



HAL
open science

SmartSIM - a virtual reality simulator for laparoscopy training using a generic physics engine

Zohaib Amjad Khan, Nabeel Kamal, Asad Hameed, Amama Mahmood, Rida Zainab, Bushra Sadia, Shamyl Bin Mansoor, Osman Hasan

► To cite this version:

Zohaib Amjad Khan, Nabeel Kamal, Asad Hameed, Amama Mahmood, Rida Zainab, et al.. SmartSIM - a virtual reality simulator for laparoscopy training using a generic physics engine. The International Journal of Medical Robotics and Computer Assisted Surgery, 2016, 16, pp.437 - 437. 10.1002/rcs.1771 . hal-01509822

HAL Id: hal-01509822

<https://hal.science/hal-01509822>

Submitted on 3 May 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SmartSIM - A Virtual Reality Simulator for Laparoscopy Training using a Generic Physics Engine

Zohaib Amjad Khan, Nabeel Kamal, Asad Hameed, Amama Mahmood,
Rida Zainab, Bushra Sadia, Shamy1 Bin Mansoor, and Osman Hasan

School of Electrical Engineering and Computer Science (SEECS)

National University of Sciences and Technology (NUST), Islamabad, Pakistan

Email: {zohaib.amjad, nabeel.kamal, asad.hameed, amama.mahmood, rida.zainab, bushra.sadia,
shamy1.mansoor, osman.hasan}@seecs.nust.edu.pk

Contact information

Email id: osman.hasan@seecs.edu.pk

Osman Hasan, Ph.D

Head of Department (Research),

Location: A.307 Faculty Block

School of Electrical Engineering and Computer Science,
NUST, Sector H12, Islamabad, Pakistan.

Voice: +92.51.90852137

Web: ohasan.seecs.nust.edu.pk

Category: Original Article

Sources of financial support: National ICT R&D Fund, Pakistan

Word Count: **8340** (from Abstract till Acknowledgements)

Number of Figures: **32** including sub-figures

Number of Tables: **1**

Abstract

Virtual reality (VR) training simulators have started playing a vital role in enhancing surgical skills, such as hand-eye coordination in laparoscopy, and practicing surgical scenarios that cannot be easily created using physical models. In this paper, we describe a new VR simulator for basic training in laparoscopy, i.e., SmartSIM, which is developed using a generic open-source physics engine called the Simulation Open Framework Architecture (SOFA). This paper describes the systems perspective of SmartSIM including design details about both hardware and software components, while highlighting the critical design decisions. Some of the distinguishing features of SmartSIM include: (i) an easy-to-fabricate custom-built hardware interface, (ii) use of a generic physics engine to facilitate wider accessibility of our work and flexibility in terms of using various graphical modelling algorithms and their implementations and (iii) an intelligent and smart evaluation mechanism that facilitates unsupervised and independent learning.

1 Introduction

Laparoscopy is a type of Minimally Invasive Surgery (MIS) in which a surgeon inserts an endoscope and other instruments into the body through small holes and operates upon the patient using the on-screen output from the endoscope. The need of training in such procedures arises due to non-traditional skill requirements, like hand-eye coordination, depth perception, ability to work within a confined space and ability to control non-intuitive pivoted motion of instruments, such as the fulcrum effect that causes the tool endpoints to move in the opposite direction to the surgeon's hands due to the pivot point. Fulcrum effect has detrimental effect on the skill acquisition in endoscopy [1]. Studies have revealed that Laparoscopic training systems provide a more effective training of novices by using inverted simulation about y-axis, to correct for fulcrum effect, compared to actual Laparoscopic footage. Gallagher and Crothers [2] deduced that the experienced surgeons

get adapted to the inverted conditions rapidly and thus the inversion of images about y-axis is also very beneficial for experts.

Traditionally, the training for laparoscopy was usually carried out using box trainers along with physical patient models. However, these methods suffer from lack of objective assessment of skills especially without supervision, inefficient method of training as used material has to be constantly replaced, and lack of a realistic environment closer to actual surgery. Although such training methodologies still persist, yet with considerable enhancement in computing performance, these are being gradually replaced with virtual reality (VR) based training simulators. The surgeons' concerns about effectiveness of VR simulators have also now been subsided to a great extent due to various validation studies that have been carried out over the last few years on existing systems [3] [4] [5].

Some of the commercially available laparoscopic VR simulators include LapSim [6], ProMIS [7], MIST-VR [8], Lap Mentor [9], LapVR [10], SIMENDO [11] and VSOOne [12]. Some other prominent non-commercial simulators include VBLaST [13], GeRTiSS [14] and VESTA [15]. Most of these systems have been developed based on close collaboration between specialists in various fields, like computer science, electrical engineering, mechanical engineering, physics, mathematics and surgery. However, either due to the commercial nature of these projects or the usage of a custom physics engine for developing the simulators, all the effort done is very specific to the particular project and thus cannot be reused. In order to overcome these issues, and to broaden the scope of VR surgical simulator development research, many initiatives, such as SPRING [16], GiPSi [17], MVL [18], SSTML [19], CyberMed [20] and SOFA [21], have emerged that provide generic open-source libraries for developing medical simulators. These simulation architectures provide very flexible options for simulator design. For instance, SOFA provides different implementations of Finite Element Methods (FEM), like tetrahedral, hexahedral and triangular. Thus, in SOFA, and the other similar generic engines, a flexible choice of algorithms is available for modelling different behaviors, like the soft tissue deformations,

and for solving system equations (CG Solver, BTD Solver etc) and for the simulation of surgical practices, like cutting and cauterization of tissue. However, to the best of our knowledge most of these open source simulation frameworks have not been used to develop a complete high-fidelity laparoscopic simulator that is able to cover all the requirements of a surgical simulator system. For example, in case of laparoscopic simulation, SOFA, which is one of the most flexible and modular simulation frameworks, has only been used to develop a part of cholecystectomy procedure [22] without any performance evaluation while SmartSIM has a number of exercises for basic training, general surgery and gynecology in which skill development of user is continuously monitored through a performance evaluation metrics.

In order to demonstrate the capabilities and assess the strengths of the above-mentioned generic open-source libraries for developing medical simulators, we describe the complete system design perspective of a new high-fidelity virtual reality laparoscopy trainer:

SmartSIM, which is based on the generic physics engine, SOFA. The mechanical interface for SmartSIM is a custom-built module that is extremely simple in design and structure.

The whole system has been developed with a modular approach, which allows it to be very flexible in terms of extension. Furthermore, we have also introduced a new type of intelligent metrics in SmartSIM that successfully implements the idea of independent learning via simulators without the need of supervision from expert surgeons.

SmartSIM comes with an edge over the existing systems in the sense that it does not necessarily require specific proprietary hardware interface to be connected to it for work. Instead we have developed our very own low-cost mechanical interface that is very simple in structure and use and serves as an example of how a custom-built interface can be easily integrated into our system. Moreover, SmartSIM also retains the flexibility of SOFA, resulting in a software architecture that can support all commercial devices and a software that is easily modifiable with gradual improvements in algorithms related to different simulation aspects. SOFA provides the capability of adding plug-ins within the system

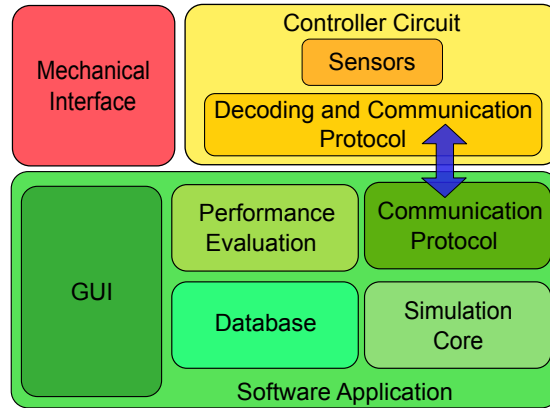


Figure 1: SmartSIM has three main modules: the Mechanical Interface, the Controller Circuit and the Software Application.

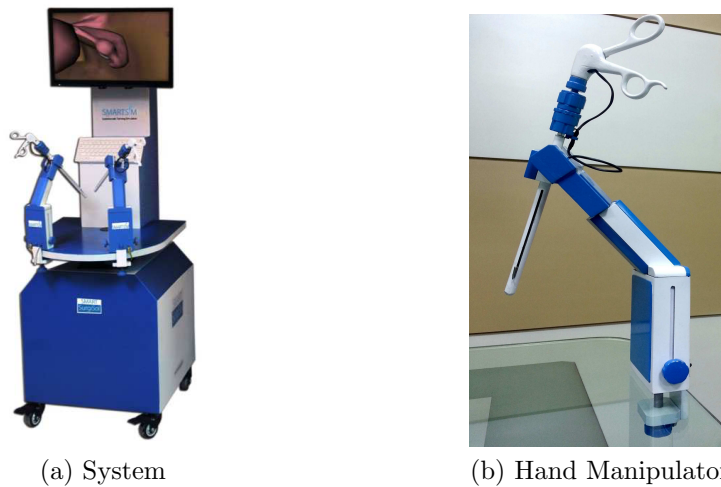


Figure 2: SmartSIM: A virtual reality simulator

which can be used as components in a specific exercise. These plug-ins can contain implementation of advanced algorithms for functionalities like collision detection, force update calculations, visual models and numerical approximations of equations.

