Georgian Technical University

Donghak Kim

Coronary Angiography Learning System

Submitted for the academic degree of Doctor of Engineering

Doctoral Program “Biomedical Engineering”

Code: 04

Georgian Technical University

Tbilisi, 0175, Georgia

July, 2018

Copyright © 2018, Donghak Kim

Tbilisi

2018

The thesis was developed at the Department of Biomedical Engineering of the Faculty of Informatics and Control Systems of Georgian Technical University

Academic Supervisor: Prof. Irine Gotsiridze

Reviewers: Prof. Paata Kervalishvili

Prof. Alexandre Aladashvili

The thesis defense will take place at the session of the board of the dissertation council of the Faculty of Informatics and Control Systems of Georgian Technical University, on July 20, 2018 at 16:00.

Block IV, Auditorium 318

Address: 77 Kostava, Tbilisi, 0175

The thesis is available at the GTU library,

the abstract - on the Faculty webpage.

The Secretary of the Dissertation Council: Professor, Tinatin Kaishauri

Georgian Technical University

The Faculty of Informatics and Control Systems

We, undersigned, confirm that we familiarized ourselves with the thesis by Donghak Kim on “Coronary Angiography Learning System” and recommend it for reviewing by the dissertation council of the Faculty of Informatics and Control Systems of Georgian Technical University as a doctorate thesis.

Date .....

Academic Supervisor: Prof. Irine Gotsiridze

Review: Prof. Paata Kervalishvili

Review: Prof. Alexandre Aladashvili

Georgian Technical University

Author: Donghak Kim

Title: Coronary Angiography Learning System

Faculty: Informatics and Control Systems

Degree: Doctor of Engineering

The session held on: .....

In case of requests by individuals or institutions to get familiarized with the thesis titled as indicated above the right for its non-commercial replication and dissemination rests with Georgian Technical University.

Author's signature

The author retains all other copyrights and reprinting or reproducing in any other form of the entire thesis or its individual components is prohibited without a prior written permission of the author.

The author certifies that copyrighted materials are lawfully used in the original thesis (except for small quotations which require only specific references to the cited literature which is a widely accepted practice in thesis development) and assumes responsibility for all of them.

Abstract

This paper presents coronary angiography learning system that is designed and built for coronary angiography training. This learning system is based on simulation training to prepare healthcare professionals for coronary angiography, providing virtual hands-on experience outside a catherization laboratory. To provide this experience to a wide range of people, it is designed to be simple and affordable.

This system has three components: display, pc, and keyboard. Display provide visual feedbacks. A keyboard is an interface between a trainee and the system. PC process the graphical calculations and execution of algorithms for the simulations according to the inputs from a trainee.

The simulation is the result of the interactions of the following components: imaging, instrument model (guidewire and catheter,) anatomy model (heart, skeleton, coronary arteries), and contrast dye. These four components are integrated in the system.

Anatomy model was built with a software, called Blender (v 2.79). It is a popular free open-source tool for computer graphic design. The software helps to modify and build a model. The model of coronary arteries was created with the imported heart model.

The simulation of the guidewire and catheter was implemented with a software, UNITY, that is a multi-platform game engine. It includes physics engine that makes the simulation seem real. The guidewire and catheter were modelled after the mass-spring model.

Imaging is regarding angiographic views. As angiographic views depend on the position of C-arm, the imaging component controls the C-arm's position and provides desired images through panning and zooming.

Contrast dye makes the coronary arteries visible under X-ray. This component is related stenosis. It creates coronary arteries with stenosis based on the two parameters: number of segment and the level of severity.

The coronary angiography is a procedure to visualize coronary arteries and locate abnormalities such as stenosis. The system starts with catherization introducing a guide wire and catheter into the aortic root. The next step is cannulation. Once the catheter is positioned, it is manipulated to be engaged in an ostium. The manipulation of the catheter is done through keyboard inputs. It includes forward and backward movement, and clockwise and counterclockwise rotation.

Once the cannulation is done, contrast dye is injected. The injection is activated through the input from the keyboard. As the contrast dye is injected, the coronary arteries appear on the display. For the better view of coronary arteries, the x-ray angiographic images can be panned up, down, left or right and zoomed in and out while the contrast dye is being injected. The learning system will store the coronary angiography data for a review later.

