical simulation development. These frameworks combine some or all of other third-party APIs into a single, unified API.

In recent years, there have been a number of attempts at designing software frameworks to facilitate the development of medical simulation. Examples include CAML [13], ISReal [40], NeuroVR [67], Spring [51], and AlaDyn-3D [39]. Some of these early frameworks only provided conceptual architecture for the integration of medical simulation components, but no implementation issues or technical considerations were addressed. Some of them have capabilities of simulating dymanic models, however, the underlying models are woven into the specifications of the overall frameworks developed, with a limited choice of algorithms for physical modeling, time integration and collision detection.

Later, an open architecture framework, GiPSi (General Physical Simulation Interface), was developed for organ-level surgical simulation [28]. Unlike earlier dynamic modeling frameworks, the APIs in GiPSi were designed to be independent of the specifics of the implemented modeling methods. This allows GiPSi to integrate heterogeneous models and processes and enforce time-dependent spatial relationships among them. It integrates open source libraries, such as TAO, OPCODE, and OpenHaptics, to facilitate development of simulation components for medical simulation with haptic feedback. However, this framework no longer appears to be active and it does not include any re-usable components for tissue simulation.

More recently, another open source framework called SOFA (Simulation Open Framework Architecture) [21] was developed to primarily target at real-time simulation, with an emphasis on medical simulation. Although their aim is identical to previous frameworks, they propose a different approach, through a very modular and flexible software framework that allows independently developed algorithms to interact together within a common simulation. Different types of deformable models are supported based on traditional force-based modeling methods, such as mass-springs, linear and co-rotational finite element method (FEM), and fluid models along with a large array of collision detection and resolution algorithms are provided. Although SOFA has a lot to offer, it requires developers to conform or adapt to its architecture.

A more advanced open-source toolkit called iMSTK [42] was released recently for rapid prototyping of interactive simulation applications. iMSTK provides an easy to use framework that can be extended and interfaced with other third-party libraries for the development of medical simulators without restrictive licenses. Besides traditional force-based physical modeling methods, iMSTK has included PBD [54], which has gained great popularity in recent years for simulating large deformation tissues with high efficiency, stability, and real-time performance. In addition, the input/output module for visualization and haptics supports over 30 devices, including various haptic devices and VR HMD. However, this software was only available two years after our simulation framework was developed in early 2019.

In summary, the number of VR medical simulations is growing. While graphical realism is high, realistic interactive tissues and haptic feedback remain a key challenge. Although our aim is similar to some of the previous work, our framework enables the rapid and efficient development of medical VR simulators that integrates PBD for real-time simulation of heterogeneous human body constituents, including bones, soft tissues, and fluids, in a unified solver instead of a collage of specialized solvers for different object types in previous frameworks. Our framework not only provides realistic simulation of the procedure that allows full interactions with realistic haptic feedback, but also offers important components for training and skill assessment. Unlike most simulators only focusing on training of basic skills with little variations, our system supports input of realistic patient-specific profiles from many sources with different pathologies that can parametrically adjust features to cover a wide variation of difficulty levels. More importantly, due to various unsolved technical issues, current simulation systems are not able to capture precise motion data since they cannot simulate the entire procedure in a realistic manner. In contrast, our framework provides the components required for automated assessment that captures all motions and gives a complete visualization of the procedure for real-time guidance and post-trial assessment, which are not available in current medical simulation frameworks.

Chapter 3: Physics-based VR Simulation Framework

Many medical simulations involve the computation of visual feedback, haptic feedback and interactions between medical instruments and anatomical structures (such as probing, grasping, cutting, cauterizing, etc.). This typically translates into a simulation loop where, at each time step, collisions between objects are detected, deformation and collision resolution are computed, and the resulting state can be visually and haptically rendered (Figure 3.1). By observing the commonalities in these procedures, our proposed system is designed to suit many different medical procedures, and thus, allows easy transition from one class of procedures to another class of procedures.

The proposed system will be a general simulation framework with a broad base of technological features and an emphasis on real-time simulation and automated assessment. This will offer a highly novel flexible platform for the quick and efficient development of new medical simulators. It will allow for the input of patient-specific anatomy acquired from computed tomography (CT)/magnetic resonance imaging (MRI) scans. It will perform soft-tissue modeling, rigid-body dynamics, and fluid interaction and interfaces to many different interaction devices. Our system embeds independent dynamics models and interaction devices in separate modules while allowing them to interact with each other within the same environment, thereby minimizing the efforts required for development.

3.1 Architecture

The overall system architecture of the proposed framework is provided in Figure 3.2. It consists of a main simulation engine, an object representation structure (object class), an abstraction of I/O devices (sensor class), and an automated assessment component, in which includes the automated visualization and scoring implemented for medical simulation.

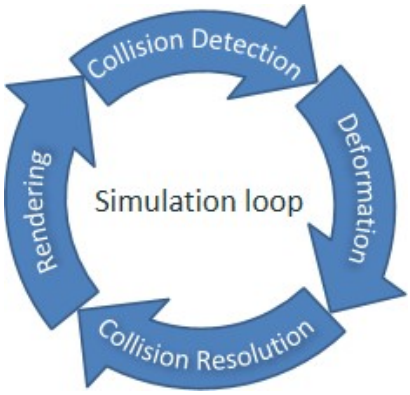

Figure 3.1: Illustration of the simulation loop.

3.2 Main Program

The main simulation program acts as the central core, which contains the main thread(s) of control, simulation functions, collision detection/resolution methods, data structures, and interface routines. Its tasks include the management of the dynamics of objects (deformable, rigid bodies, and fluids), coordination of their interactions, and arbitration of the communication between components.

3.3 Anatomy Acquisition

Our simulation system supports input of patient-specific anatomy from many sources including serial-section, volumetric (CT/MR scans), or surface scans segmented using any segmentation tools, followed by mesh generation and simplification. Once the geometry is created using these tools, it can be loaded into the system and various attributes can be specified. Each object comprises arrays of geometry attributes (including vertices, edges, faces, and/or tetrahedral elements), graphical attributes (such as textures, material properties, etc), simulation properties (such as dynamics, numerical method, constraints, etc), collision detection and resolution attributes (such as detection method, faces to consider, collision filter, etc), and other attributes. Note that, for a particular object, it may not contain all of the above geometry attributes. Fluids, such as blood, mouth secretions, are represented as

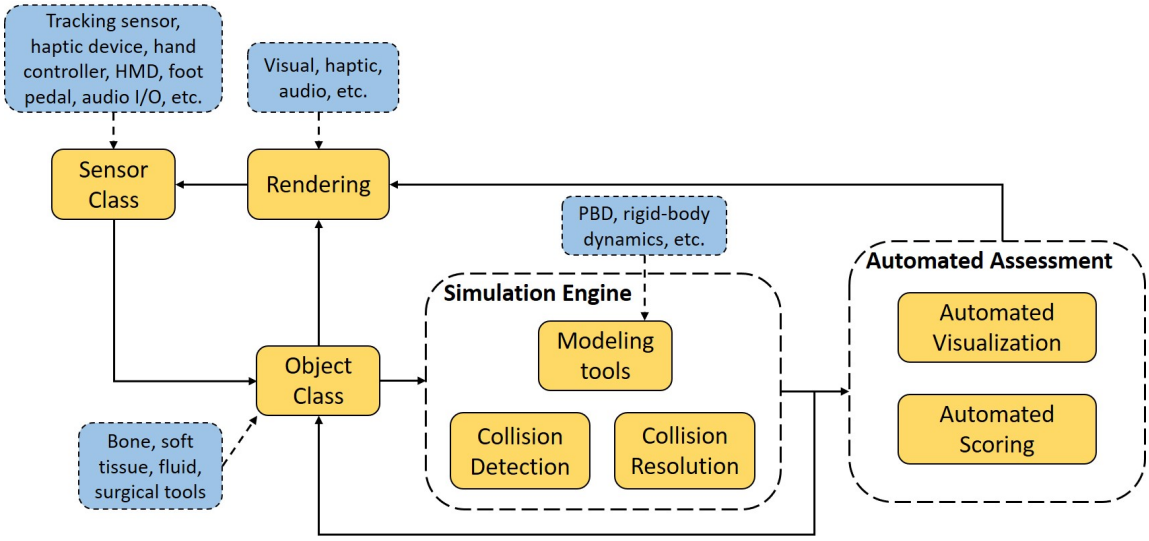

Figure 3.2: The overview of the proposed architecture.

an object consisting only of nodes. A suture/thread is modeled as an object with only nodes and edges. Most objects (such as organs, medical instruments) can be modeled with nodes, edges, and faces, while tetrahedrons are only required for some forms of cutting or volume preservation.

3.4 Object Representation

In our system, each simulated object can have several representations linked together through skinning (Figure 3.3). In this multi-model representation, each object is decomposed into various representations where each representation is more suited toward a particular task - modeling, collision detection and visualization. This multi-model representation permits, for instance, switching from one modeling or collision detection method to another, without affecting other components and allows new models to be integrated efficiently. These representations are linked together so they can be coherently updated. In this way, objects in different simulation layers are able to communicate with each other.

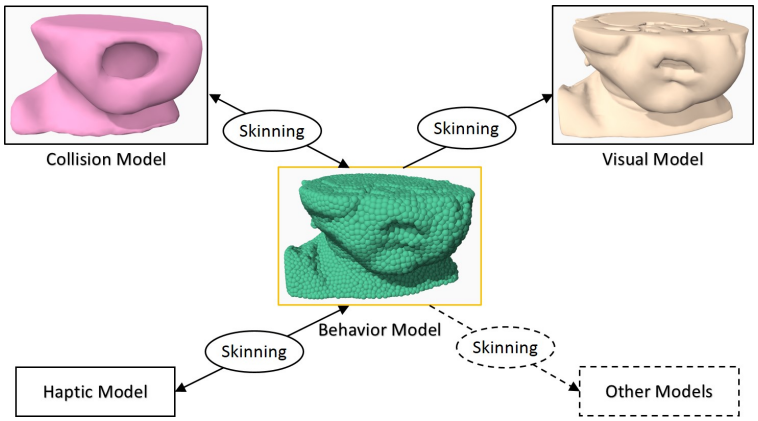

Figure 3.3: Illustration of the multi-model representation for a patient model. Notice how the visual model is more detailed than the collision model and how the behavior model relies on a very different representation. The behavior model controls the update of the other representations through skinning.

3.5 Sensors

The simulator interfaces with many different interaction devices through the sensor class. A sensor is an abstraction of a 6 degree-of-freedom (DOF) tracking and/or haptic device. It contains a position in 3D space, along with orientation information (rotation matrix) and includes an array of activation values to store information from any buttons associated with the device. The activation values can be used to specify hinge angles (e.g., to control how opened or closed the scissors are) or telescoping/plunger depth (for a syringe or resectoscope handle) for the object's subparts.

Medical instruments are simulated as kinematic objects in the virtual scene, whose poses are driven by the tracking sensors. We have created various instruments for different medical procedures and have functionally classified them into a number of different categories, including single pieced (e.g., laryngoscope, scalpel, cautery hook, needle tip cautery, probe) and hinged (forceps, endoscopic scissors, graspers) tools, based on the measurements of real instruments. New instruments can be added as new procedures are simulated.

Each sensor also contains a transformation matrix that is used by the calibration module to register each sensor to the world space. In our system, a sensor can be linked to a particular

object (typically a virtual instrument) through either direct coupling or virtual coupling with a 6-DOF damped spring. When the sensor pose is updated, the virtual instrument is transformed by the sensor data. HMDs are also supported in our system. By attaching the viewing location to the HMD, the virtual environment can be rendered from the user's viewpoint, which allows the user to perform the procedure from any viewpoint they want just as in real situation instead of a fixed view as in screen-based computer simulators. Moreover, by attaching the viewing location to a sensor linked to a virtual instrument, we can easily obtain an endoscopic view from any given instrument.

3.6 Simulation Core

The simulation core contains modeling tools and processes the dynamics of each object (deformable, rigid-body, or fluids) at every simulation timestep. Our framework includes simulation of all human body constituents, such as bones, soft tissues and fluids (e.g., blood, secretions) using PBD [54]. In previous simulation frameworks, visual effects are made using a combination of elements created using specialized solvers for rigid bodies, fluids, clothing, etc. However, in our framework, all object types are simulated using a unified particle representation based on PBD, which enables new effects where different simulated substances can interact with each other seamlessly. This allows for efficient modeling of large soft tissues with many different material properties in real time. PBD's strength lies in enabling visually plausible effects that enhance the visual experience. It is not designed to build medical simulation affecting physics–for example, it lacks functionality such as contact callbacks, scene queries (ray-casting, overlap test, etc.), joints, serialization, etc. For this reason, we also integrated a rigid-body dynamics model into our framework [58]. Benefiting from our multi-model representation, objects in PBD are able to communicate with objects in the rigid-body dynamics.

In PBD, each object is comprised of arrays of N particles (each consisting of positions, velocities and masses) and M constraints that describe the object. Particles are generated by

performing uniform voxelization of the surface mesh of the object and placing particles at each occupied voxel interior to the mesh. To simulate surgical cutting, a tetrahedral mesh of the object is generated, and particles are placed at vertices of each tetrahedron. Deformations are simulated by adding shape-matching constraints [55], stretching constraints [38], bending constraints [45], torsion constraints [45], or density constraints [47] (for simulation of fluids) within each object.

3.7 Collision Detection and Resolution

In PBD, collisions are solved by adding additional collision constraints to the particles. One advantage of the position-based approach is how simply collision detection and resolution can be realized since all dynamics in one scene are represented as particles with a single collision radius. Two types of collisions are handled in PBD: particle-to-particle collision and particle-to-mesh collision. Meshes are used to represent all kinematic/static objects, including plane, sphere, box, capsule, convex mesh, triangle mesh and SDF. At each time step, a set of M_{coll} collision constraints are generated from scratch. While the first set of M constraints given by the object representation are fixed throughout the simulation, the number of collision constraints M_{coll} varies and depends on the number of colliding particles.

The virtual model of each instrument is represented as kinematic object and approximated with simple collision shapes, such as capsules or convex meshes, for fast collision detection against particles. We enclose each instrument within one or multiple collision shapes. The sensor device provides the input position and orientation for the graphical model of the instrument and its collision shapes, which allows the user to interact with various instruments in the virtual scene.

Various tool-tissue interactions can be simulated easily within the PBD simulation layer. A probing interaction induces an instantaneous displacement of the particles to resolve the collision. A grasping interaction attracts the particles on the object that are in contact with

the instrument tip with the rest of the particles deformed according to the constraints between them. Cutting interaction is implemented by dividing the corresponding tetrahedrons at the location of contact and inserting new particles and edges. A cauterizing interaction is triggered when the instrument is in contact with the tissue, and color of the mesh faces in contact is changed progressively. Blended textures can also be used to achieve desired graphical results (e.g., smoke). In each of the above cases, the collision detection and resolution algorithm processes each of the collision pair list, performs their interaction function, and the contact information is sent to the haptic thread to compute the resulting haptic force of that interaction upon the instrument and to the automated visualization component for the computation of various visual feedback tools.

Since the instrument under interaction is kinematic and free from physics in the virtual world, external forces from gravity or collisions will no longer have any effect on it because the position is set absolutely by the user. The held instrument could still apply forces to dynamic objects that are free to move, but if it pushes a dynamic object against a static object (such as the tissue attached to the bones), the held instrument would easily penetrate the tissue. Moreover, there is no interaction or collision between the held instrument and static objects, such as the bones. Therefore, we need the rigid-body dynamics model as an additional simulation layer in parallel to the PBD layer to account for the collisions for rigid objects.

In the rigid-body simulation layer, we create the same collision shape representations for the instrument as that in the PBD layer. We also create the collsion shapes for those static objects (such as the bones) in the rigid-body dynamics layer. Like the PBD layer, the kinematic instrument in the rigid-body simulation layer has no interaction or collision with static objects. Therefore, we employed a virtual coupling scheme by creating a dynamic proxy instrument and constraining it to the kinematic instrument. While the kinematic instrument can penetrate a collision shape, the proxy instrument will respond to collisions and remain on the outside of the volume of the shape. In each simulation cycle, the rigid-

body dynamics layer fetches the cluster positions of objects from the PBD layer to update its collision shapes through skinning. Conversely, the PBD layer receives the last non-colliding position of the instrument from the rigid-body dynamics layer to update the poses of the collision shapes of the tool. The implementation details of collision handling will be introduced in Chapter 5 with the ETI simulation.

3.8 Haptic Rendering

In order to create a realistic VR simulation, visual feedback is extremely important. Equally important in bringing a real immersive experience is to integrate haptics into the simulation. Just as a computer display requires algorithms and rendering techniques to deliver visual realism, haptic devices also require the development of specialized algorithms to deliver the same levels of realism to our sense of touch. Haptics is the use of force to recreate a touch sensation. It can be used to give a user the perception that they are interacting with real objects that are in fact virtual. A haptic device is one that can receive inputs from a user as to how they would like to interact with a virtual environment and can provide a force to the user as an output. The simulated force applied to the haptic device must be updated at a high rate of about 1kHz to maintain a realistic haptic feedback. However, the computational process of the collision detection, deformable models and the complex physical simulation are so time consuming that the required update rate of haptic force cannot be guaranteed. Therefore, we implemented a multi-thread haptic rendering process to separate the haptic thread from the physical thread. Our haptic rendering algorithm utilizes different mechanisms for computing the force due to contact with rigid bodies (Figure 3.4) and with soft tissues (Figure 3.5).

3.8.1 Force Feedback with Rigid Body

In order to enhance the stability of the haptic system, we employed the virtual coupling scheme instead of the direct coupling scheme for force calculation with rigid bodies in

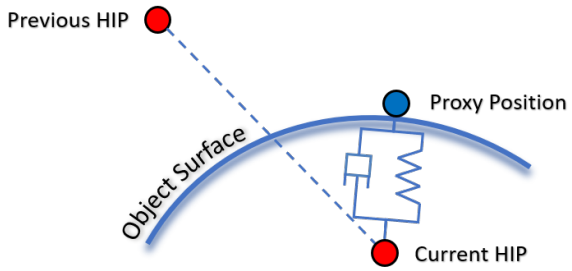

Figure 3.4: Calculation of rigid body force feedback.

the scene. The general structure of virtual coupling is shown in Figure 3.4. The virtual instrument is attached to the haptic interface through a virtual spring and virtual damper in mechanical parallels. The haptic device controls the position and orientation of its virtual counterpart. The spring's displacement generate forces on the dynamic virtual instrument and an opposite force on the haptic device. Mass properties are assigned to the virtual instrument which can be felt through the user's hand. The force is computed as follows:

$$\mathbf{f}_{rigid} = k_s(\mathbf{p}_{proxy} - \mathbf{p}_{HIP}) - k_d(\mathbf{v}_{proxy} - \mathbf{v}_{HIP}), \tag{3.1}$$

where k_s and k_d are spring stiffness and damping stiffness, $\mathbf{p}_{proxy}$ and $\mathbf{p}_{HIP}$ are the positions of the virtual instrument and the haptic device. $\mathbf{v}_{proxy}$ and $\mathbf{v}_{HIP}$ are the velocities of the virtual instrument and the haptic device.

3.8.2 Force Feedback with Soft Body

For the force feedback from the interaction with soft tissues, collision tests between the virtual instrument and the particles are computed in parallel in the haptic thread. We calculate the force as the average of all vectors pointing from the current positions of the particles $\mathbf{p}_i$ (in contact with the virtual instrument) to their rest positions $\mathbf{p}_i^0$ (Figure 3.5), which are backed up at the initialization of the simulation. We calculate an average vector for each type of tissue that is in contact with the virtual instrument and multiplied each vector by a

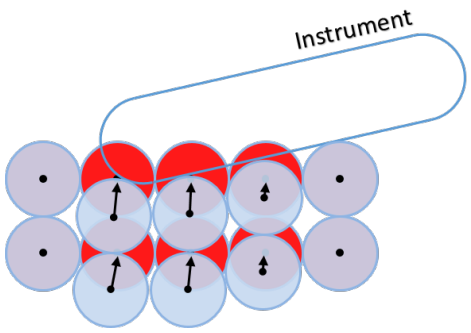

Figure 3.5: Calculation of soft tissue force feedback.

constant factor μ (stiffness of each tissue type):

$$\mathbf{f}_{soft} = \mu \frac{\sum_i (\mathbf{p}_i^0 - \mathbf{p}_i)}{N}, \tag{3.2}$$

We then average all those vectors from each type of tissue to generate the force feedback from soft tissues.