2 System Overview

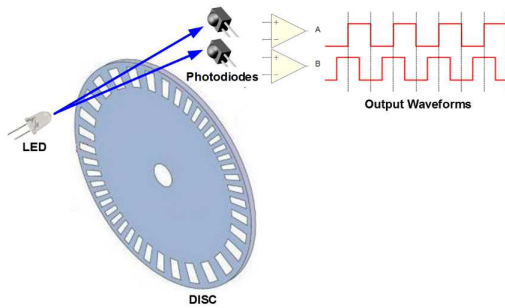
SmartSIM is made up of three main modules: Mechanical Interface, Controller Circuit and Software Application as depicted in Figure 1. The Mechanical Interface module is required for the user to interact with the simulator. The Controller Circuit module is responsible for

the interaction and communication between the Mechanical Interface and the computer. The Software Application module consists of a Graphical User Interface (GUI) and runs the simulation scene based on the selection from a list of scenes.

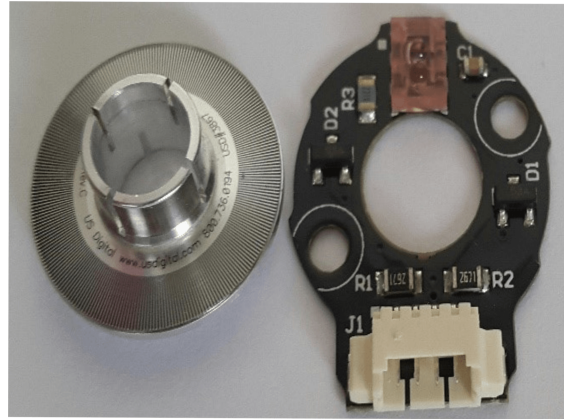
The Mechanical Interface of SmartSIM is a mechanical manipulator that is designed to mimic the actual laparoscopic tool in structure and functionality. A laparoscopic instrument is usually a 5-DoF (Degree of Freedom) mechanical structure. The point of entry into the body inhibits the free movement of the instrument and acts as a pivot around which the tip of the instrument moves. This leaves the surgeon with five possible movements: Superior/Inferior; head to foot movement(pitch), Mediolateral; left to right movement (yaw), rotation about longitudinal axis (roll), Anteroposterior; in and out (translation) and open/close of gripper (grasping). These identified motions along with the workspace requirements formed the basis for designing the Mechanical Interface for SmartSIM.

The SmartSIM system uses encoders, installed on the joints of the Mechanical Interface, to gauge the mechanical movements. These movements are then recorded by a microcontroller and transferred to the computer via a full duplex communication protocol using the USB interface. These features are handled by the Controller Circuit module of the system.

The Software Application module receives the data corresponding to the mechanical movements and translates it into motions of the simulated instrument. Besides that, it also presents the user with a graphical interface to interact with. This GUI includes various options like selecting exercises, using web resources and viewing past performance. A local database runs at the backend to maintain users records. However, the main function of this module is to run real-time simulations for training the users. These simulations are developed using SOFA as the underlying physics engine, which constitutes the core of this entire module.



(a) Optical Shaft Encoder



(b) Coded disc and encoder circuit board

Figure 3: Optical Encoder

3 Mechanical Interface

The Mechanical Interface module of the SmartSIM system, shown in Figure 2b, is a 5-DoF passive mechanical device through which the surgeon interacts with the virtual environment. In MIS, the surgical tool enters the body through a small incision and therefore the possible degrees of freedom are pitch and yaw about the incision (pivot) point in addition to roll, translation and gripper open/close. The Mechanical Interface of SmartSIM is designed to have the same DoFs, in order to provide a natural feeling to the surgeons.

The basic function of the Mechanical Interface is to track the movements made by a user's hand and to transfer this data to the virtual environment. It utilizes optical encoders for this purpose. Optical incremental encoders are selected because of their compact size and high accuracy. Rotary optical incremental encoder is an electromechanical position/speed sensing device, which primarily consist of an LED, light detectors, coded disc, and signal conditioner circuit board as depicted in the Figures 3a and 3b. Incremental encoder normally generates two square waves, each corresponding to an increment of disc rotation. Typically, the LED beam passes through a diaphragm type lens, which splits it into two beams of 90 apart. Light beam passes from channels A and B through opaque and

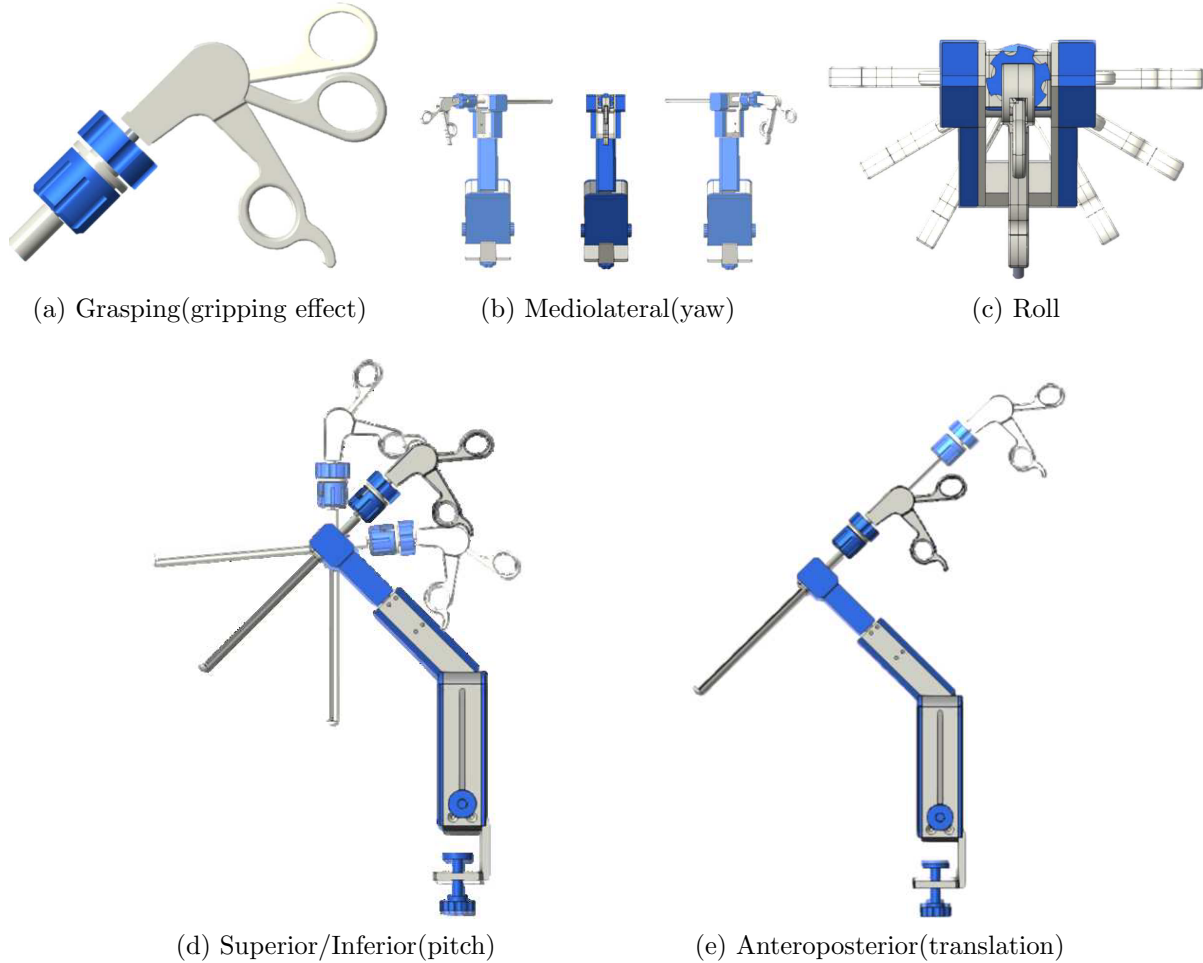


Figure 4: SmartSIM has 5 decoupled DoFs

transparent sections of a disc, which create a light-dark pattern onto the photodiodes correspondences to the disc rotation. This pattern of beams A and B are each received by a separate diode and converted into two square-wave signals, which are 90 out of phase. It is then fed into a microcontroller based dedicated circuit that processes the signals to determine the precise angular position of the shaft.