This coronary angiography learning system can be a tool to expose the procedure of catherization and cannulation to a trainee. It also helps to locate any abnormal artery with stenosis.

This learning system provides visual feedback only. For the better experience of coronary angiography, the learning system is designed for serial communication through USB. It can be readily connected with an external module such as a haptic device for tactile feedback.

Simulation training is suitable for interventional cardiology, especially coronary angiography since certain number of procedures is required to reach a certain level of competency. Through the simulation training with this learning system, without real patients, the level of competency may increase quickly. Consequently, it will shorten the time and the cost of training.

It can be also an educational tool for medical students to study coronary arteries and learn to identify pathological condition of coronary arteries and for educators to prepare teaching materials.

ავტორეფერატი

ეს ნაშრომი წარმოგვიდგენს კორონარული ანგიოგრაფიის სასწავლო სისტემას, რომელიც შემუშავდა და შეიქმნა კორონარულ ანგიოგრაფიაში ტრენინგის მიზნებისათვის. ეს სასწავლო სისტემა ეფუძნება სიმულაციურ სწავლებას კორონარულ ანგიოგრაფიაში სამედიცინო სპეციალისტების მზადებისა და მათთვის ვირტუალურად, კათეტერიზაციის ლაბორატორიის ფარგლებს მიღმა, პრაქტიკული გამოცდილების მიცემას. ამ გამოცდილების ადამიანთა ფართო სპექტრისათვის მიცემას უზრუნველყოფს სისტემის მარტივი დიზაინი და ადვილი ხელმისაწვდომობა. სისტემა შედგება სამი კომპონენტისაგან, ესენია: მონიტორის ეკრანი, პერსონალური კომპიუტერი და კლავიატურა. მონიტორის ეკრანით ხდება ვიზუალური უკუკავშირის მიღება. კლავიატურით მყარდება კავშირი პრაქტიკანტსა და სისტემას შორის. პერსონალური კომპიუტერით ხდება გრაფიკული მონაცემების დამუშავება და სიმულაციის ალგორითმების შესრულება პრაქტიკანტის მიერ განსაზღვრული შესავალი მონაცემების საფუძველზე.

სიმულაცია არის შემდეგი კომპონენტების ურთიერთქმედების შედეგი: ვიზუალიზაცია, ინსტრუმენტების მოდელი (ლითონის გამტარი და კათეტერი), ანატომიური მოდელი (გული, ჩონჩხი, კორონარული არტერიები) და კონტრასტული ნივთიერება. ეს ოთხი კომპონენტი ინტეგრირებულია ერთიან სისტემაში.

ანატომიური მოდელი შექმნილია Blender (v2.79) კომპიუტერული პროგრამის გამოყენებით. იგი წარმოადგენს კომპიუტერული გრაფიკული დიზაინის პოპულარულ, ღია კოდზე დაფუძნებულ საპროგრამო გადაწყვეტას. იგი გვეხმარება მოდელის მოდიფიკაციასა და დახვეწაში. კორონარული არტერიების მოდელი შეიქმნა გულის მოდელის გამოყენებით, რომელიც საპროგრამო პაკეტში იქნა იმპორტირებული.

ლითონის გამტარისა და კათეტერის სიმულაციისთვის გამოყენებულ იქნა კომპიუტერული პროგრამა - UNITY, რომელიც მულტიპლატფორმული თამაშის ძრავის წარმოადგენს. იგი მოიცავს ფიზიკურ ძრავს, რომელიც „სიმულაციას რეალობასთან აახლოებს. ლითონის გამტარისა და კათეტერის მოდელირება სისტემის - „ზამბარა-ტვირთი“ მოდელირების ანალოგიურია.

ვიზუალიზაცია გულისხმობს ანგიოგრაფიული გამოსახულებების მიღებას. იმდენად, რამდენადაც ანგიოგრაფიული გამოსახულებები დამოკიდებულია რენტგენოლოგიური აპარატის C-რკალის მდებარეობაზე, ვიზუალიზაციის კომპონენტი აკონტროლებს C-რკალის მდებარეობას და უზრუნველყოფს სასურველი გამოსახულებების მიღებას პანორამული ან ახლო ხედის გადაღების მეშვეობით.