Finally, the summation of the interaction forces due to the contact with the rigid bodies and soft bodies is sent to the haptic device to calculate the overall force vector that will be realized upon the virtual instrument.

3.9 Automated Assessment

Traditionally, performance assessments require experts to review and rate individual examinee performance on patients or simulators, which is subjective, highly variable and resource-intensive. Moreover, current simulators do not provide complete 3D information concerning the relationship of the instrument with the simulated patient or quantitative feedback regarding the quality of an attempt. As a results, learners have poor knowledge about what went wrong and how to improve. And poor visualization during the procedure prevents instructors from observing the events that are occluded from instrutor's viewpoint, which introduces an additional source of error into the assessment.

These limitations have driven efforts to develop automated assessment systems for

the quality of performance, which is essential in acquiring precise data, setting objective standards, and quantifying the performance. There have been some attempts at capturing the motions of the instruments and limited fiducial points on manikins/patients and/or measuring the forces imparted on specific parts of the manikins/patients using various tracking or force sensors [11][15][16][26][64][85]. Although these efforts provide some additional information about trainees' performance, they do not provide the complete picture of the interaction of the instruments within the manikin. Moreover, these works only included preliminary analysis on differentiating subjects with different skill levels, but no automated assessment method was developed for skill classification or score prediction on subjects' performances. Capturing the complete motion is difficult without a virtual representation of all the components. Various computer-based simulators have been developed for training medical professionals as reviewed in Chapter 2. However, since the virtual models are not physically simulated, and collisions between different object types are not implemented, their systems are not capable of capturing the entire motions of the instruments with respect to the internal geometric change of the virtual model. Therefore, they suffer from the same issues as the above manikin-based simulators by not providing a complete picture of the entire procedure.

With respect to automated skill assessment, a few of works have been using hidden Markov model [79][91] or convolutional neural network [84] to classify surgical skills and/or predict performance score of a surgery. However, all these machine learning methods lack interpretable result, which can be given as feedback that is understandable and intuitive in helping trainees reach higher skill levels. In order to make the assessment model more interpretable, some works have been using fully convolutional neural networks [22][23], regression models [20][91], or multilevel models [33] to predict the overall score of surgical performances.

In contrast, our simulation framework is capable of simulating all the components involved in a procedure in a realistic manner and therefore, is able to capture the entire 3D

motions of instruments in relation to the internal geometric change of the virtual patient. Our automated visualization tools provide a complete visualization of the procedure, offering trainees and instructors with comprehensive information for real-time guidance and post-trial assessment. By using the performance parameters extracted from the motions and the scores rated by experts, a multinomial regression model is developed to automatically assess the trainees' performances.

3.9.1 Automated Visualization

With the virtual representations of all the components involved in a procedure, we can visualize an even larger set of performance parameters in relation to the geometric change of the virtual model than those physical simulators during the entire procedure. Our system captures the entire motions in 3D allowing for real-time and post-trial viewing, visualization from different angles including cross-sectional and/or laryngoscopic/endoscopic views and measurements on the virtual model to provide detailed visual feedback and analysis of multiple aspects of a procedure. Our visualization tools not only provide trainees with real-time feedback throughout the entire procedure, but also provide instructors with a clear understanding of what is happening during each step of a procedure, which is impossible in a physical simulator.

From the raw motion sequences, the simulation system can extract various performance parameters (such as the 3D trajectory of the tip of the instrument, color-coded force applied on the virtual patient model, line of sight with the name of the anatomical structure that the subject is looking at, etc.) that are chosen *a priori* based on the qualities that the expert instructor deems as important features for a certain procedure. All these visualization features can be implemented based on the contact information and scene query (raycasting, overlap, etc.) results returned from the rigid-body dynamics simulation layer and can be rendered on the final visual models as additional information to guide the trainee throughout the entire procedure and offer instructors with comprehensive visualization for accurate

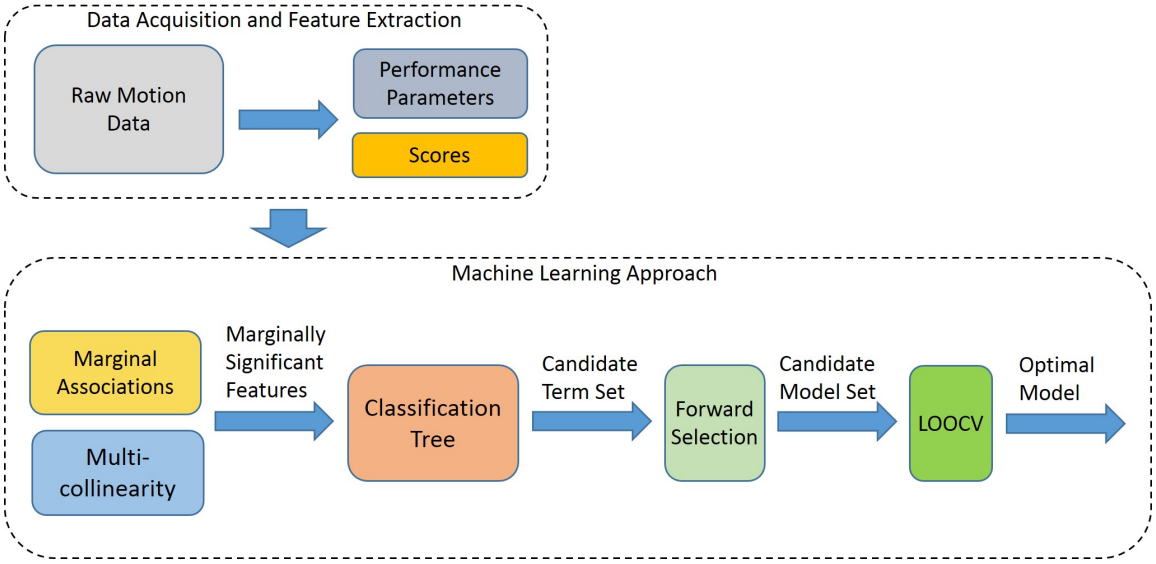

Figure 3.6: Pipeline of our proposed automated assessment algorithm.

assessment.

3.9.2 Automated Scoring

In addition, all the extracted performance parameters can also be fed into our automated scoring system, which automatically assess trainee's performances using an interpretable machine learning algorithm. These performance parameters help further characterize subjects' performances and reflect the qualities of motion that are subtle and difficult to quantify by merely looking at the motion playback. The choice of parameters selected will affect the automated scoring system. A multinomial regression model is developed by using the extracted performance parameters as input features and the scores assigned by the rater as input labels. Our automated scoring model offers interpretable results and gives a consistent and complete evaluation of trainee's performance, which would obviate the need to have an instructor present to rate each trial and would be more objective and less variable. The process of finding the optimal model is illustrated in Figure 5.2.

With all the performance parameters (features), we first select those ones which have statistically significant marginal associations with the score on a n-point scale. We fit

multinomial regression models for each feature with the overall score as follows:

$$\log\left(\frac{P(\text{Score} \leq j)}{1-P(\text{Score} \leq j)}\right) = \beta_{0j} + \beta_1 F, \tag{3.3}$$

where β_{0j}, $j = 1, 2, ..., n-1$ are the category-specific intercepts and β_1 is the coefficient of feature F. Each model is fitted by using the multinomial generalized estimating equation method (GEE) to account for the repeated measures [82]. A feature is considered marginally significant if the multinomial regression model against this feature has a P-value ≤ 0.05. In addition, for those marginally significant features, if any two of them are highly correlated (e.g., correlations > 0.8), we exclude one of them from the subsequent steps to prevent collinearity.

As it is difficult to build a robust regression model for a complex scoring rubric with only linear terms, we evaluate the discontinuous breakpoints of the selected features from the previous step to extract both piecewise linear terms and interaction terms as part of the candidate set. Thus, we construct a classification tree for the overall score with the selected features [80]. The classification tree can hierarchically identify predictive features and their discontinuous breakpoints. We use the Gini index as the splitting metric which is a measure of node purity. The optimal breakpoints are those with maximal impurity reduction in each level. The closer to the root, the more predictive the feature is. We consider these features within four layers from the root of the tree as the most predictive ones. The correlation between multiple trials performed by the same subject are ignored when constructing the classification tree. This drawback will be addressed in the next step.

The candidate model set is determined by the forward selection algorithm (see Algorithm 1) using the candidate term set acquired from above, which contains three categories of terms: linear terms (the marginally significant features which did not appear in the classification tree), piecewise linear terms (the predictive features in the classification tree), and their two-way interaction terms. We first define a candidate model set S for choosing the optimal model in the algorithm and initialize it as empty. We then define a reference model r for the

Algorithm 1 Forward selection

Initialize the candidate model set as $S = \emptyset$ and enumerate all the m terms in the candiate term set
Initialize the reference model $r = 1$
loop
    for each $i = 0, 1, ..., m-1$ **do**
        $c_i = r$+one additional term
        p_i =WaldTest(c_i, r)
    end for
    Consider all possible c_i models and determine the one c' that has the minimum P-value p' of the Wald test
    if $p' > 0.05$; or numerical errors occur **then**
        break;
    end if
    $S = S \cup c'$
    $r = c'$
end loop
for each $s_i \in S$ **do**
    Compute the classification accuracy of each model with LOOCV
end for
Return the optimal model with the highest classification accuracy

forward selection in each iteration. It is initialized to a null model that only contains the intercept term. In each iteration, we add one additional term to the reference model r as the current model c_i and compared the P-values by the Wald test between r and all possible c_i models. Each model is fitted by the multinomial GEE method to account for the repeated measures. We select the model c' with the minimum P-value in the Wald test as the new reference model: $r = c'$. We then update the candidate model set: $S = S \cup r$. We repeat the above steps until the minimum P-value at one iteration is > 0.05 or numerical errors occur from Wald test.

Lastly, with the candidate model set S, we find the optimal model that has the highest classification accuracy by performing leave-one-out-cross-validation (LOOCV). Due to the nature of cross-validation and repeated trials from each subject, we take one subject out as the test data for cross-validation [66].

Chapter 4: Position-based Simulation

4.1 Background

The simulation of deformable objects has been an active research topic in computer graphics over the past decades. Traditional force-based dynamics, including FEM, mass-spring systems, meshless methods and particle systems, have been widely used. A few comprehensive reviews of such approaches can be found in [5][56]. These force-based approaches model deformable objects through the manipulation of internal and external forces, which are transformed into accelerations, using Newton's second law of motion. The positions of elements that comprise an object are then determined through numerical integration of the derived accelerations, which usually cause overshooting under large time steps (Figure 4.1). These force-based approaches can provide an accurate computation model for elastic objects of different material properties but require high computational cost. Therefore, simulating the entire medical procedure using force-based methods is not practical.

PBD [54] has gained great popularity recently due to its simplicity, high efficiency, unconditional stability and real-time performance. In contrast to force-based methods that achieve equilibrium configurations through the integration of accelerations, PBD approach directly projects positions as a solution to a set of geometrical constraints, which eliminates the overshooting problem in force-based methods and simplifies the implementation process (Figure 4.1). Although PBD is not as accurate as force-based methods, its efficiency and controllability far outperform those methods in simulating medical procedures while providing visually plausible results [10]. The PBD approach has already been successfully applied in the medical field. Kubiak et al. [45] developed a real-time surgical thread simulator for an interactive and robust simulation of knot tying. Qian et al. [63] developed a laparoscopic surgery simulation that can simulate complex anatomic structures composed of soft tissues with different properties. Berndt et al. [7] implemented an efficient surgical

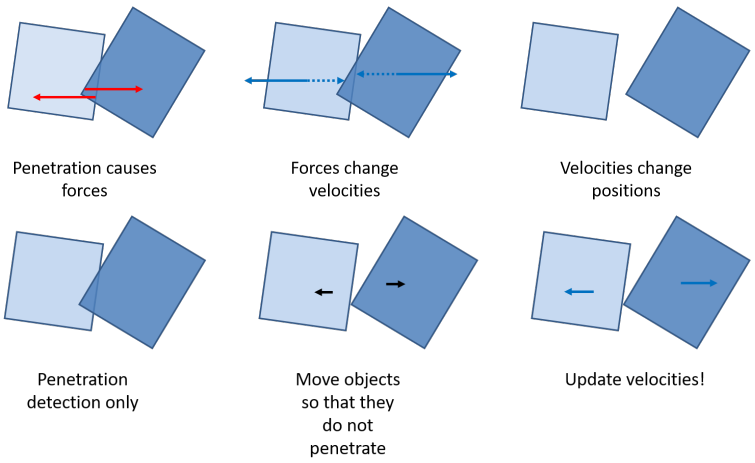

Figure 4.1: Comparison of force based update and position-based update.

cutting method for soft tissues of human body. Heredia-Pérez et al. [37] developed a robotic surgery simulation for resection of pituitary tumours in the brain. However, all those simulators only use the PBD to simulate deformable objects, such as organs and surgical threads, there is no modeling of heterogeneous scenes, such as rigids and fluids, and their interactions. In contrast, our framework integrates the modeling of all the heterogeneous object types using PBD for medical simulation. In this chapter, we will introduce the alogorithm of PBD and some common constraint examples to model rigid, soft, fluid, and cloth objects in a unified particle representation . The implementation details of integrating the simulation of heterogeneous objects types will be described in Chapter 5.

4.2 Particle Representation

The objects to be simulated are represented by a set of N particles and M constraints that describe the objects. Each particle $i \in [1,...,N]$ has three attributes, namely position $\mathbf{x}_i$, velocity $\mathbf{v}_i$ and mass m_i. Aside from these attributes, each particle has a phase identifier that organizes particles into groups and controls how they behave and collide with other groups of particles. Each object is represented by a group of particles that have the same phase value, which acts as a collision filter used for collision detection against other groups of

particles. Each constraint $j \in [1,...,M]$ is defined by the following five attributes:

- a cardinality n_j;
- a function C_j: $\mathbb{R}^{3n_j} \rightarrow \mathbb{R}$;
- set of indices $i_1,...,i_{n_j}, i_k \in [1,...,N]$;
- a stiffness parameter $k_j \in [0,...,1]$ and
- a type of either equality or inequality.

Constraint j with type equality is satisfied if $C_j(\mathbf{x}_{i_1},...,\mathbf{x}_{i_{n_j}}) = 0$. Otherwise its type is inequality if $C_j(\mathbf{x}_{i_1},...,\mathbf{x}_{i_{n_j}}) \geq 0$. The stiffness parameter k_j defines the strength of the constraint.

The algorithm of PBD simulation is described in Algorithm 2. Given the above data and a time step Δt, the positions and the velocities of the particles are firstly specified in lines (1)-(3) before the simulation loop starts. Lines (5)-(6) perform a simple explicit forward Euler integration step on the velocities and the positions. The new locations $\mathbf{p}_i$ are not assigned to the positions $\mathbf{x}_i$ directly but are only used as predictions. Non-permanent external constraints such as collision constraints are generated at each time step from scratch in line (7) by using both the original and the predicted positions in order to perform continuous collision detection. Lines (8)-(10) then iteratively corrects the predicted positions such that they satisfy the M internal constraints and the M_{coll} external constraints. Finally, the corrected positions $\mathbf{p}_i$ are used to update the positions and the velocities of the particles.

4.3 Constraint Projection

The goal of the solver is to correct the predicted positions $\mathbf{p}_i$ of the particles such that they satisfy all the constraints (Lines (8)-(10)). The most important issue in connection with moving points directly inside a simulation loop is the conservation of linear and angular

Algorithm 2 Algorithm of PBD

for all vertices i **do**
    initialize $\mathbf{x}_i = \mathbf{x}_i^0, \mathbf{v}_i = \mathbf{v}_i^0, w_i = 1 = m_i$
end for
loop
    for all vertices i **do** $\mathbf{v}_i \leftarrow \mathbf{v}_i + \Delta t w_i \mathbf{f}_{ext}(\mathbf{x}_i)$
    end for
    for all vertices i **do** $\mathbf{p}_i \leftarrow \mathbf{x}_i + \Delta t \mathbf{v}_i$
    end for
    for all vertices i **do** generateCollisionConstraints($\mathbf{x}_i \rightarrow \mathbf{p}_i$)
    end for
    loop solverIterations **times**
        projectConstraints($C_1, ..., C_{M+M_{coll}}, \mathbf{p}_1, ..., \mathbf{p}_N$)
    end loop
    for all vertices i **do**
        $\mathbf{v}_i \leftarrow (\mathbf{p}_i - \mathbf{x}_i)/\Delta t$
        $\mathbf{x}_i \leftarrow \mathbf{p}_i$
    end for
    velocityUpdate($\mathbf{v}_1, ..., \mathbf{v}_N$)
end loop

momenta. Let $\Delta\mathbf{p}_i$ be the displacement of vertex i by the projection. Linear momentum is conserved if

$$\sum_i m_i \Delta \mathbf{p}_i = 0, \tag{4.1}$$

which amounts to conserving the center of mass. Angular momentum is conserved if

$$\sum_i \mathbf{r}_i \times m_i \Delta \mathbf{p}_i = 0, \tag{4.2}$$

where the $\mathbf{r}_i$ are the distance vectors of the $\mathbf{p}_i$ to an arbitrary common rotation center. If a projection violates one of these constraints it introduces so called ghost forces which act like external forces dragging and rotating the object. However, only internal constraints need to conserve the momenta. Collision or attachment constraints are allowed to have global effects on the object.

Let $\mathbf{p}$ be the concatenation $[\mathbf{p}_{i_1}^T, ..., \mathbf{p}_{i_{n_j}}^T]^T$ and let each constraint function C_j, $j \in [1, \ldots, M]$ take the concatenated vector $\mathbf{p}$ as input. For internal constraints, C is inde-

pendent of rigid body modes, i.e. translation and rotation. This means that rotating or translating the points does not change the value of the constraint function. Therefore, the gradient $\nabla_{\mathbf{p}} C$ is perpendicular to rigid body modes because it is the direction of maximal change. If the correction Δp is chosen to be along $\nabla_{\mathbf{p}} C$ both momenta are automatically conserved. PBD solves a system of non-linear equality and inequality constraints such that

$$C_1(\mathbf{p}) \succ 0$$

$$\ldots$$

$$C_j(\mathbf{p}) \succ 0,$$

where the symbol $\succ$ denotes either $=$ or $\geq$. Constraints are solved using the Gauss-Seidel method. For each iteration, constraints are solved sequentially through a linearization in the neighborhood of the current solution $C(\mathbf{p})$ using first-order Taylor polynomial:

$$C(\mathbf{p}+\Delta\mathbf{p}) \approx C(\mathbf{p}) + \nabla_{\mathbf{p}} C(\mathbf{p}) \cdot \Delta\mathbf{p} = 0. \tag{4.3}$$

The position displacement $\Delta\mathbf{p}$ is restricted to be in the direction of $\nabla_{\mathbf{p}} C$, which conserves the linear and angular momenta. This means that only one scalar λ - a Lagrange multiplier - has to be found such that the correction

$$\Delta\mathbf{p} = \lambda \nabla_{\mathbf{p}} C(\mathbf{p}) \tag{4.4}$$

solves Equation 4.3. This yields the following formula for $\Delta\mathbf{p}$:

$$\Delta\mathbf{p} = -\frac{C(\mathbf{p})}{|\nabla_{\mathbf{p}} C(\mathbf{p})|^2} \nabla_{\mathbf{p}} C(\mathbf{p}). \tag{4.5}$$

For the correction of a single particle $\mathbf{p}_i$, we have

$$\Delta\mathbf{p}_i = -s w_i \nabla_{\mathbf{p}_i} C(\mathbf{p}), \tag{4.6}$$

where the scaling factor

$$s = \frac{C(\mathbf{p})}{\sum_j w_j |\nabla_{\mathbf{p}} C(\mathbf{p})|^2} \tag{4.7}$$

is the same for all particles and weight $w_i = 1/m_i$. In this case a particle with infinite mass, e.g., $w_i = 0$, will not be moved. After each constraint has been solved, positions are updated. After a specified number of iterations, the change in velocity is determined by the total constraint displacement

$$\Delta \mathbf{v} = \frac{\Delta \mathbf{p}}{\Delta t}, \tag{4.8}$$

Inequality constraints are handled trivially by first checking whether $C(\mathbf{p}) \geq 0$. If this is the case, the constraint is simply skipped.