The pitch and yaw motions are attained by using two orthogonal revolute joints such that their axis of rotation are coplanar and intersecting, hence satisfying the pivot point constraint that is present in actual laparoscopic instruments. The translation motion is achieved by utilizing a rack and pinion mechanism, whereas the roll motion is obtained by using a simple gear train. The Mechanical Interface is designed to provide 5 decoupled

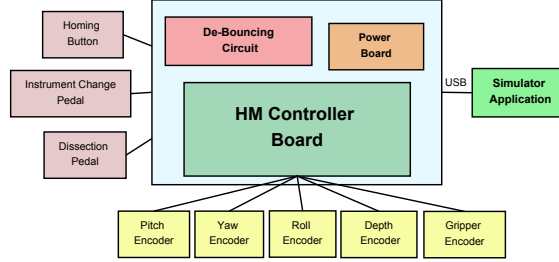


Figure 5: Hand Manipulator (HM) Architecture diagram.

DoFs, as shown in Figure 4.

In our Mechanical Interface design, we have tried to use straightforward mechanisms to achieve the required DoFs which, to the best of our knowledge, is not the case for the existing simulator systems. To achieve pitch and yaw motions, the interfaces for some of the existing simulators, like those by Mentice [23], use the spherical mechanism, which is complex and requires specialized machining for its manufacturing. Other systems, like VBLaST [13], modify existing interface devices, like PHANTOM Omni, to use them as interfaces. The VSOne [12] system uses the double parallelogram mechanism in their interface. The use of the parallelogram mechanism increases the number of mechanical links, which adds to the complexity of the apparatus.

For the Mechanical Interface design of SmartSIM, the workspace requirements are derived from the database of the **Blue Dragon metering system** [24]. Blue Dragon is a passive device equipped with force torque and position center, which calculates the torques and position in-vivo, respectively. Acquiring this data involves 30 surgeons performing 7 MIS procedures on a pig. The required dexterous workspace (DWS) is defined as the region in which 95% of the tool motions are contained. The extended dexterous workspace (EDWS) is defined as the entire abdominal cavity reachable by a MIS tool. The DWS is defined by a circular cone with a vertex angle of 60° and the EDWS is defined by a cone with an elliptical cross section by two orthogonal vertex angles of 60° and 90° [25]. Thus, the pitch and yaw of our Mechanical Interface is $\pm 45^\circ$ and its instrument translation is 200 mm with an unlimited instrument roll.

Besides taking the above-mentioned motions into account, the Mechanical Interface also includes three homing buttons and two pedals. The homing buttons are used to declare a fixed starting position for the instruments and to indicate start of an exercise. The two pedals are used in the same context as in real laparoscopic surgery, i.e., for dissection or coagulation. In addition, a clamping mechanism is also added to the base of the stand, so that it can be easily clamped to the edge of a table. The stand also has a passive translational Dof, which can be used to adjust the height of the Mechanical Interface. Our foremost goals while designing the Mechanical Interface of SmartSIM were to ensure simplicity of mechanism, ease of manufacturing and assembly. Each subassembly has been designed in a manner that it can easily be machined on either manual lathe or manual milling. However, Computerized Numerical Control (CNC) machines will definitely result in more accurate models with fine finish and texture. A considerable amount of attention is also given to ensure a simple assembling procedure of all the sub-assemblies so that even the end user can easily replace the worn-out part. This has been attained by using common fastening techniques.

4 Controller Circuit

4.1 Sensing Circuit

Figure 5 shows a detailed architecture of the sensing circuit of the Controller Circuit module. It comprises of a single high speed Atmega1284p microcontroller, a USB communication circuit, a filtration circuit and encoders interface circuit, which is equipped with dedicated quadrature decoding chips. The algorithms embedded in the microcontroller are able to gauge the gripper, translational, roll, pitch and yaw movements via the optical encoders, attached to each of the five joints of the Mechanical Interface. The homing buttons and two paddles are based on mechanical contact switches. Such a switch typically results in multiple signals against a single push, an effect known as the

switch de-bouncing effect. In order to avoid any false readings, a low pass filter circuit is placed in between the controller unit and the switches to filter out the unwanted and false signals.

Power supply filtration is also used to stabilize the interface and filter the unwanted high frequency spikes in the input power line of 5 Volts that comes from the USB connector. Decoupling capacitors are incorporated to isolate the circuitry from noise and provide a stable input to the circuit.

4.2 Control Algorithm

An event based, interrupt driven algorithm is developed in order to ensure real-time operation, reduce the controller computation effort and eliminate any possibility of missing encoders' critical signals. The microcontroller runs at a maximum frequency of 16 MHz to ensure error free encoders readings. Every encoder is attached directly to the dedicated quadrature decoding chips. Since the human hand movements fall in the range of a few Hz [26], therefore real-time requirements are met with a control loop running at 1 kHz. We used a data transmission baud rate of 19200 b/s and to transmit one tick through virtual com port requires 11 bits (start bit, data bits, parity bit and stop bit). With this baud rate, about $19200 / 11 = 1745.45$ encoder ticks can be transmitted per second. The encoder generates 256 ticks per 360 degrees or one complete revolution with a resolution of $360 \text{ degrees} / 256 \text{ ticks} = 1.40625 \text{ degrees per tick}$. Similarly, $1.40625 * 1745.45 = 2454.53$ degrees of movement can be transmitted per second.

The DoF values, from the encoders, are fed to the control loop running on a specially designed controller board and necessary interface for sensors. The microcontroller converts these values to appropriate signals that can be transmitted via the USB bridge to the computer running the simulator application. The control loop keeps track for every DoF and converts these counts into absolute values, simultaneously.

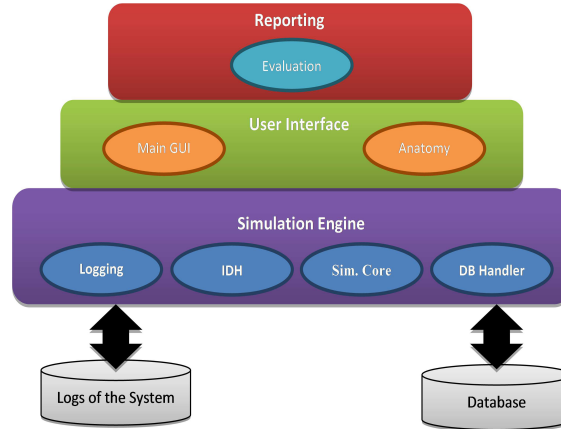


Figure 6: Layered Modular Software Architecture of SmartSIM

5 Simulator Application

SmartSIM currently focuses on training surgeons on the foundational MIS skills, like hand-eye coordination and camera navigation. The software architecture of SmartSIM is based on a layered approach, as shown in Figure 6, with the base is the simulation engine module, which has a generic simulation engine at its core, along with other sub-modules, like log manager, Interface Device Handler (IDH) and Database Handler. The Interface Device Handler sub-module acts as a bridge between the controller circuit and the simulation. In particular, this sub-module performs two functions. Firstly, it provides methods to capture mechanical properties of all the simulated instrument objects in a running exercise and to update positions of those instruments. Secondly, it implements functions to translate the signals received at the USB port of the computer from the controller circuit into position values that can be used as inputs for simulated instruments. The Database (DB) Handler sub-module exposes database to two higher modules of the software. In simple terms, all the functionality related to connection with the database and storing or retrieving information from the database is implemented here. The user interface module can use DB Handler methods to store, update, retrieve and delete account related information in the database. Similarly, the reporting module can use this sub-module to store and retrieve evaluation results for the logged in user. Some modules, like the

database and the logging module, communicate with different layers of the architecture. The database module, as the name suggests, keeps the record of users whereas the logging module maintains logs to track system performance. The application’s entry point in the software architecture of SmartSIM is the user interface module. The reporting module and the user interface module lie at the higher levels of hierarchy with the reporting module being responsible for recording and displaying performance of a user. The simulation core of SmartSIM is based on an open-source framework for medical applications, SOFA, which is preferred to other such applications, like VRASS [27], SPRING [16] and GiPSi [17].

Comparison of these open-source simulation softwares is given in Table 1.

SOFA is a scene graph based library that allows developing simulation scenarios using hierarchical graphs [28]. A particular scene graph contains different components with each component handling one specific aspect of simulation independently. For instance, mass, topology, deformable model, integration scheme and constraints are all represented by different components. This provides a considerable amount of flexibility in the modeling process of simulation since one implementation of a component can be replaced by another without modifying any of the other components in the scene graph. SOFA also provides further modularity by allowing modeling of collision, behavioral and rendering aspects of simulation using different topologies. To maintain consistency between these different topologies, SOFA provides the mechanism of mappings, as shown in Figure 7.