კონტრასტული ნივთიერება ახდენს კორონალური არტერიების რენტგენოლოგიურ ვიზუალიზაციას. ეს კომპონენტი კავშირშია სტენოზის

მოდელირებასთან. იგი ახდენს სტენოზირებული კორონარული არტერიების ფორმირებას ორი პარამეტრის - სტენოზირებული უბნების რაოდენობისა და სტენოზის ხარისხის, გათვალისწინებით.

კორონარული ანგიოგრაფია წარმოადგენს პროცედურას, რომლის დროსაც ხდება კორონარული არტერიების ვიზუალიზაცია და პათოლოგიური უბნების, მაგალითად - სტენოზის, ადგილმდებარეობის გამოვლენა. სისტემა მოქმედებაში მოდის კათეტერიზაციის ეტაპით, რომლის მსვლელობაში ხდება ლითონის გამტარისა და კათეტერის შეყვანა აორტის ფესვში. შემდეგი ეტაპია კანულაცია. კათეტერის შეყვანის შემდეგ ხდება მისით მანიპულირება ისე, რომ იგი აღმოჩნდეს კორონარული არტერიის შესართავში. მანიპულირება ხორციელდება კლავიატურის მეშვეობით. ეს მოიცავს კათეტერის მოძრაობას წინა და უკანა მიმართულებით, ასევე - მის მობრუნებას საათის ისრის მოძრაობის მიმართულებით ან მის საწინააღმდეგოდ.

კანულაციის დასრულების შემდგომ ხდება რენტგენოკონტრასტული ნივთიერების შეყვანა. ინექციის ფუნქციის აქტივაცია ხდება კლავიატურიდან მიცემული ბრძანებით. კონტრასტული ნივთიერების შეყვანის პარალელურად ხდება მონიტორის ეკრანზე კორონარული არტერიების ვიზუალიზაცია. კორონარული არტერიების უკეთ დათვალიერებისათვის შესაძლებელია რენტგენოკონტრასტული ნივთიერების შეყვანის პარალელურად რენტგენოლოგიური ანგიოგრაფიული გამოსახულებების პანორამული ხედის ზემოთ, ქვემოთ, მარცხნივ და მარჯვნივ გადაადგილება, ასევე - გამოსახულების გადიდება და დაპატარავება. სასწავლო სისტემა შეინახავს მონაცემებს კორონარული ანგიოგრაფიის შესახებ მოგვიანებით ეტაპზე განხილვისათვის.

კორონარული ანგიოგრაფიის სასწავლო სისტემა შესაძლოა იყოს ინსტრუმენტი, რომლის მეშვეობითაც პრაქტიკანტი გაეცნობა კათეტერიზაციისა და კანულაციის პროცედურას. იგი ასევე გვეხმარება, გამოვავლინოთ პათოლოგიურად შეცვლილი არტერია სტენოზის უბნებით.

ეს სასწავლო სისტემა გათვლილია მხოლოდ ვიზუალური უკუკავშირის მიღებაზე. კორონარულ ანგიოგრაფიაში გამოცდილების მიღების გაუმჯობესების მიზნით სასწავლო სისტემაში გათვალისწინებულია სერიული კომუნიკაციის ფუნქცია უნივერსალური სერიული პორტის (USB) მეშვეობით. ტაქტილური უკუკავშირის მისაღებად სისტემასთან ადვილად არის შესაძლებელი გარეგანი მოდულის, მაგალითად - ჰაპტური მოწყობილობის, მიერთება.

სიმულაციური ტრენინგი გამოსადეგია ინტერვენციული კარდიოლოგიის, განსაკუთრებით კორონარული ანგიოგრაფიის სფეროში ვინაიდან კომპეტენციის განსაზღვრული დონის მისაღწევად საჭიროა პროცედურების გარკვეული რაოდენობის ჩატარება. ამ სასწავლო სისტემით სიმულაციური ტრენინგი იძლევა საშუალებას, მოხდეს კომპეტენციის

დონის სწრაფი ამაღლება რეალურ პაციენტებთან მუშაობის გარეშე. შედეგად, მცირდება ტრენინგის ხანგრძლივობა და მისი ღირებულება.

მისი გამოყენება ასევე შეუძლიათ მედიკოს-სტუდენტებს კორონარული არტერიებისა და მათი პათოლოგიების შესახებ ცოდნის მისაღებად და პედაგოგებს სასწავლო მასალების მომზადებისათვის.