4.4 The Non-Linear Gauss–Seidel Solver

The input to the solver are the M and M_{coll} constraints, and the estimates $\mathbf{p}_1, ..., \mathbf{p}_N$ for the prediction locations of the particles. The solver tries to modify the estimates such that they satisfy all the constraints. The resulting system of equations is non-linear. In addition, the constraints of type inequality yield inequalities. The non-linear Gauss-Seidel (GS) method is used to solve such a general set of equations and inequalities. The original GS algorithm can only handle linear system. However, solving a constraint in PBD is a non-linear operation. We repeatedly iterate through all the constraints and project the particles to valid locations with respect to the given constraint alone. In contrast to a Jacobi-type iteration, modifications to point locations immediately become visible to the process, which speeds up convergence significantly. In over-constrained situations, the process can lead to oscillations if the order is not kept constant.

4.5 Specific Constraints

In the following, we will introduce some common constraint examples used in medical simulations to model rigid, soft, fluid, or cloth-type objects.

4.5.1 Spring Constraints

Spring constraints are specified as pairs of particles with a rest-length spring and a user-specified stiffness coefficient that defines the strength of the constraint in a range from 0.0 to 1.0 (Figure 4.2). They are not a spring in the classical sense, but rather a distance constraint:

$$C(\mathbf{p}_1, \mathbf{p}_2) = |\mathbf{p}_1 - \mathbf{p}_2| - d, \tag{4.9}$$

where d is the rest length between the two particles. The derivatives with respect to the points are:

$$\nabla_{\mathbf{p}_1} C(\mathbf{p}_1 - \mathbf{p}_2) = \mathbf{n}, \tag{4.10}$$

$$\nabla_{\mathbf{p}_2} C(\mathbf{p}_1 - \mathbf{p}_2) = -\mathbf{n}, \tag{4.11}$$

with

$$\mathbf{n} = \frac{\mathbf{p}_1 - \mathbf{p}_2}{|\mathbf{p}_1 - \mathbf{p}_2|}. \tag{4.12}$$

The scaling factor s is:

$$s = \frac{|\mathbf{p}_1 - \mathbf{p}_2| - d}{w_1 + w_2} \tag{4.13}$$

and the final corrections are:

$$\Delta\mathbf{p}_1 = -\frac{w_1}{w_1 + w_2}(|\mathbf{p}_1 - \mathbf{p}_2| - d)\frac{\mathbf{p}_1 - \mathbf{p}_2}{|\mathbf{p}_1 - \mathbf{p}_2|} \tag{4.14}$$

$$\Delta\mathbf{p}_2 = \frac{w_2}{w_1 + w_2}(|\mathbf{p}_1 - \mathbf{p}_2| - d)\frac{\mathbf{p}_1 - \mathbf{p}_2}{|\mathbf{p}_1 - \mathbf{p}_2|} \tag{4.15}$$

The spring constraints can be used to simulate soft tissues if the object is represented as tetrahedron mesh. Springs can be placed at all the edges of the mesh. It can also be used as external constraints by attaching one object to another, for example, to attach the tongue to the mouth, or to attach the soft tissue to the bone.

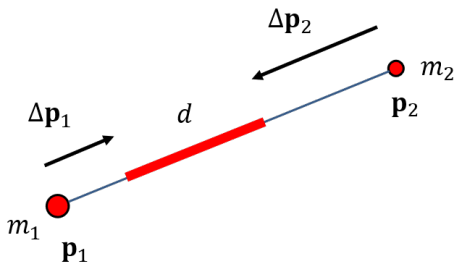

Figure 4.2: Spring constraint.

4.5.2 Bending Constraints

In membrane or cloth simulation, it is important to simulate bending in addition to stretching resistance (spring constraint). As shown in Figure 4.3, for each pair of adjacent triangles $(\mathbf{p}_1, \mathbf{p}_3, \mathbf{p}_2)$ and $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)$ a bilateral bending constraint is added with constraint function:

$$C_{bend}(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4) = acos\left(\frac{(\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_3 - \mathbf{p}_1)}{|(\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_3 - \mathbf{p}_1)|} \cdot \frac{(\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_4 - \mathbf{p}_1)}{|(\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_4 - \mathbf{p}_1)|}\right) - \phi_0 \tag{4.16}$$

and a user-specified stiffness coefficient which defines the bending stiffness of the constraint. The scalar ϕ_0 is the initial dihedral angle between the two triangles. The advantage of this bending term over adding a spring constraint between points $\mathbf{p}_3$ and $\mathbf{p}_4$ is that it is independent of stretching. This is because the term is independent of edge lengths.

4.5.3 Shape Matching Constraints

In PBD, deformable objects can be simulated by adding geometrically motivated shape-matching constraints [55]. This method is based on finding the least square optimal rigid transformation in 3D between two sets of points with *a priori* known correspondence. The algorithm requires a set of particles with masses m_i and their respective initial positions $\mathbf{p}_i^0$ as input. At each time step, the original shape $\mathbf{p}_i^0$ is matched to the deformed shape $\mathbf{p}_i$. Then, the deformed points $\mathbf{p}_i$ are forced towards the goal positions $\mathbf{g}_i$. Given two sets of points $\mathbf{p}_i^0$ and $\mathbf{p}_i$, the minimization problem is given by

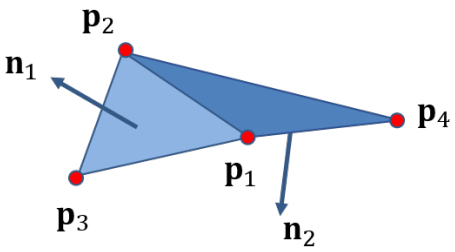

Figure 4.3: Bending constraint.

$$\sum_i m_i(\mathbf{R}(\mathbf{p}_i^0 - \mathbf{t}_0) + \mathbf{t} - \mathbf{p}_i)^2, \tag{4.17}$$

where m_i are the masses of individual particles, $\mathbf{R}$ is the rotation matrix, $\mathbf{t}$ and $\mathbf{t}_0$ are the translation vectors given by the center of mass of the initial shape $\bar{\mathbf{p}}^0$ and the deformed shape $\bar{\mathbf{p}}$ respectively. Once the optimal rotation $\mathbf{R}$ and translation vector $\mathbf{t}$ are derived, the goal positions can be computed as

$$\mathbf{g}_i = \mathbf{R}(\mathbf{p}_i^0 - \bar{\mathbf{p}}^0) + \bar{\mathbf{p}}. \tag{4.18}$$

From the goal positions, an integration scheme can be defined

$$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t) + \alpha \frac{\mathbf{g}_i(t) - \mathbf{p}_i(t)}{\Delta t} + \frac{\Delta t \mathbf{f}_{ext}(t)}{m_i}, \tag{4.19}$$

$$\mathbf{p}_i(t + \Delta t) = \mathbf{p}_i(t) + \Delta t \mathbf{v}_i(t + \Delta t), \tag{4.20}$$

where $\alpha \in [0, 1]$ is a user defined stiffness parameter which defines how far the particles are pulled toward their goal positions.

4.5.4 Cluster-based Shape Matching Constraints

The implementation of the algorithm described above allows only for small deformations from the initial shape. For larger deformations, e.g., to model soft tissue, the concept of cluster-based shape matching method [55][19][68] can be integrated. The idea is to

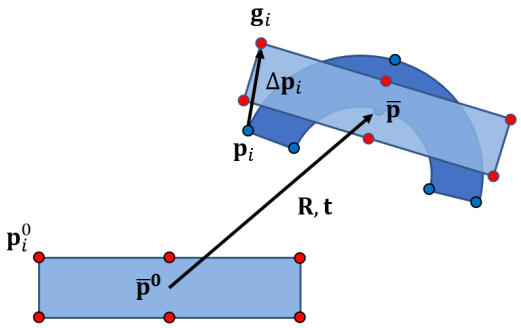

Figure 4.4: Shape matching constraint. The initial shape with the vertex positions xi is matched to the deformed configuration xi to obtain goal positions gi. The deformed shape is pulled towards these goal positions to simulate elastic behavior.

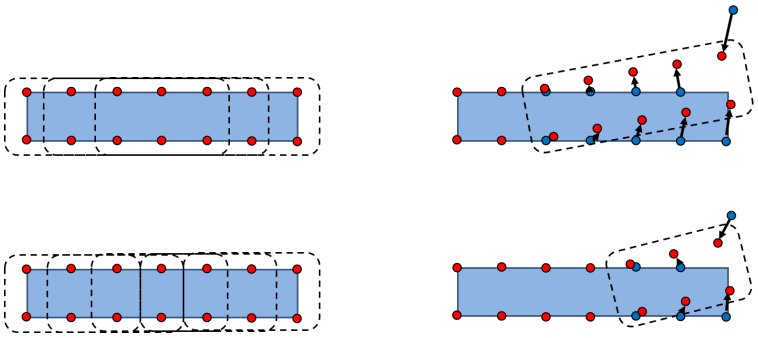

Figure 4.5: Cluster-based shape matching constraint. The stiffness of the model depends on the region size. Smaller regions (top) allow larger deformations than larger regions (bottom). The hexagons in the left images represent the overlapping regions of the model. The right images show the goal positions after one particle is displaced away.

divide the set of particles that comprise an object into multiple overlapping clusters (shape-matching constraints). Every cluster has its own position and rotation, which serve as "joint" for the object. Every particle is connected to one or more clusters with a weight between 0 and 1, which defines how much a cluster influences the transform of a particle. After performing shape matching for all clusters, we get multiple goal positions for each particle. The final goal position for a particle is determined by blending the goal positions of the n_i corresponding clusters $c_i \in C$

$$\mathbf{g}_i = \frac{\sum_{c_i \in C} \mathbf{R}_{c_i}(\mathbf{p}_i^0 - \bar{\mathbf{p}}_{c_i}^0) + \bar{\mathbf{p}}_{c_i}}{|n_i|}. \tag{4.21}$$

4.5.5 Density Constraints

It is also possible to simulate fluids in the PBD framework even though it has been widely used for the simulation of deformable objects. Fluids can be simulated by adding some specialized constraints. A straightforward approach would be to model the fluid as a system of particles constrained to maintain a minimum distance from each other, however this leads to granular-like behaviour and will typically fail to reach hydrostatic equilibrium when coming to rest. An alternative method is presented by Macklin et al. [47]. This method uses position-based density constraints to achieve lower compression and better stability than traditional Smoothed Particle Hydrodynamics (SPH) methods [50]. To enforce constant density, a constraint function

$$C_i(\mathbf{p}) = \frac{\rho_i}{\rho_0} - 1 \leq 0 \tag{4.22}$$

is added to each fluid particle i. Each constraint C_i is a function of the particle's position and the positions of its neighbors. ρ_0 is the rest density and ρ_i is given by the standard SPH density estimator:

$$\rho_i = \sum_j m_j W(\mathbf{p}_i - \mathbf{p}_j, h), \tag{4.23}$$

where m_j are the masses of neighboring particles within the smoothing length radius of particle i, h is the smoothing length and W is the kernel function.

In order to solve these density constraints using PBD, the derivative of the constraint function Equation 4.22 with respect to each particle's position is required. This can be calculated using the gradient of SPH kernels

$$\nabla_{\mathbf{p}_k} C_i = \frac{1}{\rho_0} \begin{cases} \sum_j \nabla_{\mathbf{p}_k} W(\mathbf{p}_i - \mathbf{p}_j, h) \text{ if } k = i \\ -\nabla_{\mathbf{p}_k} W(\mathbf{p}_i - \mathbf{p}_j, h) \text{ if } k = j \end{cases} \tag{4.24}$$

Then, by taking advantage of symmetry in the SPH smoothing kernel W, the corrected position due to the particle's own density constraint, and the density constraints of its

neighbours is given by

$$\Delta \mathbf{p}_i = \frac{1}{\rho_0} \sum_j (\lambda_i + \lambda_j) \nabla W(\mathbf{p}_i - \mathbf{p}_j, h), \tag{4.25}$$

where λ is the per-constraint scaling factor.

Chapter 5: Endotracheal Intubation Simulation

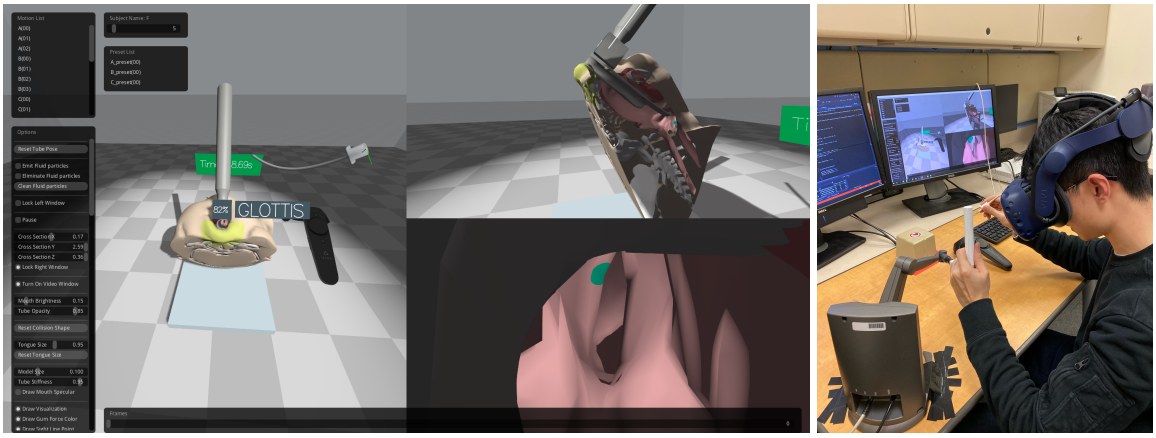

Figure 5.1: (left) Screenshot of the simulation with left-eye view from the HMD, cross-sectional view and video laryngoscopic view. Enhanced visualization tools, such as the color-coded force on the upper gums, percentage of glottis opening and laser dot of sight line, are shown to give real-time visual feedback; (right) a user interacting with the simulator through the HMD with a haptic device and an EM sensor.

Neonatal ETI is a time-sensitive resuscitation procedure essential for ventilation of newborns. It requires an unusually high level of skill due to the narrow airways, relatively large tongue, anterior glottic position and low respiratory reserve of neonates [6]. Given the difficulty of the procedure and the high rate of complications in this population, effective training is crucial for achieving positive patient outcomes [24][73][34]. Typical training procedures include participating in neonatal resuscitation training programs, practicing on manikin-based simulators and intubating newborns under supervision [3][43]. However, intubation success rates for pediatric residents are low under current resuscitation training programs and show little improvement between years 1-3 of residency (23-25% success rate) [59] [35]. This indicates the inability of current training programs to adequately prepare trainees for clinical performance in the delivery room and neonatal intensive care unit (NICU).

The best way to master the ETI skills is to perform as many supervised practice trials

as possible on patients. However, pediatric residents experience an average of 3 clinical opportunities to perform ETI during 3 years of residency [17], a number that falls far short of the 47-57 intubations required for a consistent 90% success rate [44]. Due to this limitation, ETI training is heavily reliant on physical manikins, which allow learners to achieve some level of competence prior to clinical exposure. However, practicing on current commercial airway manikins has not been found to be a substitute for training on patients [77][46][57][89]. Most of the manikins and simulator-based training typically provide little variation in anatomy or difficulty level, which are key requirements for developing expertise, and do not realistically model the look, feel and motions inherent in real patients. Consequently, learners who train on one manikin with a single airway model may develop model-specific techniques that may not transfer to patients or even other manikins [61]. Moreover, the small size of the intubation space in neonatal models prevent learners from observing the events occurring within the manikin or the patient; learners have poor knowledge about what went wrong and how to improve. Lastly, assessment of ETI performance under artificially ideal conditions likely overestimate trainees' skill level since they do not mimic the stressors and distractions that are inherent in the real clinical environment. To master the procedure, it is essential not only to develop the trainee perception in understanding the overall procedure, but also to require them to perform the operation tasks in complex scenarios. Thus, there is a pressing need for innovative training modalities that can bridge the gap left by traditional training and thereby allow rapid skill acquisition.

VR-based training systems offer an elegant solution to the current need for better training in the medical field, since realistic and configurable training environments can be created without any restriction for repetitive practicing. Many VR ETI simulators have been developed for training medical professionals as reviewed in Section 2.1.2 (Table 5.1); however, due to the various unsolved technical issues, ETI simulation has not been widely used. Such issues include modelling complex anatomical structures and large soft tissue

Table 5.1: Comparisons with current state-of-the-art of VR ETI simulators.

	Virtual Airway Skills Trainer [18]	Airway Lab VR [83]	ETI Simulator [87]	AirwayVR Trainer [65]	Our ETI Simulator [88]
Immersive Environment	✓	✓	✗	✓	✓
Simulation of Entire Procedure	✗	✓	✓	✓	✓
Physical Modeling	Mandible, Neck	Mandible, Tube	Tongue	Mandible	All Object Types
Collision Detection	✗	Skeleton-Tube	Tool-Tongue	Mandible-Tongue	All Object Types
Realistic Haptics	✗	✗	✓	✗	✓
Levels of Difficulty	✗	✗	✗	✓	✓
Skill Assessment	✗	✗	✗	✗	✓

deformations, handling collisions of frequent tool interactions with large forces and rendering realistic haptic feedback. A successful ETI simulator should integrate all these required components in a balanced and efficient manner to achieve both visual and haptic quality at interactive rates.

In this chapter, we aim to address the notable gaps in previous work by developing a physics-based VR ETI simulation by using the proposed simulation framework as described in Chapter 3. The simulation of ETI captures the most important aspects of the aforementioned general problems we are trying to address, such as, multiple heterogeneous scenes composed of different object types, dynamic and real-time interactions of the laryngoscope with multiple tissues, intensive collisions due to large interaction forces and multimodal rendering of the results. The realism of the VR simulator compared to the current manikin-based simulator is evaluated by conducting a validation study with a group of neonatologists.

5.1 Endotracheal Intubation Simulation System

Our VR-based immersive ETI simulation system includes a HTC® Vive™ Pro head-mounted display (HMD) with one hand controller, a Geomagic® Touch™ X haptic device (6 DOF, 3 degree of force output) and a trakSTAR™ electromagnetic (EM) sensor (6 DOF) (Figure 5.1). The HMD with combined resolution of 2880×1600 pixels allows the user to perform the procedure in the virtual environment from any viewpoint just as in real situation instead of a fixed view as in screen-based computer simulators. The hand controller is used to adjust various parameters in the VR simulator, such as head elevation, operation table angle and jaw opening, and to perform oral suction. The haptic device and the EM sensor allow bimanual interaction with the virtual laryngoscope and the ETT during the procedure. To further improve the sensation of grasping the laryngoscope and the ETT, 3D printed laryngoscope handle and ETT connector modeled on measurements of a Miller 1 laryngoscope and a 3.0 mm ETT were attached to the haptic device and the EM sensor respectively. The motion was captured in 3D allowing for real-time and post-trial viewing, visualization from different angles (cross-sectional and video laryngoscopic views) (Figure 5.1) and measurements on the virtual model during the procedure. The motion data were recorded throughout the entire procedure and streamed to a computer for post-trial assessment and statistical analysis.