However, the generic physics of SOFA comes with its own set of challenges. One major challenge is to intelligently categorize and model each simulation object in a scene. With the scene graph architecture, it becomes necessary to choose only the required components for a simulation object for avoiding unnecessary computational load. Furthermore it is also essential to identify the right combination of parameters for all the components for a particular simulation object. Examples of these parameters are number of iterations for a linear solver and spring stiffness for a mass-spring model. While some of the parameters may be derived from literature, others need to be adjusted according to simulation

Table 1: Comparison:Open Source Surgical Simulation Tools

Features	SPRING	GiPSi	SOFA	VRASS
Visual Modeling			Y	
DirectionalLight			Y	
OglShadowShader			Y	
SlicedVolumetricModel			Y	
ClipPlane			Y	
Collision Models			Y	
Spheres			Y	
TriangularMeshes			Y	
DistanceFields			Y	
Collision Detection	Y	Y	Y	Y
Proximity			Y	
Continuous			Y	
Deformation Models	Y	Y	Y	Y
Mass-Springs	Y	Y	Y	
LinearFEM	Y	Y	Y	Y
Co-rotationalFEM			Y	
Fluid Simulation		Y	Y	Y
SPH			Y	
Eulerian			Y	
Collision Response	Y	Y	Y	
Penalties			Y	
LCP-based Constraints			Y	
Mapping			Y	
Mechanical Integration	Y	Y	Y	
Euler Explicit			Y	
RK2	Y		Y	
RK4	Y		Y	
Static	Y	Y	Y	
Implicit Euler	Y		Y	
Linear Solver			Y	
Chlesky			Y	
BTDLInear			Y	

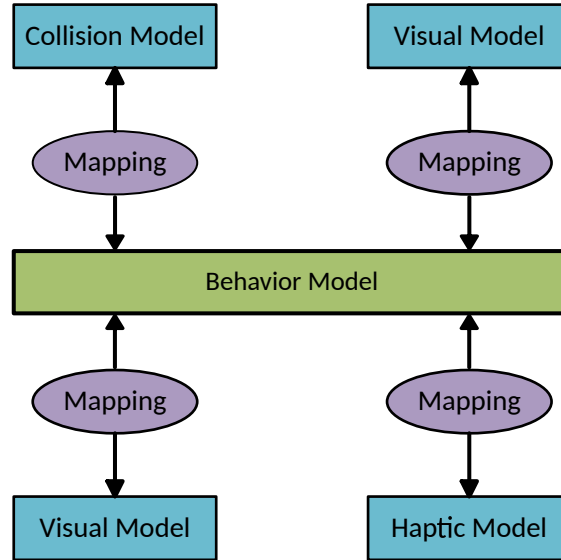


Figure 7: SOFA: Mapping of different models

requirements and topologies being used.

Another major drawback of using SOFA is that it only provides algorithms for simulating instrument functionalities without integrating them within the simulated instruments. For instance, algorithms for topological changes, like those required in cutting and dissection, are included in SOFA, but all these interactions can only be carried out using a mouse cursor. To make these algorithms useful for laparoscopic training in SmartSIM, they are tweaked in such a way that they synchronize with the movements of instruments during simulation and gives a realistic view to the user.

Besides these challenges, another requirement of a VR simulator is to provide features like performance evaluation and visual effects to alert a user when he or she touches a forbidden object. All these features are strictly dependent upon the simulation scene.

Similarly, blood flow and smoke simulation are too generalized and need to be manipulated according to specific requirements. In the following subsections, we explain our methodologies to overcome these challenges in SmartSIM.

5.1 Simulation Object Categorization

Each VR simulation consists of various objects that are either interactive, such as the disks in the peg-transfer exercise in Figure 8a, or non-interactive, such as the floor in the same exercise. In order to facilitate modeling using SOFA in SmartSIM, we classified each simulation object into one of five broad categories, i.e., (i) Visual-only objects (ii) Stationary mechanical objects (iii) Movable mechanical objects (iv) Topologically modifiable objects and (v) Fluids. Classification of each object into one of these five types helps in efficient modeling and in keeping the scene real-time. Once each object is classified, modeling can be done using templates, i.e., a defined way of modeling each type of object using the required components only.

To achieve the state of *suspension of disbelief* i.e., when a user is made to think and feel as if he or she is working in a real environment, collision effects must be accurate. As evident from Table 1, SOFA provides the best collision detection and response as compared to other open-source surgical simulation tools because the models used for visualization in SOFA can be different from the models used for simulation. SOFA provides collision effects for rigid as well as deformable objects. Collision detection approaches implemented in SOFA include distance calculation between primitive pairs, distance calculation between colliding meshes with ray-tracing and points in distance fields [28]. For each simulation object, a collision topology is defined that associates a collision geometry to that object. In SOFA, collision process is divided into different phases which altogether make up a collision pipeline. First the broad phase detection takes place returning pairs of colliding bounding volumes. This is followed by a narrow phase detection process which finds pairs of geometric primitives along with the corresponding contact points. For these contact points, different contact interactions are applied. SOFA supports both penalty-based as well as constraint-based repulsion effects. These repulsive effects are processed with other forces on the object by the solver.

For the purpose of exercise development, one main challenge is to define a suitable

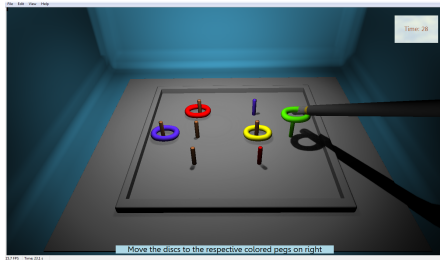
topology for collision for each object. A very fine mesh defined for collision makes the collision responses more realistic but it can slow down the simulation. A coarser mesh, on the other hand, might reduce realism of collision but keeps the simulation real-time. In order to keep the simulation real-time, coarser but well-defined collision meshes are used for simulation objects in each exercise without compromising a lot on realism.

Once the collision models are applied to each simulation object, the standard collision pipeline of SOFA takes care of the collision effects. Every time two objects collide, the contact forces are calculated and are used to give a realistic repulsive effect. To provide enhanced visual feedback based on collisions, we have added exercise specific visual cues to highlight user mistakes. So for instance, if a user collides the instrument with a wrong organ in an exercise, the background turns red to warn the user that a mistake has been committed.

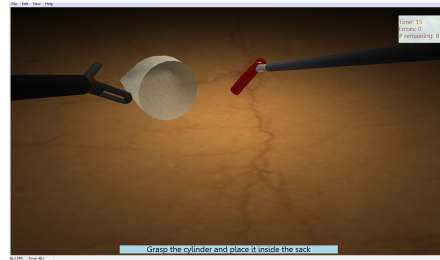
Visual-only objects are non-interactive and do not play any significant role in actual performance of an exercise. Their sole purpose is to fill in the empty spaces, such as providing the background surface. We do not enable collisions for these kinds of objects to avoid intersection testing, which may overburden the simulator unnecessarily.

There are certain objects in a simulation that do not change their positions but do contain collision models based on some topology, e.g., the floor object in the peg transfer exercise (Figure 8a). We have classified such objects as stationary mechanical objects. Modeling of such objects in SOFA does not require the use of mappings or solvers.

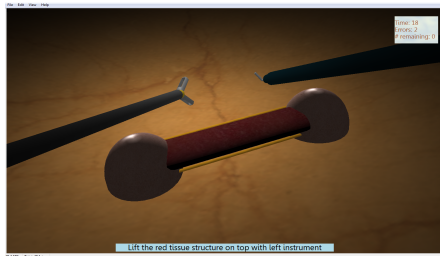
Movable mechanical objects are interactive and can change their positions during a simulation under the effect of forces. They require both the solvers and the mappers. Solvers are required to calculate force, velocity and displacement updates for such objects. Mappers are required to synchronize the visual mesh with mechanical behavior. These types of objects can either be rigid or deformable. In the case of deformable objects, components used to approximate such behavior, like mass-spring models or Finite Element Method (FEM) models, are also required to be added. These methods work on few



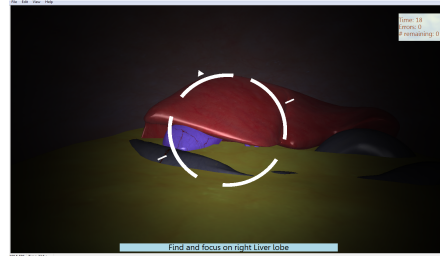
(a) Peg Transfer: hand eye coordination, depth perception.



(b) Grasping: specific grasping, instrument handling.



(c) Fine Dissection: dissection using L-hook.



(d) Camera Navigation of Abdominal Anatomy

Figure 8: SmartSIM exercises

parameters, like elasticity and spring constant, of the object that must be adjusted to get realistic simulations. In SmartSIM, an example of the deformable movable mechanical object is the cylindrical object in the grasping exercise (Figure 8b).

Topologically modifiable objects, as the name suggests, are those simulation objects that require topological changes. Example of these objects are the capillaries modeled in the fine dissection exercise of SmartSIM (Figure 8c). To model this kind of an object in SOFA, it must contain four more components that are specifically designed to handle topological changes [28].

The last category of objects that may exist in a VR surgical simulation is fluid. SOFA provides an implementation of the "Smoothed Particle Hydrodynamics (SPH)" method for modeling the fluid behavior. The particles based components of SOFA consist of particle source and particle sink. Particle source emits particles that are removed from the scene by the particle sink. We have modeled both the blood flow and smoke simulation in SmartSIM using these three components.

5.2 Instrument Simulation

Simulating instrument motion and its functionalities is one of the most important aspects of any medical simulation tool since it provides the first-level interface between the user and the software and hence is an important deciding factor for the quality of a simulator. SmartSIM implements grasping, dissection, cutting, clipping and camera navigation functionalities. The simulated instruments are designed and modeled to ensure that each real movement is translated proportionally to the simulated movement with considerable precision.