Contents

Tables

List of Figures

List of Abbreviation

AP	Anterior Posterior
CA	Coronary Angiography
LAD	Left Anterior Descending
LAO	Left Anterior Oblique
LCA	Left Coronary Artery
LCx	Left Circumflex
LMS	Left Main Stem
RAO	Right Anterior Oblique
RCA	Right Coronary Artery
PDA	Posterior Descending Artery

Acknowledgement

"For God so loved the world, that he gave his only Son, that whoever believes in him should not perish but have eternal life."
John 3:16 (Bible)

First and foremost, I would like to thank God for His love toward me through Jesus Christ. It was His love that let me enjoy my life abundantly.

I would like to thank my advisor, Irina Gotsiridze, for guiding, supporting and inspiring me. I would like to thank Prof. Zviad Gurtskaia for technical advices and thank Prof. Mariam Tsiklauri for your practical help with writing this dissertation and many other matters that I do not even know.

I would like to thank Prof. Alexandre Aladashvili for letting me observe all the coronary angiography procedures in your cath lab.

I would like to thank my NBC family. God blessed me so much through the NBC family. Thank you for your love and prayers!

Last, but not least, I would like to thank my parents and grandmother for encouraging me with your unceasing love and prayers. I would like to thank my wife, Joohyun, for her presence in my life.

Introduction

Coronary Angiography (CA) is a procedure that shows the insides of coronary arteries using contrast dye and special x rays. It is one of the most common invasive procedures in medicine exceeding 1.8 million procedures in 2001 in Europe [1]. CA is one of the most common invasive procedures in medicine and a relatively safe procedure.

In order to reach the competent level of the procedure, one must perform the procedures over and over. The training was mostly done in a catherization laboratory. It takes certain time to achieve the competency in coronary angiography. To someone who just started getting the training, it could be daunting experience, which may cause medical complications. Therefore, the training outside a cath lab got attention in the medical training and its advantages were recognized. The advancements of computer technology brought the applications of the technology to the medical field. One of the applications is simulation training.

Figure 1. Coronary Angiography Scene [2]

I. Literature Review

Chapter 1. Simulation History and Applications in Medical Field

1. Simulation Learning

1.1 History of Simulation

Simulation has been used to train professionals in their field. Especially, for the flight simulation, its history goes back to 1910. In the year, 1910, one of the first flight simulator was introduced. It was in 1903 that the Wright Brothers flew their first flight. The simulation training was adopted so quickly to train pilots. Since then, flight simulators were developed and used to train pilots as fast and more sophisticated planes were developed. The pilots had to be trained again to fly new planes. As computer technology advanced, the simulators also advanced and its applications increased. Simulation training has been adopted in training professionals in other industries as listed below [3].

- Healthcare
- Military
- Law Enforcement
- Transportation
- Athletics

The list is not limited but still growing as simulation training becomes a tool to train employees.

Figure 2. The First Flight Simulator [4]

1.2 Simulation in Healthcare

The simulation training has been adopted in the healthcare industry. The training by simulators is now in action for fellow training, post-graduate training, board certification, and maintenance of certification. Simulation became a good tool for interventional cardiology. Accreditation Council for Graduate Medical Education (ACGME) mandates that cardiovascular fellowship training programs must have some component of simulation as part of fellow training. The Food and Drug Administration (FDA) adopted simulation training as part of its new procedure device training.