An overview of the architecture is provided in Figure 5.2. It consists of a main simulation program, an object representation structure (object class) and an abstraction of I/O devices (sensor class). The main simulation program contains the simulation functions and processes the dynamics of objects (deformable or rigid-body dynamics) and their communication with each other. Each object is decomposed into various representations where each representation is more suited toward a particular task - modeling, collision detection and visualization. These representations are linked together so they can be updated coherently. In this way, objects in the PBD layer are able to communicate with objects in the rigid-body dynamics

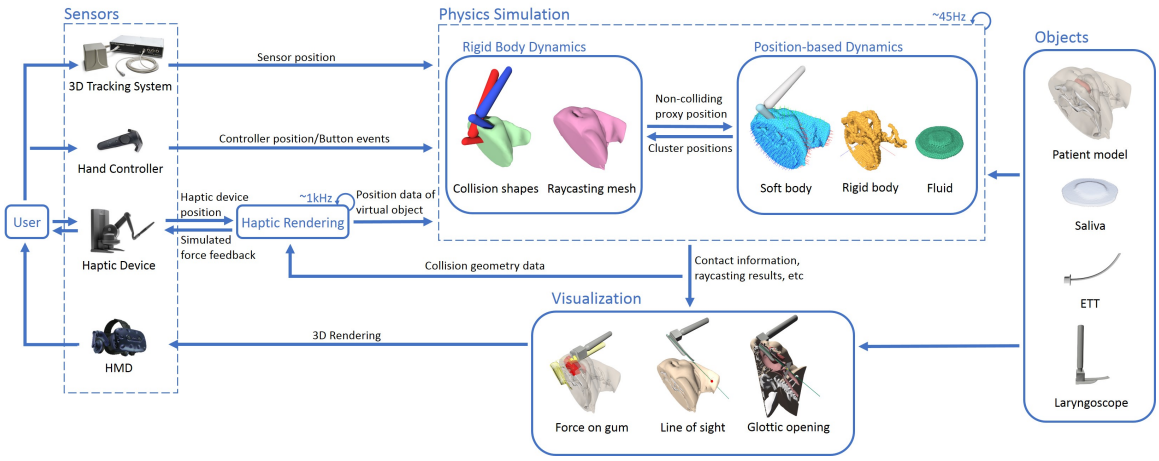

Figure 5.2: System architecture showing interaction flow.

layer.

Our framework uses PBD to simulate the behaviour of human body objects, such as soft tissues, bones and fluids (e.g., blood, secretions), in a unified particle representation, which allows for seamless modeling of different substances. In parallel, an independent rigid-body dynamics model was integrated to account for collisions between the virtual laryngoscope and bones and to prevent the virtual tool from penetrating soft tissues due to large interaction forces. In addition, it provides functionalities, such as contact callbacks, geometry queries (raycasting, overlap tests), etc, for developing real-time visualization. The details of the simulation algorithm (Figure 5.12) will be addressed in the following sections.

5.2 System Calibration

A calibration procedure was performed before the experiment to register each sensor into the same coordinate system (Figure 5.3). We first calibrated the Vive™ system and set its coordinate system as the global coordinate space. The operation area is set as seated-only. The two Vive™ base stations were mounted on two tripods diagonally, about 3m apart from each other, facing toward the center of the operating area. Once the Vive™ system is calibrated, we then registered the haptic device into the global coordinate space. Two sets of corresponding 3D points (at least 3 pairs of points) were collected by waving the haptic

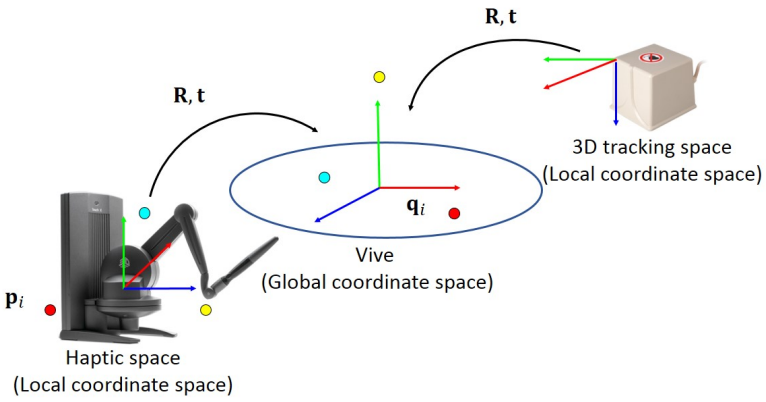

Figure 5.3: System calibration.

arm and the hand controller, with the haptic interface tip pointing at a known position on the hand controller. With the acquired 3D point correspondences, the optimal least square best-fit rigid transformation from the haptic space to the global space was found by the singular value decomposition algorithm [2]. The minimization problem is given by:

$$\sum_{i=1}^{N} ||\mathbf{R}\mathbf{p}_i + \mathbf{t} - \mathbf{q}_i||^2, \tag{5.1}$$

where $\mathbf{p}_i$ and $\mathbf{q}_i$ are sets of 3D points in the haptic space and the global space respectively, with known correspondences, $i = 1, ..., N$. $\mathbf{R}$ is a 3×3 rotation matrix and $\mathbf{t}$ is the translation vector.

Similarly, we calculated the rigid transformation matrix from the 3D tracking system space to the global space for the EM sensor. With the system fully calibrated (error under 2 mm), both the 3D printed laryngoscope and ETT were registered to their virtual counterparts and users had no trouble locating the instruments while they were in the virtual environment.

5.3 Virtual Model Reconstruction

Our virtual simulator was built on a model of CT scans of a real neonatal patient. The stack of CT images forms a 3D volume that contains the upper body of a neonate (Figure 5.4). The mouth, jaws, tongue and airway are the anatomical regions that are directly involved

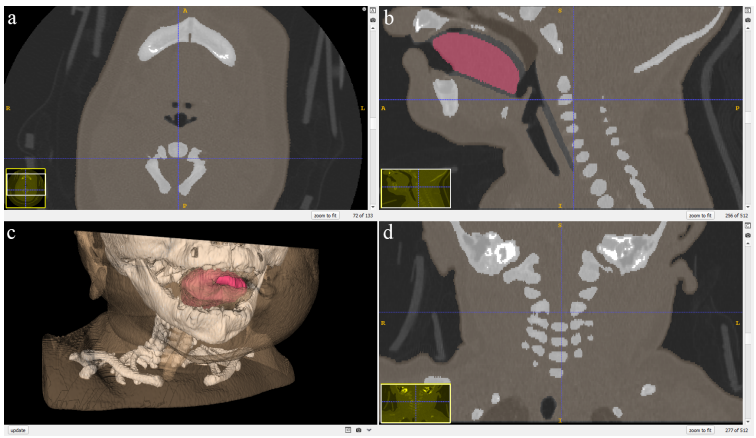

Figure 5.4: Segmentation results. Windows a, b and d display the image volume in the orthogonal axial, sagittal and coronal planes respectively. Window c displays the 3D view of the volume. The color coding highlights different areas of segmentation.

in a neonatal ETI. For simplicity, we segmented the model into three parts: body tissue (including mouth and airway), tongue and bone (including upper/lower jaws). We separated the tongue from the body tissue to allow for varying the size of the tongue or changing the tongue models to simulate different levels of difficulty. In addition, the parameterized model enables adjustments in many other features including head size, jaw opening and mouth secretions. We segmented the CT model in ITK-SNAP [90] and exported the resulting segmentations into surface meshes, which were further smoothed and simplified.

5.4 Position-based Simulation

5.4.1 Rigid Body Simulation

We represented the skeletal structure in our simulation model as rigid bodies using particles and added a rigid shape-matching constraint [55] to each rigid body to maintain particle configurations. Particles were generated by first performing uniform voxelization of each bone mesh and placing particles at each occupied voxel interior to the mesh (Figure 5.2). We then assigned all particles in each mesh the same phase identifier and added only one shape-matching constraint to the body so that all the particles are connected to one cluster

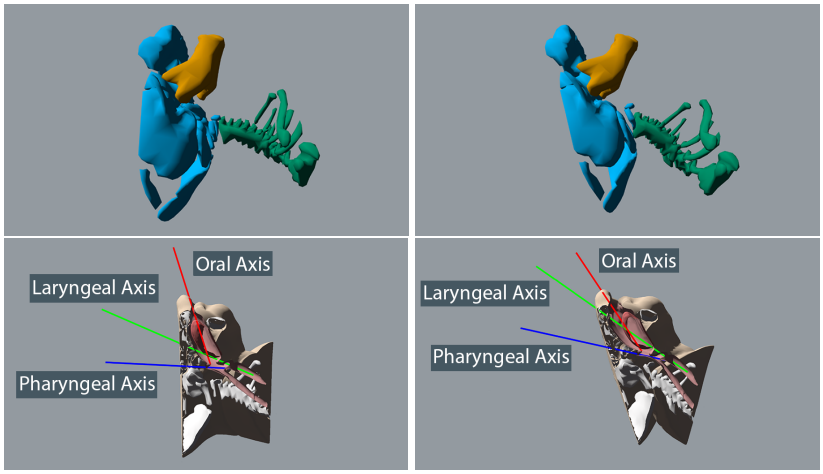

Figure 5.5: Skeletal structure of the model. (left) Original posture; (right) adjusted posture with head elevated and jaw opened. The oral, laryngeal and pharyngeal axes are superimposed on the model to help the user get the optimal head position before the procedure.

with weight 1.0. In this way, the object is treated as a rigid body.

In order to simulate the motion of the patient's head and mandible, an articulated skeletal structure was developed. We segmented the bones into 3 separate meshes as shown in Figure 5.5. The joints of the skeleton were placed at the top of the neck and at the two ends of the mandible to simulate the movement of the cervical vertebrae and the mandible respectively. In order to keep the movements of the head and the mandible realistic, the joint at the neck allows for 30 degrees of vertical movement. The pair of joints at two ends of the mandible allows for 35 degrees of vertical movement. We set all the masses of the bone particles to infinity so that they are treated as static objects. We use the hand controller to adjust the head elevation and the jaw opening. Since there is only one cluster in each rigid body, we do not need any skinning for the corresponding mesh, all the mesh vertices are transformed accordingly by the rotation and the translation of that cluster.

5.4.2 Soft Tissue Deformation

We simulated soft tissues, such as the body tissue and the tongue, and the deformable ETT as soft bodies by using cluster-based shape matching constraints [55]. Particles were firstly

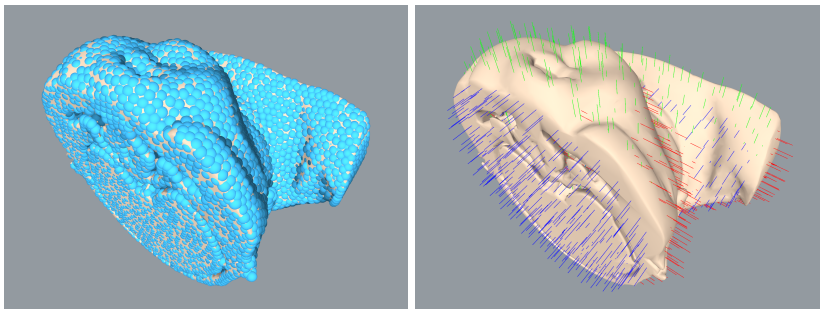

Figure 5.6: Virtual model in PBD. (left) Particle representation of the body tissue; (right) local coordinates of shape-matching clusters.

generated using the same method as described previously for rigid bodies. Instead of having one shape-matching cluster for rigid bodies, we generate multiple clusters for each soft body to allow larger deformations. Clusters are introduced one at a time at particle locations, and the process greedily removes any particles that fall within the bounds of the former, until no particles remain. The resulting shape-matching clusters from this procedure are illustrated in Figure 5.6. With different cluster spacing and stiffness, we can simulate different tissue properties. Each mesh of corresponding soft body is expressed in terms of local cluster positions through linear blend skinning [41]. Each "skinned" vertex is associated with its closest local cluster coordinate frames within a range. Weights fall off inversely with the square of the distance from vertex to local cluster origin. As a result, the mesh will deform in accordance with the manipulated particles.

For this particular procedure, the ability to manipulate the epiglottis with the laryngoscope is critical because it determines if the user can get a clear view of the glottis, which affects the outcome of the overall procedure. However, clusters are uniformly generated and may not be dense enough to allow detailed deformations of small features, such as the epiglottis. Increasing the number of clusters or adaptively generating clusters according to geometric features could solve the problem but introducing a heavy computational overhead and may still not solve the problem perfectly. Therefore, our solution was to create an additional cluster (the local coordinate axes with thicker lines in Figure 5.7(b)) at the center of the particles that belong to the epiglottis and assign all these particles to that cluster. At the

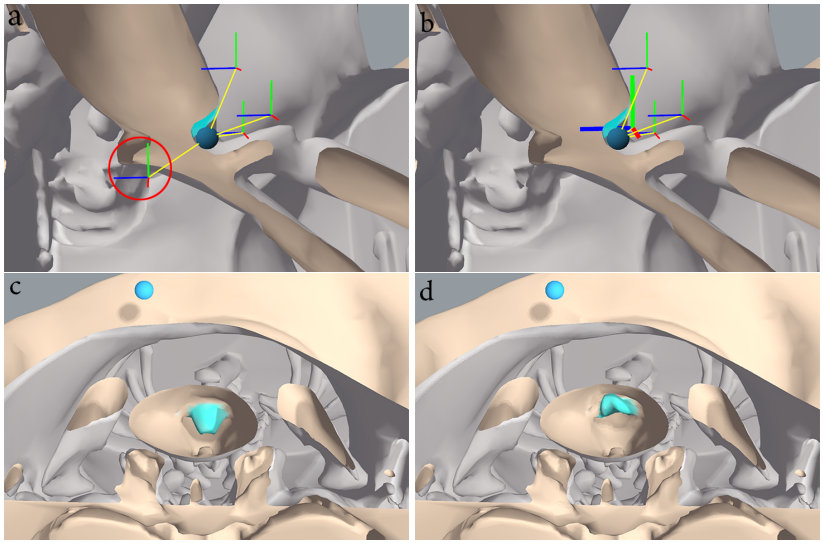

Figure 5.7: Cluster modification. (first row) Cross-sectional views of a particle on the epiglottis with its associated clusters before (a) and after (b) the cluster modification. (second row) Frontal views of a particle on the epiglottis dragged to the upper-left direction before (a) and after (b) the cluster modification.

same time, we removed any clusters from these particles that were incorrectly assigned. As shown in Figure 5.7(a), some particles on the epiglottis were incorrectly connected to those clusters that were not geomatrically close to them, which prevents the local deformation of the epiglottis (Figure 5.7(c)). Lastly, we also removed incorrectly assigned clusters from these skinned mesh vertices on the epiglottis so that the mesh would be deformed correctly.

In order to make the body tissue move along with the skeletal structure realistically, we set the masses of the particles on the body tissue that are close to the bone particles to infinity. In this way, when the user adjusts the skeletal structure, the particles on the body tissue with infinite mass will be moved together. Since all the remaining particles on the body tissue are connected with the particles with infinite mass through shape-matching constraints, they will be deformed accordingly.

The tongue particles and clusters are generated in a similar way as the body tissue but with different cluster spacing and stiffness. In order to attach the tongue to the mouth, for each particle on the bottom side of the tongue, we find its closest mouth particles within a threshold, and create a spring constraint between them (Figure 5.8). With the tongue

Figure 5.8: Virtual model in PBD. (left) Particle representation of the body tissue; (right) local coordinates of shape-matching clusters.

segmented separately, we could parameterize it so that the system can randomly introduce small variations in size that affect the intubation difficulty. A slide bar panel is provided (Figure 5.1) on the graphic user interface to allow the adjustment of the tongue size and other parameters that affect the intubation difficulty. Each time when the mesh of the tongue is changed, the underlying particles, clusters and corresponding spring constraints between the tongue particles and the mouth particles are recreated.

The ETT is also simulated as a soft body. We set the masses of the particles on the ETT connector as infinity so that its pose can be directly controlled by the transformation of the EM sensor. The stiffness values of the body tissue, tongue and ETT were manually tuned during preliminary testing experiments, until suitable visual and tactile perceptions were achieved according to the feedback from an expert neonatologist.

5.4.3 Fluid Simulation

When performing ETI on real patients, intubators often encounter situations where patients have excessive secretions in their mouth, such as saliva, which prevent the intubators from getting a clear view of the glottis. Therefore, suction needs to be performed prior to the procedure. In order to mimic this real clinical situation, saliva was simulated as fluids in our system by adding density constraints [47]. Fluid particles were emitted from the

locations of the salivary glands inside the mouth at a random speed during the procedure (Figure 5.9(d)). The simulation of fluid suction was implemented by pulling the trigger on the hand controller. A high viscosity value was set to simulate viscous effect of the saliva.

5.5 Collision Detection

In the PBD simulation layer, the virtual model of the laryngoscope was approximated with simple collision shapes for fast collision detection against particles. We enclosed the handle and the blade of the laryngoscope within three capsule collision shapes as shown in Figure 5.9(b). The haptic device provides the input position and orientation for the visual model of the laryngoscope and its collision shapes. Since the blade of the laryngoscope is approximated with two capsule shapes, the ETT cannot pass through the blade with no collision filter. Therefore, we only allow the blade capsule shapes to collide with other PBD objects by setting corresponding collision shape channels and the particle phase identifiers. We then created a signed distance field (SDF) collision shape that approximates the shape of the blade but thicker to achieve robust collision detection (Figure 5.9(b)). We chose SDF collision shape because SDFs are very cheap to calculate collisions compared to other types of collision shapes since they are stored as volume textures on the GPU and have an $O(1)$ lookup cost. We set this SDF collision shape to collide only with the ETT particles.

Another issue with the ETT is that it is composed of thin lines of particles, which are prone to penetrate other groups of particles. This makes it hard to insert the ETT into either the airway or the esophagus without penetration. Therefore, we created another SDF collision shape underneath the body by moving the body mesh inwards with a constant offset (Figure 5.9(b)). This offset collision shape, on one hand, can prevent the ETT from penetrating the body tissue. On the other hand, it can prevent the tongue from penetrating the body tissue when the laryngoscope pushes the tongue against it. Moreover, when simulating saliva, this SDF collision shape can also prevent fluid particles from leaking through the body (Figure 5.9(c) and Figure 5.9(d)). We set this SDF collision shape to only collide with

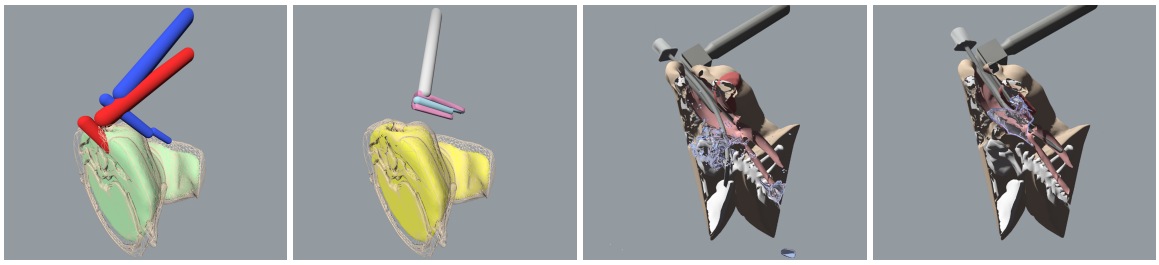

Figure 5.9: (a) and (b) are the collision shapes of the virtual model and the laryngoscope in the rigid-body dynamics layer and the PBD layer respectively; (c) and (d) are the simulation results before and after adding the SDF offset collision shape (yellow mesh in (b)). Without the offset collision shape, the ETT and the tongue easily penetrate the body tissue and the fluids are leaked through the body tissue.

the ETT, the tongue and the saliva particles.