5.2.1 Grasping

In SOFA, when a mouse cursor is used for grasping an object, a ray is cast from the cursor position to the object. If the ray intersects part of that object, a spring is created between that point and the cursor position. The object can then be moved using that spring simulating the effect of grasping. In SmartSIM, we have modified this method to change the starting position of the ray to the inside of the jaw of a grasper. Also, we have used multiple rays generated from different points of the grasper to ensure a greater probability of intersection with the object, hence creating a more realistic grasping simulation. The grasper is modeled as an articulated object and the movement of its jaws is synchronized to the gripper motion of the Mechanical Interface. Whenever the jaws are closed, these rays are activated and hence can grasp any object in between using springs. When the jaws are opened the springs are removed releasing the object.

5.2.2 Dissection

SOFA supports dissection via mouse using ray-casting. We have extended the modeling approach used for grasping in this case as well, i.e., using the casted rays with their origins on the tip of the L-hook instrument. These rays are activated whenever the pedal is pressed. Using the mesh removal algorithm, already implemented in SOFA [29], whenever a

ray from the tip of L-hook intersects a topologically modifiable structure, the touched mesh elements are removed.

5.2.3 Volumetric Cutting

SOFA does not provide a reliable solution to the problem of volumetric cutting. For cutting of a volumetric object in SmartSIM, we have used the "Topological Change Processor" component in SOFA that removes the entire mesh within a defined box. In order to apply volumetric cutting using scissors, we first divided the object into virtual symmetric blocks. The position of cutter jaws is noted whenever the jaws are closed. If their position is found to be inside any one of the virtual blocks, the corresponding block is removed using the topological change processor component.

5.2.4 Clipping

To the best of knowledge, SOFA does not support clipping, which requires the constraints to be added to the clipped tissue at the area of contact. In SmartSIM, we have implemented clipping, such that, whenever the clipper jaws are closed a new clip object is added to the scene graph at runtime. The position of the clip is extracted from the position of the jaws. Moreover, if the clipper jaws are closed over the tissue, its points above and below are attached together using a stiff spring. This mimics the compressed tissue behavior.

5.2.5 Camera

Simulation of endoscope functionality was relatively easy to implement in SOFA. For this purpose, we used the camera API, already present in SOFA based on OpenGL, to move the view in correspondence with the movement of the Mechanical Interface. Figure 8d shows the implemented camera functionality in SmartSIM in the camera navigation exercise.

5.3 Soft Tissue Modeling

Soft tissue modeling is undoubtedly the most important aspect of surgery simulation.

Despite the enormous increase in processing power of computers, a trade-off needs to be reached between accuracy of the deformable model and simulation time to keep the simulation real-time.

Various models exist in literature that can be used to simulate deformable organs [30].

Among them Finite Element Method (FEM) and Mass-spring are the most commonly used ones. FEM, being a continuum mechanics approach, gives more realistic simulation but also requires more processing. On the other hand, Mass-spring models represent the object as discrete springs between mass points. This results in a mesh of springs that is relatively less demanding computationally when it comes to its update at every time-step.

Besides these two methods, other hybrid approaches, like Tensor-mass models [31], also exist. Tensor-mass models discretize an object into tetrahedrons to define a set of corresponding local stiffness tensors. These tensors are stored in tetrahedron edges and their mass is stored in tetrahedron nodes as in lumped mass points.

SOFA provides implementation for all these methods along with different variations of them. The main challenge that we faced regarding the tissue modeling in SmartSIM is the selection of appropriate model for specific tissue along with the identification of correct parameters, like stiffness for mass-spring models, and Young modulus and Poisson's ratio for FEM. For simple deformable objects, like those used in grasping and fine dissection exercises, as well as for more complex objects, like liver, colon and uterus as simulated in procedural exercises, Tensor-mass models were used because of their stability as well as low computational requirements.

5.4 Topology Selection

We used Autodesk 3DS Max [32] and Blender [33] for creating graphical objects, which are in turn represented by topologies. The right choice of topology is an important factor in

keeping a simulation real-time. The collision surfaces, the mass-spring networks and the elements for FEM are all derived from the chosen topology in SOFA. However, being a multi-model framework, SOFA allows selection of different topologies for collision and behavioral modeling for the same simulation object. The two main types of topology in SOFA are the grid-based topology and the mesh-based topology. In the grid-based topology, an object can be modeled as a grid of equally-spaced points. Whereas, in the mesh-based topology, the points can be in any arrangement, and are read from an input file defining the mesh.

For complex curved shapes, like that of a liver, depicted in Figure 8d, a grid-based topology does not accurately model the surface. For such objects, the mesh-based topology is a better choice. However, to keep the simulation real-time, the mesh to be used for collisions and deformable behavior needs to be coarse. Hence, a mesh-based topology intelligently allows us to achieve a trade-off between accuracy and real-time performance in this case. On the other hand, collision surface and deformable behavior for simple structures, like the cylindrical object in Figure 8b, can be modeled accurately using a grid-based topology. Although such objects can also be modeled using a mesh-based topology with a coarse mesh defined in the file, yet the grid implementation in SOFA allows greater flexibility and facilitates achieving the same objective and hence is a better choice in this case. Although SOFA provides various topologies as described in [29], yet in SmartSIM, we have only used the tetrahedral topologies. This is due to the selection of Tensor Mass model for deformable objects (as mentioned in soft tissue section), which is designed to work with these topologies.

5.5 Exercise Development

SmartSIM supports ten basic exercises for training various aspects of laparoscopy, four exercises for training of general surgery procedures and three exercises for training of gynecology procedures. Modeling of each object in an exercise requires careful definition of

parameters for each component, e.g., Rayleigh damping coefficients for an Euler implicit solver or Poissons ratio and Young modulus for FEM. Once all the objects are modeled, components implementing collision algorithms are introduced in the scene graph to allow for appropriate interaction between objects.

The ten exercises included in SmartSIM are: (i) Instrument navigation to train the surgeons on basic handling of instruments in the laparoscopic environment (ii) Camera navigation to train the surgeons to handle the endoscope (iii) Coordination for training of handling an instrument in one hand with the endoscope in the other (iv) Grasping for improving hand-eye coordination and depth perception with the use of forceps (v) Lifting and grasping for training of extracting objects from underneath another object (vi) Fine dissection for dissection of capillary-like structures (vii) Cutting for training use of scissors (viii) Clipping to train the surgeons for using of clippers (ix) Peg transfer for training task-based extended exercises and (x) Speed exercise to teach the surgeons to act precisely within a confined time frame.

The four general surgery procedures included in SmartSIM are: (i) Camera anatomy training to train the surgeons in handling the endoscope and identifying different organs of anatomy, (ii) Appendectomy to teach the surgeons the skill of removing appendix from the body, (iii) Cholecystectomy to train the surgeons in the skill of dissecting cystic duct and cystic artery, applying clips and cutting the ducts and artery using the scissor instrument and (iv) Bladder Resection to teach a surgeon the skill of dissecting and removing gallbladder from the liver bed.

The three gynecology procedures included in SmartSIM are: (i) Camera anatomy training to teach the surgeons the skill of handling the endoscope and identifying female pelvic anatomy, (ii) Tubal Occlusion to train the surgeon in performing tubal occlusion by cutting the fallopian tubes and (iii) Salpingectomy to teach the surgeons the skill to extract ectopic pregnancy from the fallopian tube. A short video of some of these exercises can be viewed at [34]. Following are the details of basic and advanced exercises.

5.5.1 Basic Exercises

1. Instrument Navigation

This exercise is designed to teach the skill of handling laparoscopic instruments. The user is presented with different shapes on the floor. The user is required to touch a highlighted pattern by navigating the highlighted instrument till the object is no more coloured(Figure 9a). Performance is evaluated on the basis of time and economy of movement, collision of instrument with floor and errors due to use of instrument other than the highlighted one.

2. Camera Navigation

The objective of this exercise is to practice the skill of handling an endoscope. The user is presented with an object on the floor. The is required to navigate an endoscope towards the object and focus the camera on it till the object is highlighted(Figure 9b). Performance is evaluated on the basis of time and economy of movement and errors due to collision of endoscope with the floor.

3. Coordination

Coordination exercise is designed to practice the skill of handling an instrument and an endoscope simultaneously. The user is presented with an object on the floor. The user is required to navigate an endoscope towards the object and focus the camera on it till the object is highlighted and then touch the object with the instrument(Figure 9c). Performance is evaluated on the basis of time and economy of movement and errors due to collision of endoscope with the floor.

4. Cutting

To practice the skill of cutting an object with laparoscopic scissors, the user is presented with a cylinder-shaped object, which is fixed on its ends close to the floor. The user is required to grasp the object from one end and cut it from the other end

at a specified location(Figure 9d). Performance is evaluated on the basis of time and economy of movement, collision of instruments with floor and errors due to cutting at the wrong location.