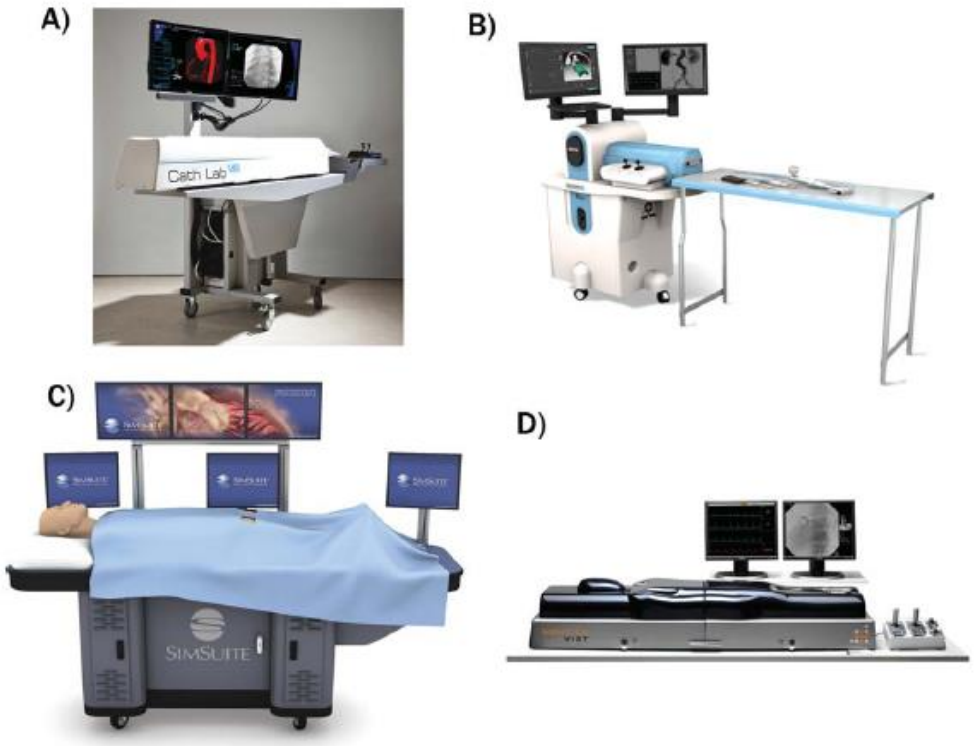


Figure 3. Coronary Angiography Simulators in the Market: (A) CathLabVR (B) Angiomentor (C) Simsuite (D) Procedicus VIST [5]

1.3 Simulation of Coronary Angiography

CA Simulators have become increasingly popular in medical training with cautions. There are researches on the use of CA simulators in medical training. Jensen et al. [6] concluded that the use of simulators did not improve learning of CA. Several years later, Jensen et al. [7] showed the better performance of trainees who

were trained with simulators than the trainees who were not. There were significant advancements of technology that can be adopted in CA simulators such as graphic processors, sensors. Those advancements improved computing speed and graphics, which resulted in more efficient simulators. Four commercial simulators were introduced and compared: CathLabVR, Angiomentor, Simsuite and Procedicus VIST [5].

The common components of the simulators are PC, monitor, control box, device and accessories. All of the above simulators employ mechanical haptic feedback. The simulators generate active or passive resistance, which gives tactile feedback, in response to user movement of tools or instruments. All of the simulators offer quantitative metrics about the use of contrast, fluoroscopy time and recording how well the trainees respond to adverse events. A full range of angiographic procedures are available, including the insertion and manipulation of devices (guidewires, catheters, balloons, stents, and endografts) into the vascular system, in all of the simulators. The simulators also provide simulated control of a C-Arm, patient table with various fluoroscopic projections and the manipulation of fluoroscopic images.

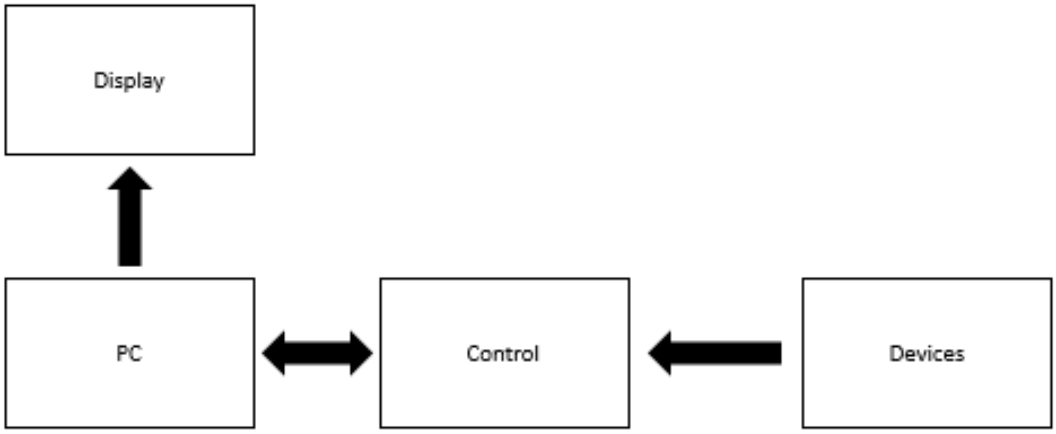


Figure 4. Simulation System Schematic Diagram

The purpose of CA simulators is to train trainees so that they become competent in the performance of CA. In order to meet the purpose, the simulators should produce ideally the same tactile feedback and be compatible with the same devices that are used in the real CA. Since the purpose of CA is to find out whether there is any blockage in coronary arteries. If any blockage is found, the blockage should be located through X-ray images from CA. CA simulators have to train trainees to anomaly of coronary arteries quickly and precisely so that patients may get right interventions in time. The ideal characteristics of simulators are biofidelic, metric-based, properly validated, realistic tactile feedback and proficiency-based progression [8].