Since the laryngoscope under interaction is kinematic and free from physics in the virtual world, external forces from gravity or collisions will no longer have any effect on it because the position is set absolutely by the user. The held laryngoscope could still apply forces to dynamic objects that are free to move, but if it pushes a dynamic object against a static object (such as the body tissue attached to the bones), the held laryngoscope would easily penetrate the body tissue. Moreover, there is no interaction or collision between the held laryngoscope and static objects, such as the bones. Therefore, we integrated another rigid-body dynamics model [58] as an additional simulation layer in parallel to the PBD layer to account for the collisions for rigid objects.

In the rigid-body simulation layer, we created the same collision shape representation for the laryngoscope as was used in the PBD layer with three capsule shapes (Figure 5.9(a)). The collision shape for the bones was approximated with a single mesh that is the same as the offset collision shape used in the PBD layer, which encloses all the bones (Figure 5.9(a)). This offset collision shape can speed up the collision detection. It can also set a boundary for the amount of deformation that the laryngoscope could apply to the body tissue, preventing penetration. As in the PBD layer, the kinematic laryngoscope in the rigid-body simulation layer would have no interaction or collision with static objects. Therefore, we employed a virtual coupling scheme by creating a dynamic proxy laryngoscope and constraining it to the

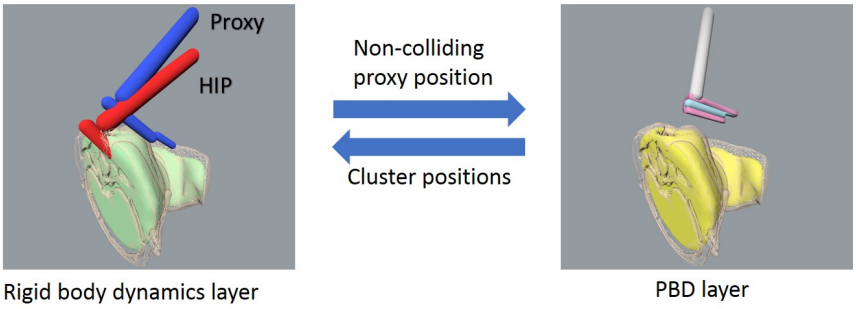

Figure 5.10: Communication between simulation layers.

kinematic laryngoscope (blue and red capsule shapes in Figure 5.9(a)). While the kinematic laryngoscope could penetrate a collision shape, the proxy laryngoscope will respond to collisions and remain on the outside of the volume of the shape.

Both the offset body collision shapes in the PBD layer and the rigid-body dynamics layer are static collision shapes and need to be updated each time the user adjusts the position of the head, the table or the mandible. In each simulation cycle, the rigid-body dynamics layer fetches the cluster positions from the PBD layer to update its collision shapes through skinning. Conversely, the PBD layer receives the last non-colliding position of the proxy laryngoscope from the rigid-body dynamics layer to update the poses of its collision shapes of the laryngoscope (Figure 5.10).

5.6 Haptic Rendering

The simulated force applied to the haptic device must be updated at a high rate of about 1kHz to maintain a realistic haptic feedback. However, the computational process of the collision detection, deformable models and the complex physical simulation are so time consuming that the required update rate of haptic force could not be guaranteed. Therefore, we implemented a multi-thread haptic rendering process to separate the haptic thread from the physical thread. In our implementation, we used two different approaches for computing the force response of the interaction of the laryngoscope with rigid bodies and with soft tissues.

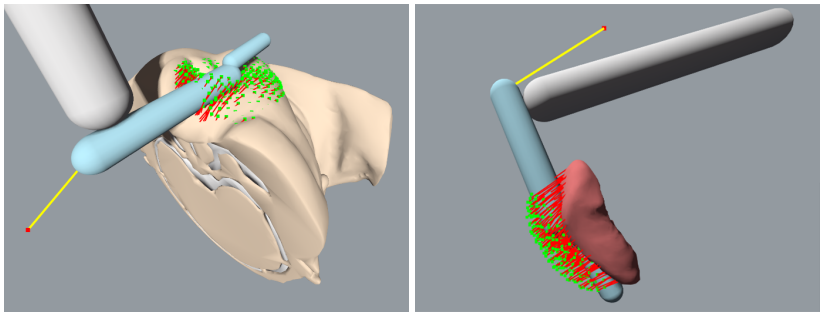

Figure 5.11: Calculation of soft tissue force feedback. The green points are the rest poses of the particles that are in contact with the laryngoscope. The red lines are the displacements of particles. The yellow line on the laryngoscope shows the resulting averaged displacement.

To calculate the force feedback of rigid-body contacts, we first transformed the current position of the haptic interface and the non-colliding position of the proxy laryngoscope to the workspace of the haptic interface. Then, the force was calculated by simulating a virtual spring-damper between the kinematic and the proxy laryngoscope positions. The farther they are separated, the longer the spring is and hence the greater resistance the user will feel, which impedes the motion of the haptic interface from further penetrating the object in contact.

For the force feedback during the interaction with soft tissues, collision tests between the proxy laryngoscope and the particles were computed in parallel in the haptic thread. We calculated the force as the average of all vectors pointing from the current positions of the particles (in contact with the proxy laryngoscope) to their rest positions (Figure 5.11), which were backed up at the initialization of the simulation or reset after each time the user adjusted the position of the head, the table or the mandible. We calculated an average vector for each type of tissue (e.g., body tissue and tongue) that was in contact with the proxy laryngoscope and multiplied each vector by a constant factor μ (stiffness of each tissue type). We finally averaged all those vectors from each type of tissue to generate the force feedback from soft tissues. The summation of the forces due to interactions with rigid bodies and soft tissues was sent to the haptic device for force rendering.

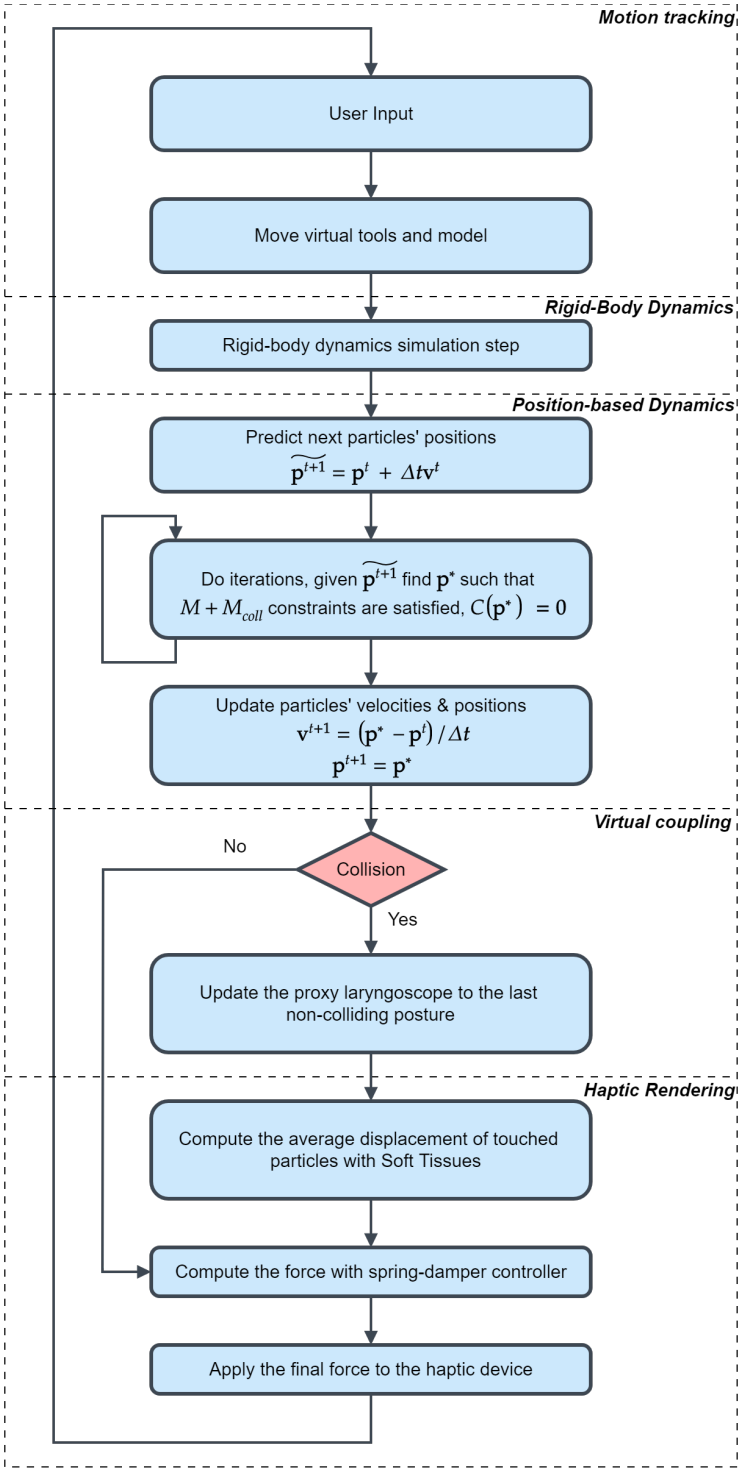

Figure 5.12: Flow chart depicting the overall algorithm of the simulator.

5.7 Automated Visualization

With our manikin-free VR system, we can visualize an even larger set of performance parameters in relation to the internal geometric change of the virtual model during the entire procedure. Our full suite of enhanced visualization tools allows real-time and post-trial viewing, visualization from different angles including cross-sectional and simulated video laryngoscopic views and measurements on the virtual model (Figure 5.1) to provide detailed visual feedback and analysis of multiple aspects of ETI performance. During the procedure, real-time visualization of performance parameters, such as color-coded force on the upper gums, will provide the trainee with visual feedback throughout the entire procedure. At the same time, split windows showing the view of the HMD together with the cross-sectional view and the video laryngoscopic view will provide instructors with a clear understanding of what is happening during each step of an ETI, which is impossible in a manikin-based ETI. After the procedure, 3D visualization of the performance from any viewpoints will provide trainees with information about exactly where and why they failed their attempts. This will not only decrease the time and increase the quality of intubation required to achieve expert-level performance, but also improves retention of skills. For the instructor, post-trial visualization is also useful for evaluation and rating of an ETI performance.

From the raw motion sequences, the simulation system extracted 12 primary performance parameters (Table 5.2). These parameters were chosen *a priori* based on the qualities that our expert instructor deemed as important features. We chose a subset of these parameters that are interpretable for direct visualization. The rest of the parameters are used only for statistical analysis. Based on the contact information and raycasting results returned from the rigid-body dynamics simulation layer, we have implemented visualization of the following performance parameters and some additional features. All these visualizations were rendered on the final visual models to guide the trainees during the procedure.

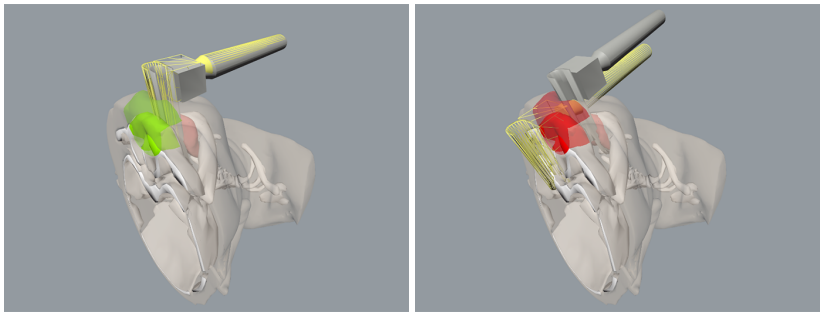

Figure 5.13: Color-coded force indicator on the upper gums, where green stands for acceptable force and red stands for excessive force.

Oral, laryngeal, pharyngeal axes In order to successfully visualize the glottis during the ETI, the head needs to be adjusted to an optimal position where the three axes are best aligned [4]. We manually picked pivot points on the meshes of the virtual model according to the opinion of an expert neonatologist and visualized the three axes to assist trainees in getting the optimal position of the head (Figure 5.5).

Color-coded force on the upper gums During the ETI, the blade of the laryngoscope needs to be adjusted to obtain a view of the glottis and force should be applied on the mandible without pivoting on the upper gums to prevent trauma to the tissue. Therefore, we visualized the force with color changed from green to red on the upper gum area in real time during the procedure, where green stands for acceptable force and red stands for excessive force (Figure 5.13). Force was estimated by calculating the penetration depth of the laryngoscope into the upper gums. At each time step, the contact point and the surface normal at that point is returned if there have collisions between the proxy laryngoscope and the triangle mesh of the upper gums. We then calculate the corresponding point (as the contact point on the proxy laryngoscope) on the kinematic laryngoscope. The vector pointing from the contact point to the corresponding point on the kinematic laryngoscope is the displacement vector. We further calculate the dot product between the surface normal and the displacement vector. If the result is negative, then the displacement vector is the penetration depth; if the result is positive, we translate the kinematic laryngoscope so that

its blade tip overlaps with the proxy laryngoscope. We then repeat the steps we described above to calculate the displacement vector and check if there is penetration. We manually tuned the mapping of the amount of force to color according to the opinion of an expert neonatologist.

Laser dot of sight line In order to assist the subject in identifying the anatomic features viewed through the laryngoscope, a ray-triangle intersection test is performed at each time step to find the first triangle on the mesh model that the sight line emanating from the laryngoscope hits. We visualize the intersection point as a red dot superimposed on the mesh. Whenever the dot is on some important features (uvula, epiglottis, esophagus or glottis), a message showing the name of the feature will also be generated next to the laryngoscope (Figure 5.1). The color of the dot will change to green if the subject is looking at the glottis.

Percentage of glottic opening Visibility of the glottis affects the success of inserting the ETT into the trachea [12]. Therefore, we display the percentage of the glottic opening on the laryngoscope handle in real time to help the subject achieve the optimal view of the glottis with the largest visible area. A glottis plane is defined by manually selecting three pivot points on the vocal cord as shown in red, green and blue points in Figure 5.14. The line passing through the top and the bottom points (red and blue points) is the longest axis of the glottis and the segment between the two points is the longest diameter of the glottis. At each time step, an intersection point (yellow points) is calculated between the sight line and the glottis plane. We then project the vector pointing from the bottom point to the intersection point (blue point to yellow point) onto the longest axis. The resulting projection is the length on the longest axis of the glottis that is visible to the subject. We further divide the projection by the longest diameter to approximate the percentage of the glottis area that is visible to the subject.

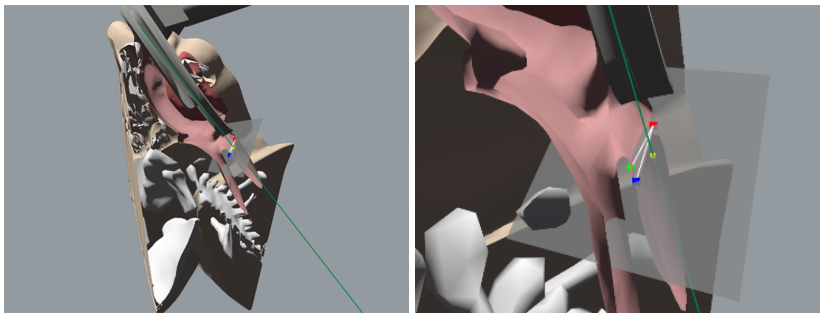

Figure 5.14: Calculation of the percentage of glottic opening. A glottis plane was defined by selecting 3 pivot points (red, green and blue) on the vocal cords. The white line passing through the top (red) and the bottom (blue) points is the longest axis of the glottis and the segment between the two points is the longest diameter of the glottis. The yellow point is the intersection point between the sight line and the glottis plane.

Trial Outcome Correct ETT placement on manikin-based simulators is usually checked by inflation test via bag-valve ventilation. The small sizes of the intubation space in neonatal models do not allow instructors to observe the ETT placement directly. However, with our VR simulator, not only can instructors directly observe the ETT placement through the cross-sectional view, but also the system can automatically check if the ETT is inside the airway or the esophagus. Mesh vertices were manually selected on the airway and the esophagus respectively. Two oriented bounding boxes were constructed to enclose the airway mesh vertices and the esophagus mesh vertices. At each time step, we check if any of the ETT particles are inside the bounding boxes. A success message displays if the ETT is in the bounding box of the airway; otherwise, a fail message displays.

5.8 Validation Study

5.8.1 Study Design

To quantify the relative realism of the VR simulator compared to the manikin-based simulator, we conducted an experimental study using a modified backward transfer or concurrent validation study [75]. The study was approved by the Children's National Health Systems' Institutional Review Board. A group of experts (attending neonatologists) performed ETI on

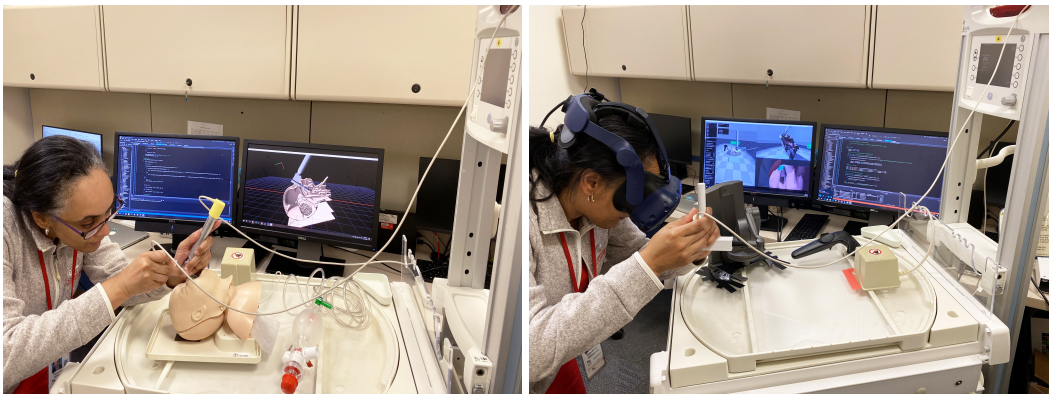

Figure 5.15: AR and VR setup for the validation study.

both the manikin-based augmented reality (AR) simulation system and the VR simulation system (Figure 5.15). The AR system includes a standard full-term Laerdal® task trainer manikin, a laryngoscope with a Miller 1 blade and a 3.0 mm ETT, which were registered to their virtual CT scanned counterparts. The motion of the laryngoscope and the manikin were captured by trakSTAR™ EM sensors.

Four conditions were tested for each subject using a cross-over design: (1) using the AR system without enhanced visualization; (2) using the VR simulator without enhanced visualization; (3) using the VR simulator with increased level of difficulty (without enhanced visualization) and (4) using the VR simulator with enhanced visualization. In condition (2), subjects intubated a virtual patient that had the same size as the manikin in the AR system with a virtual Miller 1 blade and a virtual 3.0 mm ETT. In condition (3), the virtual patient was configured to a smaller premature infant size with relatively larger tongue and smaller jaw opening. The saliva simulation was turned on with specular light simulating glossy tissue surface. A virtual Miller 0 blade and a virtual 2.5 mm ETT were used. In condition (4), the virtual patient had the same configurations as the one in condition (2) but with full-suite of enhanced visualization tools. Before using the AR and the VR simulators, the subjects were provided instructions on their use and given a chance to perform one successful intubation on each system. The order of using the AR or the VR system (condition (1) and condition (2)) was counterbalanced, and subjects were randomly assigned to each simulator. After that,

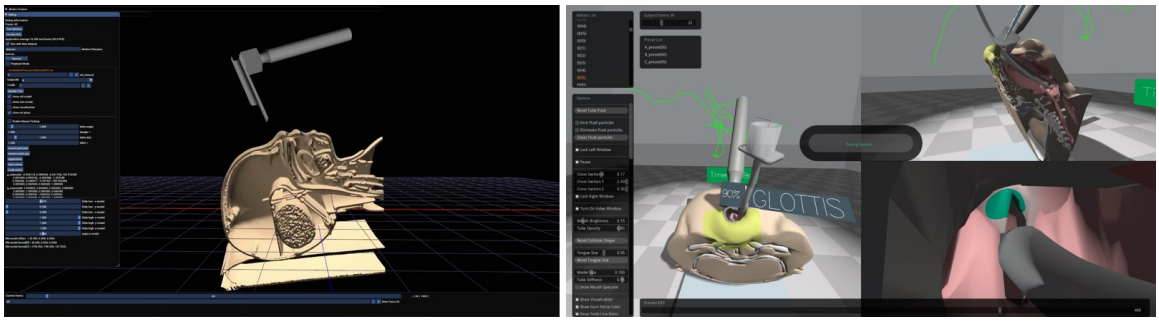

Figure 5.16: 3D playback of the AR and the VR systems.

each subject performed condition (3) and (4) respectively in the same order. During each intubation attempt, motions of the laryngoscope and the ETT were tracked and recorded.