5. **Clipping**

To practice the skill of clip application, the user is presented with a cylinder-shaped object fixed on its ends close to the floor. The user is require to grasp the object from one end using forceps and apply a clip at a specified location(Figure 9e).

Performance is evaluated on the basis of time and economy of movement, collision of instruments with floor and errors due to misplaced or dropped clips.

6. **Fine Dissection**

To teach the skill of safely removing pieces surrounding a vessel structure, the user is presented with a broad tissue-like structure that covers two fine vessels underneath.

The user is required to stretch the center structure and cauterize the fine vessels underneath using L-hook(Figure 8c). Performance is evaluated on the basis of time and economy ot movement, collision of instruments with floor and errors due to burning of center tissue structure.

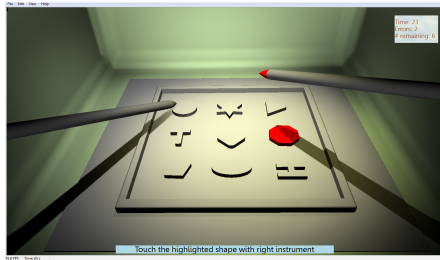
7. **Grasping**

This exercise is designed to practice the skill of grasping an object using laparoscopic forceps. The user is presented with a cylinder-shaped object fixed to the floor and is required to grasp the object and move it to a basket placed nearby(Figure 8b).

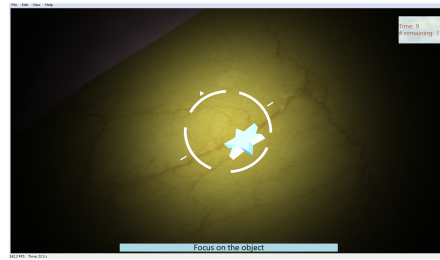
Performance is evaluated on the basis of time and economy of movement, collision of instruments with floor and errors due to wrong use of instrument.

8. **Lifting and Grasping**

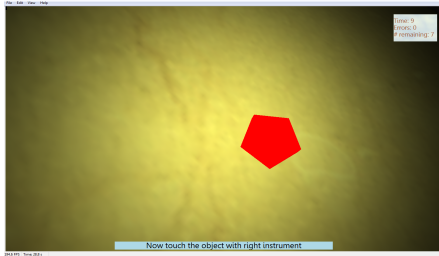
The objective of the exercise is to practice the skill of lifting an object with one instrument and grasping another object with the instrument simultaneously. The



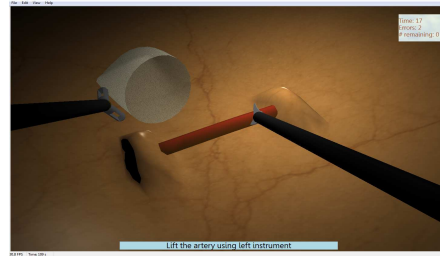
(a) Instrument Navigation: laparoscopic instrument handling.



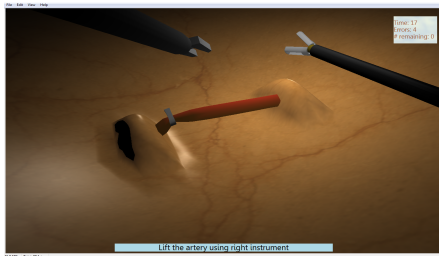
(b) Camera Navigation: endoscope navigation.



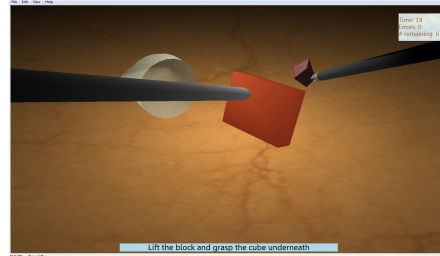
(c) Coordination: simultaneous instrument and endoscope handling.



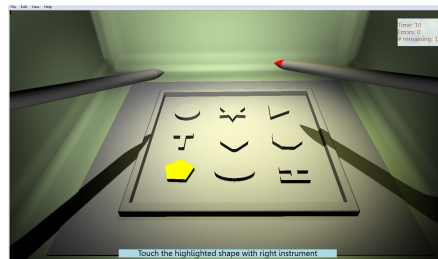
(d) Cutting: object cutting with scissors.



(e) Clipping: clip application at specified location.



(f) Lifting and Grasping: simultaneously lifting an object and grasping the other one.



(g) Precision and Speed: timely and precise handling of instruments.

Figure 9: SmartSIM: Basic Exercises

user is presented with a hinged cube on the floor. The user is required to lift the cube and move the object placed underneath the cube to a basket(Figure 9f). Performance

is evaluated on the basis of time and economy of movement, collision of instruments with floor and errors due to wrong use of instrument.

9. Peg Transfer

To improve the hand-eye coordination, the user is required to move four coloured disks from pegs on the left side to those on the right side and back to those on the left in a specified order of colour(Figure 8a). Performance is evaluated on the basis of time and economy of movement, collision of endoscope with floor and errors due to placing disks in mismatching pegs.

10. Precision and Speed

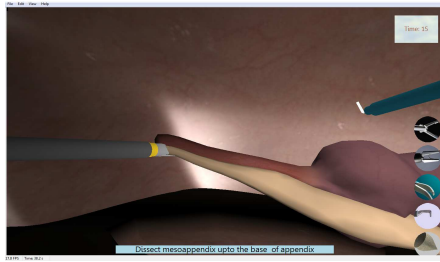
To practice the skill of handling laparoscopic instruments timely and precisely, the user is presented with different shaped objects on the floor and is required to touch the highlighted object by navigating the highlighted instrument till the object is no more coloured, in a given amount of time(Figure 9g). Performance is evaluated on the basis of time and economy of movement, collision of instruments with floor and errors due to failing to touch the required shape in time.

5.5.2 General Surgery Exercises

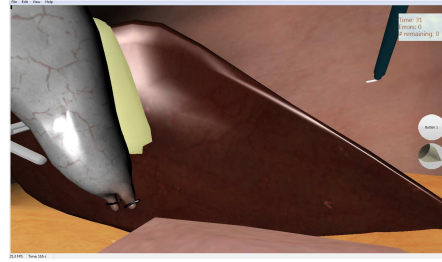
1. Appendectomy

This exercise teaches the basics of performing the laparoscopic appendectomy procedure. The user is presented with a scene showing a part of the large intestine along with the appendix and is required to remove the tissue around the intestine region, followed by clipping and cutting the appendix at the right region(Figure 10a). Performance is evaluated on the basis of time and economy of movement and errors due to incomplete burning, wrong clipping and faulty cutting.

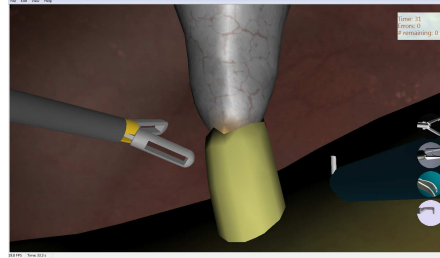
2. Bladder Separation



(a) Appendectomy



(b) Bladder Separation



(c) Cholecystectomy

Figure 10: SmartSIM: General Surgery Exercises

This exercise teaches the basics of separating the gall-bladder from the liver bed. The user is presented with a scene showing a part of the gall-bladder and liver and is required to remove the tissue attaching the gall-bladder with the liver bed(Figure 10b).

3. Camera Navigatioon of Abdominal Anatomy

This exercise is designed to practice the skill of handling an endoscope and identifying specific anatomical structures in the abdomen. The user is required to focus the camera on certain areas in the upper right quadrant of the body(Figure 8d). Performance is evaluated on the basis of time and economy of movement and errors due to collision of endoscope with floor.

4. Cholecystectomy

This exercise is designed to learn the basics of performing the laparoscopic cholecystectomy procedure. The user is presented with a scene showing the lower part of the gall-bladder and is required to remove the tissue around the gall-bladder,

followed by clipping and cutting the cystic duct and artery at the right regions(Figure 10c). Performance is evaluated on the basis of time and economy of movement and errors due to incomplete burning, wrong clipping and faulty cutting.

5.5.3 Gynaecology Exercises

1. Camera Navigation of Pelvic Anatomy

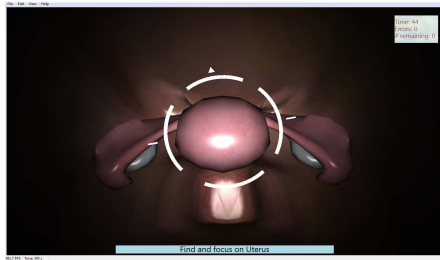
The objective of this exercise is to teach the user the skill of handling an endoscope and identifying specific anatomical structures located in the abdomen. The user is required to focus the camera at specified female pelvic region of the body(Figure 11a). Performance is evaluated on the basis of time and economy of movement and errors due to collision of endoscope with floor.