5.8.2 Questionnaire & Measures

A questionnaire with 17 questions was administered immediately following the completion of certain conditions (Table 5.3). The questions were grouped into realism of anatomy and soft tissue, motion consistency of tools, difficulty of level change and acceptance of visualization. Lastly, each subject rated their overall recommendation for the VR simulator. Each question was scored on a 5-point Likert scale, where 1 stands for "very poor" and 5 stands for "very good" (For Q10, 1 = "much easier", 5 = "much harder" and for Q17, 1 = "definitely not recommend", 5 = "definitely recommend"). Subjective feedback and demographic information were collected at the end of the study.

After the trials, an expert instructor (not including the participants themselves) viewed the 3D playback of motions and rated the performances under the 4 conditions using the enhanced visualization tools for both systems (Figure 5.16). Performance ratings consist of a single score on a 5-point scale. Before rating the performances, the rater engaged in a calibration exercise, in which she viewed several bad and good performances (based on the rater's response during the capture stage) and give preliminary ratings with the goal of maximizing intra-rater reliability. To minimize bias, the rater was blinded to the identity of the subjects and the order of the playback was randomized.

Table 5.2: Performance parameters extracted from motion.

Feature	Description
Force on gum	Force of laryngoscope applied against upper gums
Depth	Insertion depth of laryngoscope into mouth
Pitch peaks	Number of up and down movements of blade
Yaw peaks	Number of side to side movements of blade
Pitch rocking	Angular displacement of up and down motion of blade
Yaw rocking	Angular displacement of side to side motion of blade
Attempts	Number of complete attempts for successful placement of ETT
Trial time	Time to complete the procedure
Percentage of glottic opening (VR)	Glottic opening area visible to the subject
Velocity of ETT tip (VR)	Velocity of ETT tip during insertion
Head angle (VR)	Angle between the adjusted pose and the initial natural pose of head
Blade slips (VR)	Number of blade tip sliding off the epiglottis

In addition, the computer systems extracted 8 primary performance parameters from the AR simulator and additional 4 more parameters from the VR simulator (Table 5.2). Among these parameters, the pitch/yaw related parameters were extracted from the angular displacements of the blade in the pitch/yaw directions during the procedure. In particular, pitch/yaw peaks are those peaks on corresponding angular curves that have a prominence of at least 5 degrees, which reflects the number of repositions of the blade. Pitch/yaw rocking were evaluated based on the residuals between the corresponding angular curves and their denoised curves by using wavelet transform, which quantifies the displacement of the repositions of the blade in pitch/yaw direction. Data were analyzed using software developed in-house and Matlab®. To compare participants' ratings, performance scores and parameters between manikin and VR simulators, different levels of difficulty and with/without visualization, we performed bootstrap paired mean tests. A P-value ≤ 0.05 implies a statistically significant difference between two conditions.

5.8.3 Results

In total, 15 subjects participated in the study, with an age range of 30-61 years. The subjects have an average of 14 years of work experience in neonatal intubation. The results of the questionnaire are summarized in Table 5.3. Subjects' performance scores and parameters are presented in Figure 5.17 and Table 5.6.

Comparison between Manikin and VR With respect to the realism of the anatomy (Q1 vs. Q2) and the soft tissue (Q3 vs. Q4) compared to live patients, subjects have rated significantly in favor of the VR simulator ($P = 0.011$ and $P < 0.001$) (Table 5.3). In addition, though not significant, the feelings of manikin and VR patient (Q5 vs. Q6) compared to live patients are on average better for the VR simulator. For the motion consistency of virtual tools in the VR simulator, subjects gave an average score of 4.0 for the laryngoscope and an average score of 3.2 for the ETT.

The relatively lower ratings on questions Q6 to Q8 are associated with the hardware limitations of the haptic device and the EM sensor. 56% of the subjects pointed out that the haptic force was not strong enough to stop them from moving forward when the virtual laryngoscope had a collision. Moreover, 33% of the subjects noticed that there was no force resistance from the upper gums when they were rotating the blade to get a view of the glottis. These motion inconsistencies can be explained by the fact that the haptic device lacks torque output from rotational movements and has a maximum force output. Therefore, torque output from 6 DOF force feedback devices with stronger maximum forces would solve the problem and could be integrated in the future. For Q8, 44% of subjects noticed that they could not feel the force resistance from insertion of the ETT into the mouth, which is due to the fact that the EM sensor does not offer force feedback. Another haptic device or other devices with force feedback would solve the problem. Also, 78% of the subject said that the virtual ETT was too floppy compared to real ETTs and hard to control its movement. This is a limitation of the physics simulation, which is a trade-off between precision and

Table 5.3: Results of the questionnaire.

Condition	Question	Mean (SD)	*P*-value
After (1) and (2)	1. Realism of Manikin anatomy	3.8 (0.83)	0.011
	2. Realism of VR anatomy	4.6 (0.53)	
	3. Realism of Manikin soft tissue	3.0 (1.00)	< 0.001
	4. Realism of VR soft tissue	4.6 (0.53)	
	5. Feel of Manikin	3.2 (0.97)	0.667
	6. Feel of VR patient	3.4 (1.13)	
	7. Motion consistency of scope	4.0 (1.32)	
	8. Motion consistency of ETT	3.2 (1.30)	
After (3)	9. Realism of saliva	3.7 (0.71)	
	10. Difficulty of level change	4.2 (0.97)	
After (4)	11. Helpfulness of three axes	4.4 (0.73)	
	12. Helpfulness of force indicator on upper gum	4.4 (0.53)	
	13. Helpfulness of laser dot	4.6 (0.53)	
	14. Helpfulness of percentage of glottic opening	4.1 (0.93)	
	15. Helpfulness of timer	4.2 (0.83)	
	16. Acceptance of overall visualization	4.4 (0.73)	
Overall	17. Recommendation of VR	4.9 (0.33)	

2.5-3.5 3.5-4.5 ≥4.5

robustness.

The average score of subjects' performance rated by instructors on VR for condition (2) (4.0) is higher than that on the manikin (3.4), even though the difference did not reach statistical significance ($P = 0.105$) (Figure 5.17). With respect to the subjects' performance parameters in those two conditions, the average force on gum and pitch rocking are both significantly lower on VR than those on manikin ($P = 0.028$ and $P < 0.001$) (Table 5.6), which indicate subjects performed better ETI on the VR patient than on the manikin. There is no significant difference on the rest of the parameters except the time, which is significantly longer on VR than on manikin ($P < 0.001$). This can probably be explained by the fact that all of the subjects had experience with manikin-based simulators, but only one of them had experience with VR simulators and none with video games.

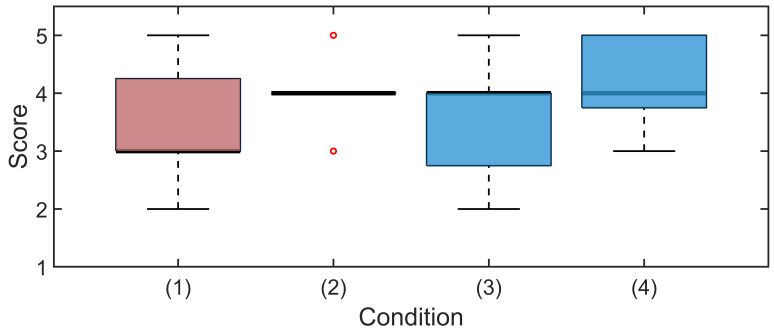

Figure 5.17: Results of performance scores rated by experts for 4 conditions.

Table 5.4: Results of performance parameters for condition (1) and (2). Mean (SD) values are presented for each parameter. For force, depth, glottic opening and ETT velocity, the maximum (first row) and mean (second row) values were both extracted from the corresponding time sequences of each subject.

	Condition (1)	Condition (2)	*P*-value
Force on gum	5.3±2.98	4.6±2.45	0.541
	3.2±1.45	1.8±1.41	0.028
Depth	7.3±0.88	7.3±0.55	0.938
	5.5±0.67	5.8±0.66	0.387
Pitch peaks	3.9±2.32	3.4±1.94	0.703
Yaw peaks	4.2±3.23	4.4±2.13	0.923
Pitch rocking	6.7±1.88	5.2±0.67	0.001
Yaw rocking	5.2±1.54	5.5±1.00	0.640
Attempts	1.0±0.00	1.0±0.00	1.000
Time	17.3±5.18	28.6±6.78	<0.001

Comparison between Levels of Difficulty The difficulty of condition (3) (Q10) has been rated much harder compared to condition (2), with an average score of 4.2 (Table 5.3). The result indicates that the VR simulator has the ability to simulate different levels of difficulty, which is one important advantage over manikin-based simulators. This is essential in developing expertise and preventing trainees from learning the skills by memory. The realism of saliva received an average score of 3.7. Even though not strongly positive, all the subjects stated that the simulation of saliva was realistic enough to distract them from getting a clear view of the glottis. 44% of the subjects suggested that the saliva should have

Table 5.5: Results of performance parameters for condition (2) and (3). Mean (SD) values are presented for each parameter. For force, depth, glottic opening and ETT velocity, the maximum (first row) and mean (second row) values were both extracted from the corresponding time sequences of each subject.

	Condition (2)	Condition (3)	*P*-value
Force on gum	4.6±2.4	5.0±2.31	0.623
	1.8±1.41	1.7±1.09	0.635
Depth	7.3±0.55	5.4±0.64	<0.001
	5.8±0.66	4.0±0.74	<0.001
Pitch peaks	3.4±1.94	8.2±2.83	0.026
Yaw peaks	4.4±2.13	9.4±3.01	0.074
Pitch rocking	5.2±0.67	6.8±1.56	0.027
Yaw rocking	5.5±1.00	5.5±0.67	0.933
Attempts	1.0±0.00	1.7±0.52	0.046
Time	28.6±6.78	59.7±28.19	0.008
Percentage of glottic opening	69.7±14.56	54.1±32.08	0.160
	62.3±13.21	42.2±38.75	0.076
ETT Velocity	18.5±5.72	32.6±28.16	0.108
	6.1±1.51	5.8±2.68	0.764
Head angle	18.0±7.57	17.7±8.84	0.928
Blade slips	0.3±0.50	0.8±0.76	0.339

a foamier appearance.

In addition, though not statistically significant, the average score in condition (3) is lower than that in condition (2) (3.6 and 4.0, $P = 0.215$) (Figure 5.17). With respect to the performance parameters, there are significant increases in pitch peaks and rocking in condition (3) compared to condition (2) ($P = 0.022$ and $P = 0.024$) (Table 5.6), which indicate that increasing the difficulty level caused more repositions of the blade with a larger maximum displacement. Moreover, the subjects had more attempts ($P = 0.042$) and spent longer time ($P = 0.008$) with increased level of difficulty. The maximum and mean depth in condition (3) are both significantly smaller than that in condition (2) ($P < 0.001$ and $P < 0.001$), which is because the virtual model in condition (3) is a premature infant.

Table 5.6: Results of performance parameters for condition (2) and (4). Mean (SD) values are presented for each parameter. For force, depth, glottic opening and ETT velocity, the maximum (first row) and mean (second row) values were both extracted from the corresponding time sequences of each subject.

	Condition (2)	Condition (4)	P-value
Force on gum	4.6±2.45	3.1±1.36	0.014
	1.8±1.41	1.5±1.06	0.160
Depth	7.3±0.55	7.3±0.72	0.744
	5.8±0.66	5.4±0.50	0.133
Pitch peaks	3.4±1.94	2.3±1.00	0.113
Yaw peaks	4.4±2.13	2.7±0.87	<0.001
Pitch rocking	5.2±0.67	5.4±0.58	0.602
Yaw rocking	5.5±1.00	4.2±0.16	<0.001
Attempts	1.0±0.00	1.0±0.00	1.000
Time	28.6±6.78	20.3±6.83	0.002
Percentage of glottic opening	69.7±14.56	81.5±23.87	0.200
	62.3±13.21	80.6±25.30	0.041
ETT Velocity	18.5±5.72	19.9±7.21	0.550
	6.1±1.51	8.1±2.84	0.061
Head angle	18.0±7.57	21.2±4.23	0.120
Blade slips	0.3±0.50	0.1±0.33	0.211

Evaluation of Enhanced Visualization All the questions regarding the enhanced visualization (Q11 to Q16) received average scores above 4.0, which indicate that our visualization tools are helpful in guiding trainees during the procedure. In particular, the visualization of the laser dot of sight line received an average score of 4.6, which indicates that the identification of the glottis is one of the most critical factors in performing a successful intubation.

Subjects' scores are on average higher in condition (4) compared to their scores in condition (2) (4.2 and 4.0, $P = 0.517$) (Figure 5.17). By comparing the performance parameters between these two conditions (Table 5.6), the maximum force on gum, the yaw peaks/rocking and the time are all significant lower in condition (4) ($P = 0.014$, $P < 0.001$, $P < 0.001$ and $P = 0.002$). The mean percentage of glottic opening are on average significantly higher in condition (4) with enhanced visualization than in condition (2) ($P =$

0.041). Though not significant, the head angles are also on average larger in condition (4) than those in condition (2) (21.2° vs. 17.7°, $P = 0.124$), which are closer to the optimal angle for intubation of the virtual patient (26°).

Chapter 6: Endotracheal Intubation Skill Assessment

Manual assessment from experts in medical training is a time-consuming and tedious process. Such subjective, highly variable, and resource-intensive assessment method may not only introduce interrater/ intrarater variability, but also represent a serious limitation in many large-scale training programs. Moreover, poor visualization during the procedure prevents instructors from observing the events occurring within the manikin or the patient, which introduces an additional source of error into the assessment. The small sizes of the operation space in the manikin or the patient do not allow instructors to fully visualize the events occurring within the simulator to provide feedback to learners. Use of laryngoscopic views provides instructors with better views of the procedure. However, they do not provide complete 3D information concerning the relationship of the instrument with the internal anatomy. Moreover, current simulators do not provide quantitative feedback regarding the quality of the attempt. Performance assessments require experts to review and rate individual examinee performance, which is highly variable and resource-intensive.

These limitations have driven efforts for the development of automated systems for performance assessments, which is essential in acquiring precise data, setting objective standards, and quantifying the performance. EM, inertial, and optical sensors have been used to capture the motions of the laryngoscope and limited fiducial points on manikins or patients [11][15][16][64][85]. Force transducers and pressure sensitive films have also been used to measure the forces imparted on specific parts of the manikin or patient [15][16][26][85]. Although these efforts provide some additional information about trainees' performance, they do not provide the complete picture of the interaction of the laryngoscope within the manikin. Moreover, these works only included preliminary analysis on differentiating subjects with different skill levels, but no automated assessment method was developed for skill classification or score prediction on subjects' performances. Capturing the complete

motion is difficult without a virtual representation of all the components. A few computer-based ETI simulators [65][83][87] have been developed for training medical professionals as reviewed in Section 2.1.2. However, since the virtual models are not physically simulated, and collisions between different object types are not implemented, their system are not capable of capturing various performance parameters with respect to the internal geometric change of the virtual model. Therefore, they suffer from the same issues as the above manikin-based simulators by not providing a complete picture of the entire procedure. With respect to automated skill assessment, a few of works have been using hidden Markov model [79][91] or convolutional neural network [84] to classify surgical skills and/or predict performance score of a surgery. However, all these machine learning methods lack of interpretable result, which can be given as feedback that is understandable and intuitive in helping trainees reach higher skill levels. In order to make the assessment model more interpretable, some works have been using fully convolutional neural networks [22][23], regression models [20][91], or multilevel models [33] to predict the overall score of surgical performances. However, to the best of our knowledge, no automated assessment system has been developed for ETI training.

In this chapter, we aim to address the notable gaps in previous works by developing an interpretable score prediction model that is based on our automated scoring pipeline as described in Section 3.9.2 to mimic the evaluation of human raters. In Chapter 5, we have presented a physics-based VR ETI simulation system that captures the entire motions of the laryngoscope and the ETT in relation to the internal anatomy of the virtual patient. The ETI system provides a complete visualization of the procedure, offering instructors with comprehensive information for accurate assessment. By using the performance parameters extracted from the motions and the scores rated by experts, a multinomial regression model is developed to automatically assess the ETI performances.

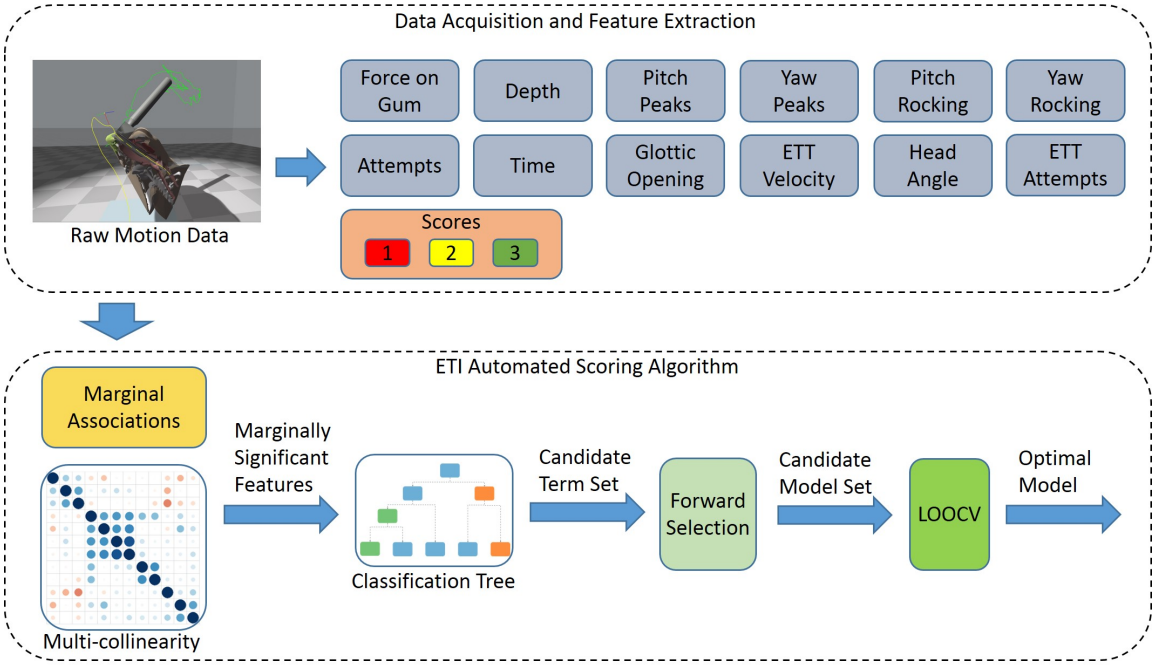

Figure 6.1: Pipeline of ETI automated scoring algorithm.

6.1 Endotracheal Intubation Automated Scoring

To develop an automated assessment system for the ETI performance, we aimed to select important performance parameters on which we can build an interpretable model for the score prediction of intubation performances. Based on the automated scoring algorithm described in Section 3.9.2, the process of finding the optimal prediction model for the ETI performance is shown in Figure 6.1.

From the raw motion sequences, the simulation system extracts 12 primary performance parameters (Table 6.1). These parameters were chosen *a priori* based on the qualities that our expert instructor deemed as important features. These features help further characterize ETI performance and reflect the qualities of motion that are subtle and difficult to quantify by merely looking at the motion playback.