2. Salpingectomy

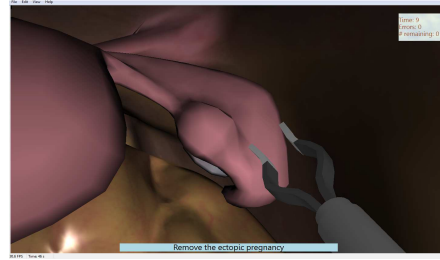
This exercise is designed to learn the basics of the laparoscopic salpingectomy procedure. The user is presented with a scene showing one side of the uterus and one fallopian tube and is required to remove the part of the tube with ectopic pregnancy using bipolar forceps(Figure 11b). Performance is evaluated on the basis of time and economy of movement and errors due to burning of wrong areas.

3. Tubal Ligation

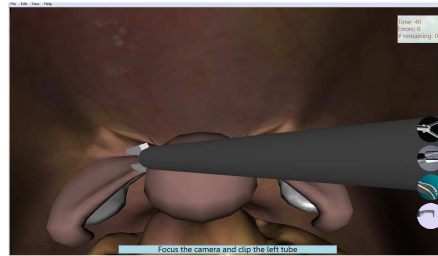
This exercise is designed to teach the basics of laparoscopic tubal ligation. The user is required to focus on the uterus and the tube using the endoscope as the left instrument and clip the tube using the clipper as the right instrument(Figure 11c). Performance is evaluated on the basis of time and economy of movement and errors due to wrong clipping.



(a) Camera navigation for pelvic anatomy



(b) Salpingectomy



(c) Tubal Ligation

Figure 11: SmartSIM: Gynaecology Exercises

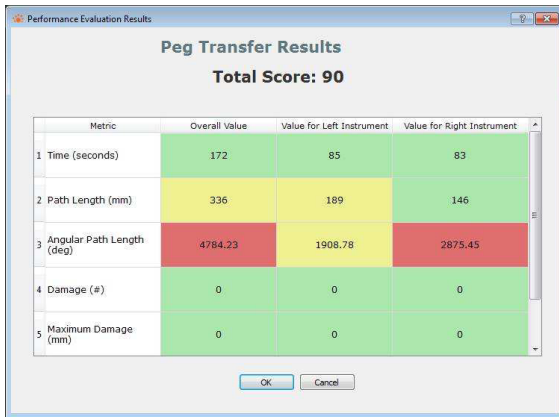
5.6 Inclusion of Visual Effects

Another advantage of a VR simulation is that it can display textual instructions and visual cues to guide the trainee. Such features are usually scenario-specific, and hence cannot be a part of a generic engine. In SmartSIM, we have included both of these features by adding specific functionalities in SOFA.

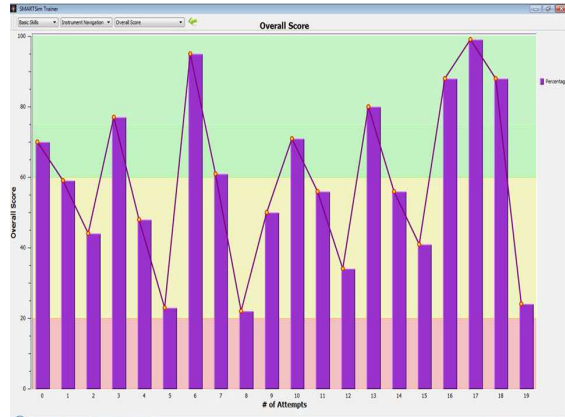
In our system, we have included visual effects like highlighting of the objects that should not be touched by the tool otherwise, such as the floor in the grasping exercise (Figure 8b), and cues like changing of color of the pegs when a disk is potted in the right peg, in the peg transfer exercise (Figure 8a). Inclusion of such features requires extraction of specific collision contacts from among the list of all contacts, maintained by a contact manager in SOFA.

5.7 Performance Evaluation

Performance evaluation is an important feature of all VR surgical simulators that distinguishes it from the conventional box trainers. It helps to provide a thorough assessment of a performer's skills that is missing in training through box trainers. Special plug-ins were added in SOFA for evaluating metrics for each exercise in SmartSIM. For instance, Figure 12a shows the performance metric generated after the peg transfer exercise. There are two types of metrics implemented in SmartSIM. The first type gives an objective assessment. Such metrics include commonly used metrics, such as task completion time, instrument path lengths and errors made while conducting the exercise tasks. The other type gives an assessment in comparison to the skills of the expert surgeons. These novel metrics are introduced so that the simulator can act as an independent learning tool and thus we can remove the requirement of direct supervision from expert surgeons. For such an assessment, there are two options introduced in the simulator to identify whether the exercise is performed by a novice or an expert surgeon. In case of an expert performance, the simulator is said to be in the *learning* mode. In this mode, the simulator only records the instrument path followed by the expert surgeon and uses it to create a benchmark [35]. If the exercise is performed by a novice, then the simulator is said to be in the *evaluation* mode. In this mode, the simulator records the path followed and calculates its deviation from the expert surgeons' paths. To grade the user on the basis of his expertise, the 'Performance Evaluation Results' window, shown in Figure 12a, displays scores of metrics critical in evaluating the performance of the user. The metrics are (i) time taken to complete the exercise, (ii) path length, (iii) angular path length, (iv) number of damages (penalties) and (v) maximum amount of damage. Results are displayed separately as well as collectively for the left and right hand instrument to provide better understanding of hand co-ordination. The score blocks are color-coded to differentiate between good, acceptable, and poor performance relative to the benchmark set in the learning mode (construct validity). The green block indicates good performance, yellow block indicates



(a) Peg transfer exercise



(b) Instrument navigation

Figure 12: Evaluation metric and performance graph

that slight improvement is needed, and red block indicates poor performance in a particular metric. Any recorded deviation is penalized. Such an evaluation is found to be extremely useful in training since it provides an additional constraint on the trainee for better maneuvering the instruments without accessing unwanted regions. SmartSIM also keeps record of the performance of every user, for longterm evaluation. It generates graphs of the performance history (figure 12b), where the overall score (total score) for multiple attempts is plotted to evaluate the improvement in particular exercise over time.

5.8 Database

SmartSIM maintains a local MySQL based database at the operating computer. The database is very simple with the main task of maintaining record of users and their passwords along with their respective performances for each exercise. SOFA does not offer any default database module and thus integrating the database system with the SOFA is one of the contributions of the reported work.

6 Conclusions

MIS is gradually replacing traditional surgical methods as it is safer, quicker, with less risk involved and results in quick patient recovery. In order to learn the special psycho motor skills for MIS, virtual reality based simulators are required that enable a surgeon to practice outside the Operation Theater.

The paper mainly shared the system-level perspective of SmartSIM, i.e., a novel virtual reality simulator for basic training in MIS. The distinguishing features of SmartSIM include its modular design approach using a generic physics engine by modifying it to overcome its limitations such as avoiding extra computational loads in *Simulation Object Categorization*, replacing the mouse cursor with instruments for realistic view in *Instrument Simulation* and a trade-off between accuracy and real-time simulations in *Soft Tissue Modeling* and *Topology Selection* for each exercise. Also, the simulator is flexible and has the potential to be updated. If, for instance, someone wants to interface a new custom device, he would only need to modify the Interface Device Handler component of the system. Similarly, if there is a need to change the database then only the database needs to be updated with a script along with the modifications in DB Handler. The ease with which the simulator can be updated with plug-ins is due to the open source nature of SOFA. Similarly, more efficient algorithms can also be added by compiling the corresponding plug-in with the existing library and use the compiled plug-in in the current system. The plug-in can be added using SOFA's plug-in manager after which the user only needs to change the tag in exercise's xml file with name and attributes of the new plug-in. A new model can also be added in the system by carefully handling these modules individually, giving user the flexibility to add techniques or medical procedures. SmartSIM also includes novel training evaluation metrics that facilitate in supervised as well as unsupervised training of surgeons for basic and some advanced (general surgery and gynecology procedures) skills of laparoscopy. Moreover, the paper also describes an easy-to-fabricate custom-designed mechanical interface, which acts as the main input device for the simulator. The custom

fabrication of mechanical interface of the simulator costed us around \$2000. Presently, the system does not support force feedback but we are working on the introduction of a custom-built haptic module in the mechanical interface and development of more basic and advanced exercises such as suturing and knotting. Also, a set of experiments is to be designed to further investigate the construct validity of the system.

We have validated our system by using it in MIS training workshops organized by a team of surgeons at a local hospital. These workshops were based on box trainers, commercial simulators, like LapSim, and SmartSIM. It was found that, in comparison with box trainers and commercial simulators, the SmartSIM renders higher level of usefulness for MIS skills training, greater degree of realism to actual surgical scenarios, better quality of graphics, and improved overall effectiveness of exercises. Further advantages of the SmartSIM include reduced training costs, the ability to practice unusual and rare medical scenarios, objective and quick assessment of laparoscopic surgical skills and ease of mobility of the system.

Acknowledgment

The authors wish to thank Dr. Asif Zafar Malik and his team at Holy Family Hospital, Pakistan, for their support in this research. This work was supported by a research grant (number ICTRDF/TR&D/2009/26) by National ICT R&D Fund Pakistan.