With all the performance parameters (features), we first select the ones that have statistically significant marginal associations with the score using the multinomial GEE method to account for the repeated measures [82]. A feature is considered marginally significant if

Table 6.1: Performance parameters extracted from motion.

Feature	Description
Force on gum	Force of laryngoscope applied against upper gums
Depth	Insertion depth of laryngoscope into mouth
Pitch peaks	Number of up and down movements of blade
Yaw peaks	Number of side to side movements of blade
Pitch rocking	Angular displacement of up and down motion of blade
Yaw rocking	Angular displacement of side to side motion of blade
Attempts	Number of laryngoscope attempts for successful placement of ETT
Trial time	Time to complete the procedure
Percentage of glottic opening	Glottic opening area visible to the subject
Velocity of ETT tip	Velocity of ETT tip during insertion
Head angle	Angle between the adjusted pose and the initial natural pose of head
ETT attempts	Number of ETT attempts

the multinomial regression model against this feature has a P-value < 0.05. In addition, for those marginally significant features, if any two are highly correlated (correlation > 0.8), we exclude one of them from the subsequent steps to prevent collinearity.

We then construct a classification tree for the score with the selected features to identify predictive features and their discontinuous breakpoints, which lead to the piecewise linear terms considered as a part of the candidate term set. We consider these features within three layers from the root of the tree as the most predictive ones. With the linear terms and the piecewise linear terms, we generate their two-way interaction terms.

The candidate model set is determined by the forward selection algorithm (see Section subsection 3.9.2) using the candidate term set acquired from above, which contains three categories of terms: linear terms (the marginally significant features which did not appear in the classification tree), piecewise linear terms (the predictive features in the classification tree), and their two-way interaction terms. Lastly, with the candidate model set S, we find the optimal model that has the highest classification accuracy by performing LOOCV.

6.2 Study Design

This study was conducted in the neonatal intensive care unit at Children's National Hospital, Washington, DC. A total of 20 subjects were recruited in the study, including both attending neonatologists and anesthesiologists with a range of 5-40 years of work experience in neonatal intubation. The study was approved by the Children's National Health Systems' Institutional Review Board. Before using the VR simulator, the subjects were provided instructions on its use and given a chance to perform one successful intubation. After that, each subject performed multiple trials (range from 2 to 8 trials) on the system. During each intubation attempt, all aspects of the motion, including the kinematic data of the instruments (laryngoscope and the endotracheal tube) and the performance parameters, were recorded and streamed to a computer for post-trial assessment.

After the trials, an expert rater (not including the participants themselves) viewed the 3D playback of motions and rated the performances using our enhanced visualization tools. Performance ratings consisted of a single score on a 3-point scale. Before rating the performances, the instructor engaged in a calibration exercise to maximize inter-rater reliability. To minimize bias, the rater was blinded to the identity of the subjects and the order of the playback was randomized. Data were analyzed using software developed in-house, Matlab®, and RStudio®.

6.3 Results

In total, 81 successful trials were collected in our experiment. Figure 6.2 shows the correlation between each of these features with the corresponding score rated by the expert. The marginally significant features were determined, including "Force", "Pitch Peaks", "Yaw Peaks", "Yaw Rocking", "Attempts", "Time", "Glottic %", "Head Angle", and "ETT Attempts". We extracted both the mean and the maximum value for these time series features and found that the maximum values have higher associations with the score. Among these

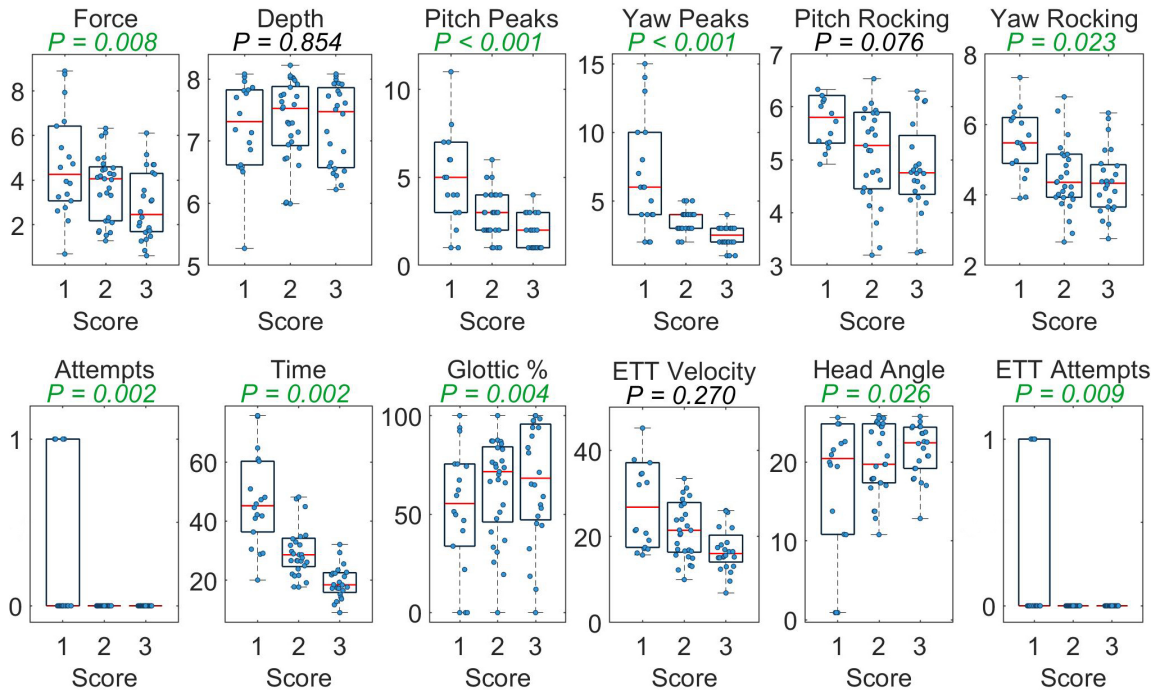

Figure 6.2: Results of the associations between the performance parameters (vertical axes) and the score (horizontal axes). The *P*-value from each GEE model is shown at the top of each plot.

marginally significant features, Pitch Peaks and Yaw Peaks are highly correlated (correlation = 0.87, Figure 6.3). To prevent collinearity, we kept Pitch Peaks in the subsequent analysis because the rater concentrated more on the up and down motion in the assessment.

The result of the classification tree shows that the most predictive features include "Time", "Head Angle", and "Yaw Rocking" (Figure 6.4). These features led to the piecewise linear terms considered in the forward selection algorithm. With the linear terms and the piecewise linear terms, we generated their two-way interaction terms as a part of the candidate term set.

The optimal multinomial regression model obtained after the forward selection and the

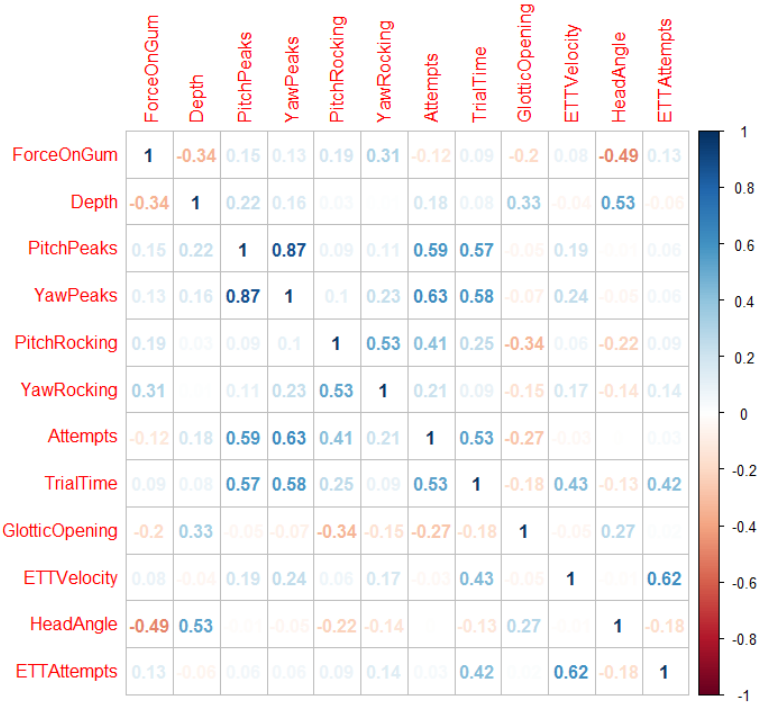

Figure 6.3: The result of the multi-colinearity test.

LOOCV is given as follows: For j =1, 2,

$$y_j = \log\left(\frac{P(\text{Score} \le j)}{1 - P(\text{Score} \le j)}\right) = \beta_{0j} + \beta_1 \cdot F_f + \beta_2 \cdot F_a + \beta_3 \cdot F_g \tag{6.1}$$

$$+ \beta_4 \cdot F_{yr} \cdot I(F_{yr} \ge 4.6) + \beta_5 \cdot F_{ha} \cdot I(F_{ha} < 18) + \beta_6 \cdot F_t \cdot I(F_t < 24.0)$$

$$+ \beta_7 \cdot F_t \cdot I(24.0 \le F_t < 36.0) + \beta_8 \cdot F_t \cdot I(F_t \ge 36.0)$$

$$+ \beta_9 \cdot F_{yr} \cdot I(F_{yr} \ge 4.6) \cdot F_t \cdot I(F_t < 24.0)$$

$$+ \beta_{10} \cdot F_{yr} \cdot I(F_{yr} \ge 4.6) \cdot F_t \cdot I(24.0 \le F_t < 36.0)$$

$$+ \beta_{11} \cdot F_{ha} \cdot I(F_{ha} < 18) \cdot F_t \cdot I(24.0 \le F_t < 36.0)$$

$$+ \beta_{12} \cdot F_{ha} \cdot I(F_{ha} < 18) \cdot F_g + \beta_{13} \cdot F_t \cdot I(24.0 \le F_t < 36.0) \cdot F_f$$

$$+ \beta_{14} \cdot F_t \cdot I(F_t \ge 36.0) \cdot F_f,$$

where F_f, F_a, F_g, F_{yr}, F_{ha}, and F_t represent the features Force, Attempts, Glottic %, Yaw Rocking, Head Angle, and Time respectively. $I(\cdot)$ is the indicator function for the piecewise

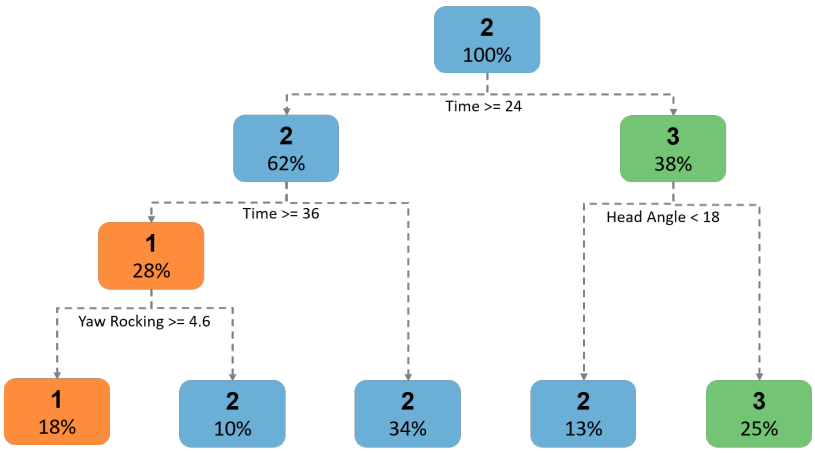

Figure 6.4: The result of the classification tree.

linear terms, and $\beta_k, j = 1, ..., 14$ are the coefficients. The estimations of the coefficients together with their standard errors (SE) and P-values are given in Table 6.2. If we focus on the significant terms where the P-values are < 0.05, we can conclude that: (1) The score is generally negatively associated with a subject's force (applied on the upper gums) and number of attempts (number of blade insertions); (2) Subjects' yaw rocking (displacement of side to side motion of the blade) are also generally negatively associated with the score, especially when the yaw rocking is above 4.6; (3) The score is significantly negatively associated with the time spent for each trial. This coincides with our assumption that the selected features in the optimal assessment model are correlated with the motion characteristics that instructors consider important in training assessment.

Based on the coefficient estimates obtained from Equation 6.1, the probability $P(\text{Score} = y_j), j = 1,2,3$ for a new trial can be derived from the fitted values of y_j, and the predicted score corresponds to the one with the largest probability, which provides an automated score assessment. The LOOCV classification accuracy of this automated scoring system is 80%, which indicates that our system can reliably conduct a consistent and standardized assessment for ETI training and would obviate the need to have an instructor present to rate each trial and would be more objective and less variable. To this end, the machine learning algorithm will need further refinement to allow it to learn from a panel of expert instructors

Table 6.2: The set of features and interactions of the optimal score prediction model.

Term	**Coefficient Estimate**	**SE**	***P*-value**
Intercept	-16.313 (β_{01})	4.677	<0.001
Intercept	-11.496 (β_{02})	3.892	0.003
F_f	0.733 (β_1)	0.338	**0.030**
F_a	4.055 (β_2)	1.400	**0.004**
F_g	-0.035 (β_3)	0.029	0.228
$F_{yr} \cdot I(F_{yr} \geq 4.6)$	0.668 (β_4)	0.186	<**0.001**
$F_{ha} \cdot I(F_{ha} < 18)$	-0.101 (β_5)	0.148	0.495
$F_t \cdot I(F_t < 24.0)$	0.426 (β_6)	0.140	**0.002**
$F_t \cdot I(24.0 \leq F_t < 36.0)$	0.593 (β_7)	0.188	**0.002**
$F_t \cdot I(F_t \geq 36.0)$	0.188 (β_8)	0.063	**0.003**
$F_{yr} \cdot I(F_{yr} \geq 4.6) \cdot F_t \cdot I(F_t < 24.0)$	-0.027 (β_9)	0.015	0.067
$F_{yr} \cdot I(F_{yr} \geq 4.6) \cdot F_t \cdot I(24.0 \leq F_t < 36.0)$	-0.017 (β_{10})	0.007	0.080
$F_{ha} \cdot I(F_{ha} < 18) \cdot F_t \cdot I(24.0 \leq F_t < 36.0)$	-0.011 (β_{11})	0.005	0.170
$F_{ha} \cdot I(F_{ha} < 18) \cdot F_g$	0.004 (β_{12})	0.002	0.260
$F_t \cdot I(24.0 \leq F_t < 36.0) \cdot F_f$	-0.029 (β_{13})	0.016	0.068
$F_t \cdot I(F_t \geq 36.0) \cdot F_f$	0.023 (β_{14})	0.017	0.183

who will develop parameters using a Delphi process; similarly, a larger training sample and a separate validation sample will be required to yield a more robust model.

Chapter 7: Conclusion and Future Work

7.1 Conclusion

In this dissertation, we have presented a practical and efficient simulation framework that offers a completely new training platform with realistic and configurable immersive virtual environments. Our simulation framework is able to simulate complex scenarios with rigid bones, soft tissues and fluids in a unified particle representation at interactive rates using stable and efficient PBD method and allow them to interact with each other seamlessly within the same environment. The test-bed application of neonatal ETI simulation demonstrated the utility of the proposed simulation framework. While the results of this dissertation contribute primarily to neonatal ETI procedure, the interaction approach and simulation could be easily adapted to other multi-modal surgical simulation scenarios that involve interactions between surgical tools and rigid/deformable anatomical structures in virtual environments, which offers a flexible solution for a variety of medical simulation domains and minimizes the efforts required for development. Results from the validation study indicate the anatomy and soft tissue of our VR simulator are more realistic than the manikin-based simulator. Most importantly, our VR simulator is capable of simulating varying levels of difficulty, capturing all motions and giving a complete visualization of the procedure, making the approach a promising platform for medical simulation and training.

Our automated assessment system can extract and visualize a whole set of performance parameters for real-time guidance and quantitatively correlate them with qualities that instructors consider important in their subjective assessment of performance. With these extracted performance parameters and scores rated by experts, we have developed an automated scoring system that gives a consistent and complete evaluation of each trial, which would obviate the need to have an instructor present to rate each trial and would be more objective and less variable.

7.2 Future Work

The simulation framework can be enhanced in a number of ways to aide continuing development of improved medical simulations.

Immersion is an important part of the simulation-based learning experience. In the future version of the framework, advance physically-based rendering, such as sub-surface scattering, shadows, and decals can be integrated besides standard rendering capabilities to increase the visual realism of the virtual models. The realism of the mouth secretions can be improved by simulating spray and foam using diffuse particles. These particles can be generated automatically when fluid particles collide, according to their kinetic energy and relative velocity. The virtual environment can also be improved by mimicking realistic hospital environment. With respect to the haptic realism, 6-DOF force feedback haptic devices with higher maximum forces could be integrated in the future to compensate for lacking of rotational force feedback from manipulating the medical instrument during the interaction with virtual objects. Multiple haptic devices could also be integrated for some procedures that need bimanual operations.

Our generalizable framework enables new simulators to be built efficiently. We will demonstrate the validity of the framework by transitioning much of the code from our existing neonatal ETI simulator to a new medical domain (decompressive hemicraniotomy for trauma (DHT) in neurosurgery).

Our current automated assessment tools provides important automated visual feedback, such as performance parameters superimposed on the virtual model to give trainees real-time feedback, but it lacks the interactivity, contextual awareness, and natural communication, which are the key strengths of live coaching. In the future, we can push this further to develop a virtual coach to deliver effective and efficient feedback in addition to detailed evaluation, demonstrations, and individualized deliberate practice sessions. Automated intelligent coaching offers the promise of human-like feedback in training and skills assessment.

Machine learning algorithms can be used to determine what information to provide at various points of a medical procedure, as well as how to convey the information in a natural and effective way. The resulting automated intelligent coaches will not only provide information that is optimized for human expectations and cognitive/perceptual abilities, but also is highly sensitive to the immediate procedural demands and the larger spatio-temporal context. These automated intelligent coaches will potentially reduce the need for expert human trainers, thus opening the door for skills assessment and refresher training of physicians in remote areas. The automated scoring can also be improved using deep learning algorithms, such as convolutional neural networks. The kinematic multivariate series motion data rather than hand-crafted performance parameters will be used for the development of the automated virtual coaching and scoring system. The approach can be generalized to accommodate a wide variety of procedures that have kinematic VR simulation data of any length and complexity.

Finally, further studies will be conducted to test the long-term training effectiveness of the VR system and the automated assessment system on learning and patient outcomes. We will conduct quantitative studies of ETI training on a group of medical residents and follow them through their 3-year residency in the NICU. This study has the potential to determine the quantitative relationship between simulator realism/better visualization and ETI training outcome.

References

[1] Rajesh Aggarwal, Jonnie Ward, Indran Balasundaram, Parvinderpal Sains, Thanos Athanasiou, and Ara Darzi. Proving the effectiveness of virtual reality simulation for training in laparoscopic surgery. *Annals of surgery*, 246(5):771–779, 2007.

[2] K Somani Arun, Thomas S Huang, and Steven D Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on pattern analysis and machine intelligence*, (5):698–700, 1987.

[3] American Heart Association, American Academy of Pediatrics, et al. Textbook of neonatal resuscitation. *Elk Grove Village, IL: American Academy of Pediatrics/American Heart Asociation*, 2006.

[4] FredaB Bannister and RonaldG Macbeth. Direct laryngoscopy and tracheal intubation. *The Lancet*, 244(6325):651–654, 1944.

[5] Jan Bender, Kenny Erleben, and Jeff Trinkle. Interactive simulation of rigid body dynamics in computer graphics. In *Computer Graphics Forum*, volume 33, pages 246–270. Wiley Online Library, 2014.

[6] J Bercic, M Pocajt, and J Drzecnik. The influence of tracheal vascularization on the optimum location, shape and size of the tracheostomy in prolonged intubation. *Resuscitation*, 6(2):131–143, 1978.

[7] Iago Berndt, Rafael Torchelsen, and Anderson Maciel. Efficient surgical cutting with position-based dynamics. *IEEE computer graphics and applications*, 37(3):24–31, 2017.