References

- [1] Gallagher A, McClure N, McGuigan J, Ritchie K, Sheehy N. An ergonomic analysis of the fulcrum effect in the acquisition of endoscopic skills. *Endoscopy*. PubMed PMID: 9826140. Sep, 1998;.
- [2] Crother I, Gallagher A, McClure N, James D, McGuigan J. Experienced Laparoscopic Surgeons are Automated to the Fulcrum Effect: An Ergonomic Demonstration.

- Endoscopy. 1999;31:365–369.
- [3] Ahlberg G, Heikkinen T, Iselius L, Leijonmarck C E, Rutqvist J, Arvidsson D. Does training in a virtual reality simulator improve surgical performance? *Surgical Endoscopy And Other Interventional Techniques*. 2002;16:126–129.
- [4] Neal E Seymour SARMKOVKBDKARMS Anthony G Gallagher. Virtual Reality Training Improves Operating Room Performance: Results of a Randomized, Double-Blinded Study. *Annals of Surgery*. 2002;236:458–464.
- [5] Aggarwal R, Ward J, Balasundaram I, Sains P, Athanasiou T, Darzi A. Proving the effectiveness of virtual reality simulation for training in laparoscopic surgery. *ANNALS OF SURGERY*. 2007;246:771–779. Available from:
<http://dx.doi.org/10.1097/SLA.0b013e3180f61b09>.
- [6] Surgical-Science-Co. LAPSim; 2001. [Online; accessed 2016].
<http://www.surgical-science.com/lapsim-the-proven-training-system/>.
- [7] Lacey G, Ryan D, Cassidy D, Young D. Mixed-Reality Simulation of Minimally Invasive Surgeries. *IEEE MultiMedia*. 2007 Oct;14(4):76–87.
- [8] M S Wilson CSRS A Middlebrook, McCloy RF. MIST VR: a virtual reality trainer for laparoscopic surgery assesses performance. *Annals of The Royal College of Surgeons of England*. 1997 November;p. 403404.
- [9] Symbionix-Co. LAP Mentor; 2005.
<http://symbionix.com/simulators/lap-mentor/>.
- [10] Immersion-Co. LapVR;. [Online; accessed 2016]. <http://www.immersion.com>.
- [11] Wentink M, Post J. SIMENDO; 2016. <http://www.simendo.eu>.

- [12] Cakmak H, Maass H, Khnapfel U. VSOne, a virtual reality simulator for laparoscopic surgery. *Minimally Invasive Therapy & Allied Technologies*. 2005;14(3):134–144. Available from: <http://dx.doi.org/10.1080/13645700510033958>.
- [13] Maciel A, Liu Y, Ahn W, Singh TP, Dunnican W, De S. Development of the VBLaST: a virtual basic laparoscopic skill trainer. *The International Journal of Medical Robotics and Computer Assisted Surgery*. 2008;4(2):131–138. Available from: <http://dx.doi.org/10.1002/rcs.185>.
- [14] Carlos Monserrat UMMACJVG Oscar Lpez. In: *GeRTiSS: A Generic Multi-model Surgery Simulator*. vol. 2673 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg; 2003. p. 59–66.
- [15] Tendick F, Downes M, Goktekin T, Cavusoglu MC, Feygin D, Wu X, et al. A Virtual Environment Testbed for Training Laparoscopic Surgical Skills. *Presence*. 2000 June;9(3):236–255.
- [16] Montgomery K, Bruyns C, Brown J, Sorkin S, Mazzella F, Thonier G, et al. Spring: A General Framework for Collaborative, Real-time Surgical Simulation. In: *Medicine Meets Virtual Reality (MMVR 2002)*. IOS Press; 2002. p. 23–26.
- [17] Cavusoglu MC, Goktekin T, Tendick F. GiPSi: A Framework for Open Source/Open Architecture Software Development for Organ-Level Surgical Simulation. *IEEE Transactions on Information Technology in Biomedicine*. 2006;10:312–322. Available from: <http://dblp.uni-trier.de/db/journals/titb/titb10.html#CavusogluGT06>.
- [18] Kuroda Y, Nakao M, Kuroda T, Oyama H, Yoshihara H. Design and implementation of MVL: medical VR simulation library. In: *32. International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2005, Los Angeles, California, USA,*

July 31 - August 4, 2005, Posters; 2005. p. 61. Available from:

<http://doi.acm.org/10.1145/1186954.1187023>.

- [19] Bacon J, Tardella N, Pratt J, ENGLISH JHJ. The Surgical Simulation and Training Based Language for Medical Simulation. *Medicine Meets Virtual Reality 14: Accelerating Change in Healthcare: Next Medical Toolkit*. 2006;119:37.
- [20] Machado LS, Moraes RM, Souza DFL, Souza LC, Cunha ILL. A framework for development of virtual reality-based training simulators. *Studies in health technology and informatics*. 2009;142:174176. Available from:
<http://europepmc.org/abstract/MED/19377142>.
- [21] Allard J, Cotin S, Faure F, j Bensoussan P, Poyer F, Duriez C, et al. SOFA an Open Source Framework for Medical Simulation. In: *In Medicine Meets Virtual Reality (MMVR 15; 2007. .*
- [22] Kim Y, Kim K, Roy F, Park S. In: Dohi T, Liao H, editors. *Development of a Laparoscopic Surgical Training System with Simulation Open Framework Architecture (SOFA)*. vol. 3 of *Proceedings in Information and Communications Technology*. Springer Japan; 2012. p. 83–91. Available from:
http://dx.doi.org/10.1007/978-4-431-54094-6_10.
- [23] Mentice. Mentice AB; 1999. [Online; accessed 2016]. <http://www.mentice.com>.
- [24] Rosen J, Brown JD, Chang L, Barreca M, Sinanan M, Hannaford B. The BlueDRAGON - a system for measuring the kinematics and dynamics of minimally invasive surgical tools in-vivo. In: *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*. vol. 2; 2002. p. 1876–1881 vol.2.
- [25] Lum MJH, Friedman DCW, Sankaranarayanan G, King H, Fodero K, Leuschke R, et al. The RAVEN: Design and Validation of a Telesurgery System. *The International Journal of Robotics Research*. Sep, 2009;28, no. 9:1183–1197.

- [26] Riviere CN, Rader RS, Khosla PK. Characteristics of hand motion of eye surgeons. In: Engineering in Medicine and Biology Society, 1997. Proceedings of the 19th Annual International Conference of the IEEE. vol. 4; 1997. p. 1690–1693 vol.4.
- [27] M Kawasaki NKYKMNTKHY M Rissanen. VRASS (Virtual Reality Aided Simulation); 2006. [Online; accessed 2016].
www.kuhp.kyoto-u.ac.jp/~mi/en/index.php.
- [28] Faure F, Duriez C, Delingette H, Allard J, Gilles B, Marchesseau S, et al. SOFA: A Multi-Model Framework for Interactive Physical Simulation. In: Payan Y, editor. Soft Tissue Biomechanical Modeling for Computer Assisted Surgery. vol. 11 of Studies in Mechanobiology, Tissue Engineering and Biomaterials. Springer; 2012. p. 283–321. Available from: <https://hal.inria.fr/hal-00681539>.
- [29] André B, Delingette H. Versatile Design of Changing Mesh Topologies for Surgery Simulation. In: Proceedings of the 4th International Symposium on Biomedical Simulation. ISBMS '08. Berlin, Heidelberg: Springer-Verlag; 2008. p. 147–156. Available from: http://dx.doi.org/10.1007/978-3-540-70521-5_16.
- [30] Delingette H, Ayache N. Soft Tissue Modeling for Surgery Simulation. In: Computational Models for the Human Body. vol. 12 of Handbook of Numerical Analysis. Elsevier; 2004. p. 453 – 550. Available from:
<http://www.sciencedirect.com/science/article/pii/S1570865903120054>.
- [31] Cotin S, Delingette H, Ayache N. A hybrid elastic model for real-time cutting, deformations, and force feedback for surgery training and simulation. The Visual Computer. 2000;16(7):437–452. Available from:
<http://dblp.uni-trier.de/db/journals/vc/vc16.html#CotinDA00>.
- [32] Autodesk-Inc. 3DSmax: 3D Modeling, Animation, and Rendering Software; 1990. [Online; accessed 2016]. <http://www.autodesk.com/products/3ds-max/overview>.

- [33] Roosendaal T. Blender; 2002. [Online; accessed 2016]. <https://www.blender.org>.
- [34] SMART-SurgiSol. SmartSIM: A Virtual Reality Simulator for Laparoscopy Training using a Generic Physics Engine; 2016. [Online; accessed 2016]. <http://smartsurgisol.seecs.nust.edu.pk/smart-sim/>.
- [35] Ahmad MA, Mansoor SB, Khan ZA, Aqeel WA, Kabir SH. Benchmarking Expert Surgeons' Path for Evaluating a Trainee Surgeon's Performance. In: Proceedings of the 12th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry. VRCAI '13. New York, NY, USA: ACM; 2013. p. 57–62. Available from: <http://doi.acm.org/10.1145/2534329.2534345>.