[8] Hannes Bleulerc and Gabor Szekelya. Highly-realistic, immersive training environment for hysteroscopy. *Medicine Meets Virtual Reality 14: Accelerating Change in Healthcare: Next Medical Toolkit*, 119:176, 2006.

[9] Matthew Bridges and Daniel L Diamond. The financial impact of teaching surgical residents in the operating room. *The American Journal of Surgery*, 177(1):28–32, 1999.

[10] Mafalda Camara, Erik Mayer, Ara Darzi, and Philip Pratt. Soft tissue deformation for surgical simulation: a position-based dynamics approach. *International journal of computer assisted radiology and surgery*, 11(6):919–928, 2016.

[11] Jestin N Carlson, Samarjit Das, Fernando De la Torre, Clifton W Callaway, Paul E Phrampus, and Jessica Hodgins. Motion capture measures variability in laryngoscopic movement during endotracheal intubation: a preliminary report. *Simulation in healthcare: journal of the Society for Simulation in Healthcare*, 7(4):255, 2012.

[12] RS Cormack and J Lehane. Difficult tracheal intubation in obstetrics. *Anaesthesia*, 39(11):1105–1111, 1984.

[13] Stephane Cotin, David W Shaffer, Dwight A Meglan, Mark P Ottensmeyer, Patrick S Berry, and Steven L Dawson. Caml: A general framework for the development of medical simulation systems. In *Digitization of the Battlespace V and Battlefield Biomedical Technologies II*, volume 4037, pages 294–300. International Society for Optics and Photonics, 2000.

[14] Hans De Visser, Marcus O Watson, Olivier Salvado, and Joshua D Passenger. Progress in virtual reality simulators for surgical training and certification. *Medical journal of Australia*, 194:S38–S40, 2011.

[15] Nathan Delson, Conan Sloan, Thomas McGee, Suraj Kedarisetty, Wen-Wai Yim, and Randolph H Hastings. Parametrically adjustable intubation mannequin with real-time visual feedback. *Simulation in Healthcare*, 7(3):183–191, 2012.

[16] Nathan J Delson, Nada Koussa, and Nabyl Tejani. Measuring 3d force and motion trajectories of a laryngoscope in the operating room. *Journal of Clinical Engineering*, 28(4):211–217, 2003.

[17] Stephen D DeMeo, Lakshmi Katakam, Ronald N Goldberg, and David Tanaka. Predicting neonatal intubation competency in trainees. *Pediatrics*, 135(5):e1229–e1236, 2015.

[18] Doga Demirel, YU Alexander, Tansel Halic, Ganesh Sankaranarayanan, Adam Ryason, David Spindler, Kathryn L Butler, CAO Caroline, Emil Petrusa, Marcos Molina, et al. Virtual airway skills trainer (vast) simulator. *Studies in health technology and informatics*, 220:91, 2016.

[19] Raphael Diziol, Jan Bender, and Daniel Bayer. Robust real-time deformation of incompressible surface meshes. In *Proceedings of the 2011 ACM SIGGRAPH/eurographics symposium on computer animation*, pages 237–246. ACM, 2011.

[20] Mahtab J Fard, Sattar Ameri, R Darin Ellis, Ratna B Chinnam, Abhilash K Pandya, and Michael D Klein. Automated robot-assisted surgical skill evaluation: Predictive analytics approach. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 14(1):e1850, 2018.

[21] François Faure, Christian Duriez, Hervé Delingette, Jérémie Allard, Benjamin Gilles, Stéphanie Marchesseau, Hugo Talbot, Hadrien Courtecuisse, Guillaume Bousquet, Igor Peterlik, et al. Sofa: A multi-model framework for interactive physical simulation. In *Soft tissue biomechanical modeling for computer assisted surgery*, pages 283–321. Springer, 2012.

[22] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. Evaluating surgical skills from kinematic data using convolutional neural networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 214–221. Springer, 2018.

[23] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. Accurate and interpretable evaluation of surgical skills from kinematic data using fully convolutional neural networks. *International journal of computer assisted radiology and surgery*, 14(9):1611–1617, 2019.

[24] Elizabeth E Foglia, Anne Ades, Natalie Napolitano, Jessica Leffelman, Vinay Nadkarni, and Akira Nishisaki. Factors associated with adverse events during tracheal intubation in the nicu. *Neonatology*, 108(1):23–29, 2015.

[25] Anthony G Gallagher and Oscar Traynor. Simulation in surgery: opportunity or threat? *Irish journal of medical science*, 177(4):283–287, 2008.

[26] J Garcia, A Coste, W Tavares, N Nuno, and K Lachapelle. Assessment of competency during orotracheal intubation in medical simulation. *British journal of anaesthesia*, 115(2):302–307, 2015.

[27] Sarah Gibson, Joe Samosky, Andrew Mor, Christina Fyock, Eric Grimson, Takeo Kanade, Ron Kikinis, Hugh Lauer, Neil McKenzie, Shin Nakajima, et al. Simulating arthroscopic knee surgery using volumetric object representations, real-time volume rendering and haptic feedback. In *CVRMed-MRCAS'97*, pages 367–378. Springer, 1997.

[28] Tolga Gokce Goktekin, Murat Cenk Çavuşoğlu, Frank Tendick, and Shankar Sastry. Gipsi: An open source/open architecture software development framework for surgical simulation. In *International Symposium on Medical Simulation*, pages 240–248. Springer, 2004.

[29] Paul J Gorman, Andreas H Meier, Chantal Rawn, and Thomas M Krummel. The future of medical education is no longer blood and guts, it is bits and bytes. *The American journal of surgery*, 180(5):353–356, 2000.

[30] Guy Haller, Paul S Myles, Patrick Taffé, Thomas V Perneger, and Christopher L Wu. Rate of undesirable events at beginning of academic year: retrospective cohort study. *Bmj*, 339:b3974, 2009.

[31] Randy S Haluck, Renee L Marshall, Thomas M Krummel, and Michael G Melkonian. Are surgery training programs ready for virtual reality? a survey of program directors in general surgery. *Journal of the American College of Surgeons*, 193(6):660–665, 2001.

[32] Matthias Harders, Denis Steinemann, M Gross, and Gábor Székely. A hybrid cutting approach for hysteroscopy simulation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 567–574. Springer, 2005.

[33] Randolph H Hastings, Suraj Kedarisetty, Jennifer Moitoza Johnson, Dale Glaser, and Nathan Delson. Manikin laryngoscopy motion as a predictor of patient intubation outcomes: A prospective observational study. *The journal of education in perioperative medicine: JEPM*, 20(1), 2018.

[34] L Dupree Hatch, Peter H Grubb, Amanda S Lea, William F Walsh, Melinda H Markham, Gina M Whitney, James C Slaughter, Ann R Stark, and E Wesley Ely. Endotracheal intubation in neonates: a prospective study of adverse safety events in 162 infants. *The Journal of pediatrics*, 168:62–66, 2016.

[35] Laura Y Haubner, James S Barry, Lindsay C Johnston, Lamia Soghier, Philip M Tatum, David Kessler, Katheryne Downes, and Marc Auerbach. Neonatal intubation performance: room for improvement in tertiary neonatal intensive care units. *Resuscitation*, 84(10):1359–1364, 2013.

[36] David Hellier, Evren Samur, Josh Passenger, Ulrich Spälter, Hans Frimmel, Mark Appleyard, Hannes Bleuler, and Sébastien Ourselin. A modular simulation framework for colonoscopy using a new haptic device. *Studies in health technology and informatics*, 132:165–170, 2008.

[37] Saúl A Heredia-Pérez, Kanako Harada, Miguel A Padilla-Castañeda, Murilo Marques-Marinho, Jorge A Márquez-Flores, and Mamoru Mitsuishi. Virtual reality simulation of robotic transsphenoidal brain tumor resection: Evaluating dynamic motion scaling in a master-slave system. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 15(1):e1953, 2019.

[38] Thomas Jakobsen. Advanced character physics. In *Game developers conference*, volume 3, pages 383–401. IO Interactive, Copenhagen Denmark, 2001.

[39] Ammar Joukhadar and Christian Laugier. Dynamic simulation: Model, basic algorithms, and optimization. *Algorithms For Robotic Motion and Manipulation*, pages 419–434, 1997.

[40] Patrick Kapahnke, Pascal Liedtke, Stefan Nesbigall, Stefan Warwas, and Matthias Klusch. Isreal: an open platform for semantic-based 3d simulations in the 3d internet. In *International Semantic Web Conference*, pages 161–176. Springer, 2010.

[41] Ladislav Kavan. Part i: direct skinning methods and deformation primitives. In *ACM SIGGRAPH*, volume 2014, pages 1–11, 2014.

[42] Kitware. Interactive medical simulation toolkit (imstk) from `http://www.imstk.org/`. 2018.

[43] Jessica J Klotz, Suzanne L Dooley-Hash, Joseph B House, and Pamela B Andreatta. Pediatric and neonatal intubation training gap analysis: instruction, assessment, and technology. *Simulation in Healthcare*, 9(6):377–383, 2014.

[44] Christoph Konrad, Guido Schupfer, Markus Wietlisbach, and Helmut Gerber. Learning manual skills in anesthesiology: is there a recommended number of cases for anesthetic procedures? *Anesthesia & Analgesia*, 86(3):635–639, 1998.

[45] Blazej Kubiak, Nico Pietroni, Fabio Ganovelli, and Marco Fratarcangeli. A robust method for real-time thread simulation. In *Proceedings of the 2007 ACM symposium on Virtual reality software and technology*, pages 85–88. ACM, 2007.

[46] Jeffrey Lubin and Robert Carter. The feasibility of daily mannequin practice to improve intubation success. *Air medical journal*, 28(4):195–197, 2009.

[47] Miles Macklin and Matthias Müller. Position based fluids. *ACM Transactions on Graphics (TOG)*, 32(4):104, 2013.

[48] James Mayrose, T Kesavadas, Kevin Chugh, Dhananjay Joshi, and David G Ellis. Utilization of virtual reality for endotracheal intubation training. *Resuscitation*, 59(1):133–138, 2003.

[49] James Mayrose and Jeffrey W Myers. Endotracheal intubation: application of virtual reality to emergency medical services education. *Simulation in Healthcare*, 2(4):231–234, 2007.

[50] Joe J Monaghan. Smoothed particle hydrodynamics. *Annual review of astronomy and astrophysics*, 30(1):543–574, 1992.

[51] Kevin Montgomery, Cynthia Bruyns, Joel Brown, Stephen Sorkin, Frederic Mazzella, Guillaume Thonier, Arnaud Tellier, Benjamin Lerman, and Anil Menon. Spring: A general framework for collaborative, real-time surgical simulation. *Studies in health technology and informatics*, pages 296–303, 2002.

[52] Andrew B Mor and Takeo Kanade. Modifying soft tissue models: Progressive cutting with minimal new element creation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 598–607. Springer, 2000.

[53] Dan Morris, Christopher Sewell, Nikolas Blevins, Federico Barbagli, and Kenneth Salisbury. A collaborative virtual environment for the simulation of temporal bone surgery. In *International conference on medical image computing and computer-assisted intervention*, pages 319–327. Springer, 2004.

[54] Matthias Müller, Bruno Heidelberger, Marcus Hennix, and John Ratcliff. Position based dynamics. *Journal of Visual Communication and Image Representation*, 18(2):109–118, 2007.

[55] Matthias Müller, Bruno Heidelberger, Matthias Teschner, and Markus Gross. Meshless deformations based on shape matching. *ACM transactions on graphics (TOG)*, 24(3):471–478, 2005.

[56] Andrew Nealen, Matthias Müller, Richard Keiser, Eddy Boxerman, and Mark Carlson. Physically based deformable models in computer graphics. In *Computer graphics forum*, volume 25, pages 809–836. Wiley Online Library, 2006.

[57] Akira Nishisaki, Aaron J Donoghue, Shawn Colborn, Christine Watson, Andrew Meyer, Calvin A Brown, Mark A Helfaer, Ron M Walls, and Vinay M Nadkarni. Effect of just-in-time simulation training on tracheal intubation procedure safety in the pediatric intensive care unit. *Anesthesiology: The Journal of the American Society of Anesthesiologists*, 113(1):214–223, 2010.

[58] NVIDIA Gameworks. Physx from https://developer.nvidia.com/gameworks-physx-overview. 2018.

[59] Colm PF O'Donnell, C Omar F Kamlin, Peter G Davis, and Colin J Morley. Endotracheal intubation attempts during neonatal resuscitation: success rates, duration, and adverse effects. *Pediatrics*, 117(1):e16–e21, 2006.

[60] Vanessa N Palter and Teodor P Grantcharov. Simulation in surgical education. *Cmaj*, 182(11):1191–1196, 2010.

[61] John L Plummer and Harry Owen. Learning endotracheal intubation in a clinical skills learning center: a quantitative study. *Anesthesia & Analgesia*, 93(3):656–662, 2001.

[62] Sukitti Punak and Sergei Kurenov. A simulation framework for wound closure by suture for the endo stitch suturing instrument. In *MMVR*, pages 461–465, 2011.

[63] Kun Qian, Junxuan Bai, Xiaosong Yang, Junjun Pan, and Jianjun Zhang. Virtual reality based laparoscopic surgery simulation. In *Proceedings of the 21st ACM Symposium on Virtual Reality Software and Technology*, pages 69–78. ACM, 2015.

[64] Tariq Rahman, Swapna Chandran, David Kluger, Joseph Kersch, Laurens Holmes, Akira Nishisaki, and Ellen S Deutsch. Tracking manikin tracheal intubation using motion analysis. *Pediatric emergency care*, 27(8):701–705, 2011.

[65] Pavithra Rajeswaran, Na-Teng Hung, Thenkurussi Kesavadas, John Vozenilek, and Praveen Kumar. Airwayvr: Learning endotracheal intubation in virtual reality. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 669–670. IEEE, 2018.

[66] John A Rice and Bernard W Silverman. Estimating the mean and covariance structure nonparametrically when the data are curves. *Journal of the Royal Statistical Society: Series B (Methodological)*, 53(1):233–243, 1991.

[67] Giuseppe Riva, Andrea Gaggioli, Daniela Villani, Alessandra Preziosa, Francesca Morganti, Riccardo Corsi, Gianluca Faletti, and Luca Vezzadini. Neurovr: an open source virtual reality platform for clinical psychology and behavioral neurosciences. In *MMVR*, pages 394–399, 2007.

[68] Alec R Rivers and Doug L James. Fastlsm: fast lattice shape matching for robust real-time deformation. In *ACM Transactions on Graphics (TOG)*, volume 26, page 82. ACM, 2007.

[69] Maria Andréia F Rodrigues, Duncan Fyfe Gillies, and P Charters. Modelling and simulation of the tongue during laryngoscopy. *Computer networks and ISDN systems*, 30(20-21):2037–2045, 1998.

[70] Maria Andréia F Rodrigues, Duncan Fyfe Gillies, and P Charters. Realistic deformable models for simulating the tongue during laryngoscopy. In *Proceedings International Workshop on Medical Imaging and Augmented Reality*, pages 125–130. IEEE, 2001.

[71] Maria Andréia Formico Rodrigues, D Gillies, and Peter Charters. A biomechanical model of the upper airways for simulating laryngoscopy. *Computer methods in biomechanics and biomedical engineering*, 4(2):127–148, 2001.

[72] Evren Samur, Lionel Flaction, Ulrich Spaelter, Hannes Bleuler, David Hellier, and Sebastien Ourselin. A haptic interface with motor/brake system for colonoscopy simulation. In *2008 Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 477–478. IEEE, 2008.

[73] Ronald C Sanders, John S Giuliano, Janice E Sullivan, Calvin A Brown, Ron M Walls, Vinay Nadkarni, Akira Nishisaki, National Emergency Airway Registry for Children Investigators, Pediatric Acute Lung Injury, Sepsis Investigators Network, et al. Level of trainee and tracheal intubation outcomes. *Pediatrics*, 131(3):e821–e828, 2013.

[74] Richard M Satava. Virtual reality surgical simulator. *Surgical endoscopy*, 7(3):203–205, 1993.

[75] Robert E Sedlack, Todd H Baron, Steven M Downing, and Alan J Schwartz. Validation of a colonoscopy simulation model for skills assessment. *The American journal of gastroenterology*, 102(1):64, 2007.

[76] Neal E Seymour, Anthony G Gallagher, Sanziana A Roman, Michael K O'brien, Vipin K Bansal, Dana K Andersen, and Richard M Satava. Virtual reality training improves operating room performance: results of a randomized, double-blinded study. *Annals of surgery*, 236(4):458, 2002.

[77] Ronald D Stewart, Paul M Paris, Gregory H Pelton, and Douglas Garretson. Effect of varied training techniques on field endotracheal intubation success rates. *Annals of emergency medicine*, 13(11):1032–1036, 1984.

[78] Gábor Székely, Ch Brechbühler, R Hutter, Alex Rhomberg, Nicholas Ironmonger, and P Schmid. Modelling of soft tissue deformation for laparoscopic surgery simulation. *Medical Image Analysis*, 4(1):57–66, 2000.

[79] Lingling Tao, Ehsan Elhamifar, Sanjeev Khudanpur, Gregory D Hager, and René Vidal. Sparse hidden markov models for surgical gesture classification and skill evaluation. In *International conference on information processing in computer-assisted interventions*, pages 167–177. Springer, 2012.

[80] T Therneau, B Atkinson, and B Ripley. Rpart: Recursive partitioning and regression trees, r package version 4.1-13. 2018, 2019.

[81] Geb Thomas, Lynn Johnson, Steven Dow, and Clark Stanford. The design and testing of a force feedback dental simulator. *Computer methods and programs in biomedicine*, 64(1):53–64, 2001.

[82] Anestis Touloumis. R package multgee: A generalized estimating equations solver for multinomial responses. *arXiv preprint arXiv:1410.5232*, 2014.

[83] Arch Virtual. Airway lab vr from https://archvirtual.com/project/airway-lab-medical-training-adtalem-global-education/. 2019.

[84] Ziheng Wang and Ann Majewicz Fey. Deep learning with convolutional neural network for objective skill evaluation in robot-assisted surgery. *International journal of computer assisted radiology and surgery*, 13(12):1959–1970, 2018.

[85] B Weinger. Quantifying expert vs. novice skill in vivo for development of a laryngoscopy simulator. In *Medicine Meets Virtual Reality*, volume 11, page 45, 2003.

[86] James D Westwood et al. An interactive simulation environment for craniofacial surgical procedures. *Medicine Meets Virtual Reality 13: The Magical Next Becomes the Medical Now*, 111:334, 2005.

[87] Zhaohui Xia, Nicholas Milef, Adam Ryason, Daniel Jones, Suvranu De, and Stephanie Jones. Development of a physics-based interactive virtual endotracheal intubation (eti) simulator. In *SAGES 2017 Annual Meeting*. SAGES, 2017.

[88] Xiao Xiao, Shang Zhao, Yan Meng, Lamia Soghier, Xiaoke Zhang, and James Hahn. A physics-based virtual reality simulation framework for neonatal endotracheal intubation. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 557–565. IEEE, 2020.

[89] JH Yang, YM Kim, HS Chung, J Cho, HM Lee, GH Kang, EC Kim, T Lim, and Young Soon Cho. Comparison of four manikins and fresh frozen cadaver models for direct laryngoscopic orotracheal intubation training. *Emergency Medicine Journal*, 27(1):13–16, 2010.

[90] Paul A. Yushkevich, Joseph Piven, Heather Cody Hazlett, Rachel Gimpel Smith, Sean Ho, James C. Gee, and Guido Gerig. User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *Neuroimage*, 31(3):1116–1128, 2006.

[91] Aneeq Zia and Irfan Essa. Automated surgical skill assessment in rmis training. *International journal of computer assisted radiology and surgery*, 13(5):731–739, 2018.

[92] Rolf Ziegler, Georg Fischer, Wolfgang Müller, and Martin Göbel. Virtual reality arthroscopy training simulator. *Computers in biology and medicine*, 25(2):193–203, 1995.