1.4 Commercial Simulators - ANGIO MENTOR™ and Mentice VIST®

Two commercial simulators are examined in this section regarding how they realize the simulation of coronary angiography. All the information of the two simulation systems were from patents by Simbionix Ltd. [9] and Mentice AB [10].

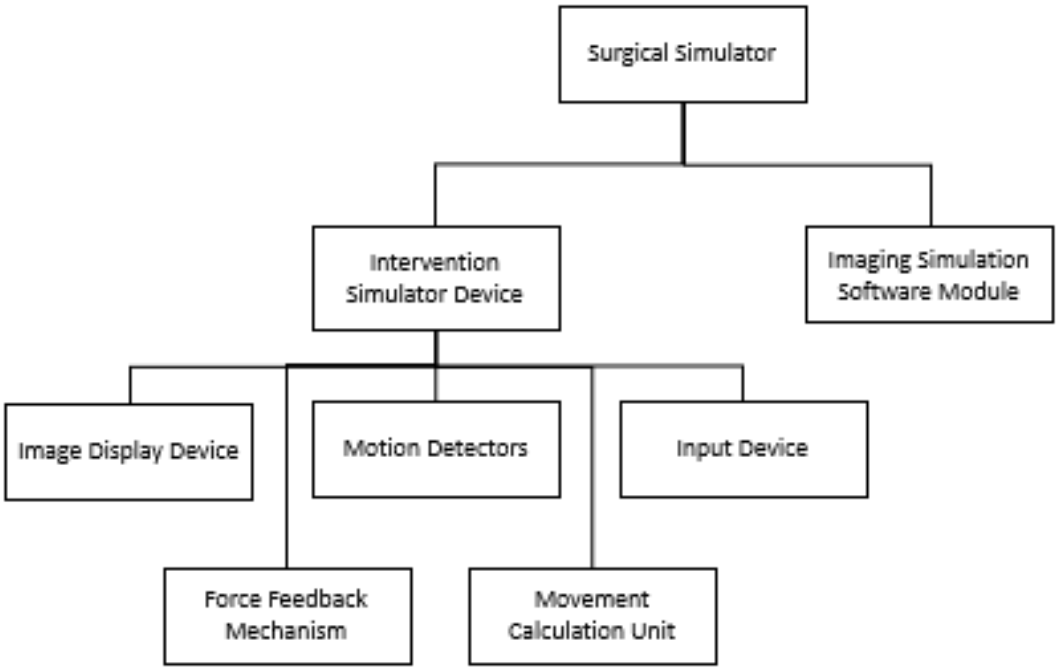


Figure 5. Block diagram of surgical simulation system

1.4.1 ANGIO MENTOR™

Angio Mentor simulators offer hands-on practice of endovascular procedures performed under fluoroscopy in the cath lab, interventional suite or an OR. Now 3D SYSTEMS manufactures the simulator systems. 3D SYSTEMS is formerly Simbionix, located in Iliois, US.

The surgical simulator consists of an intervention simulator device and an imaging simulation software module. The intervention simulator device provides user inputs to a computer and tactile feedbacks like a real coronary angiography procedure. The intervention simulator device is composed of an image display device, motion detectors, an input device, force feedback mechanism, and movement circulation unit. The motion detectors determine the location of the interventional instrument in a predetermined area near the detector. The movement calculation unit updates the position of a corresponding software simulation based on the navigation signals from motion detector. The force feedback mechanism provides simulated tactile feedback according to the calculated position of the software interventional instrument in the computer simulation.

The second subsystem an imaging simulation software module. It has the following functions.

- Receiving inputs from the motion detectors
- Analyzing the inputs using the movement calculation unit
- Translating the outcome to visual and tactile outputs and transferring them to the display device
- Force the feedback mechanism

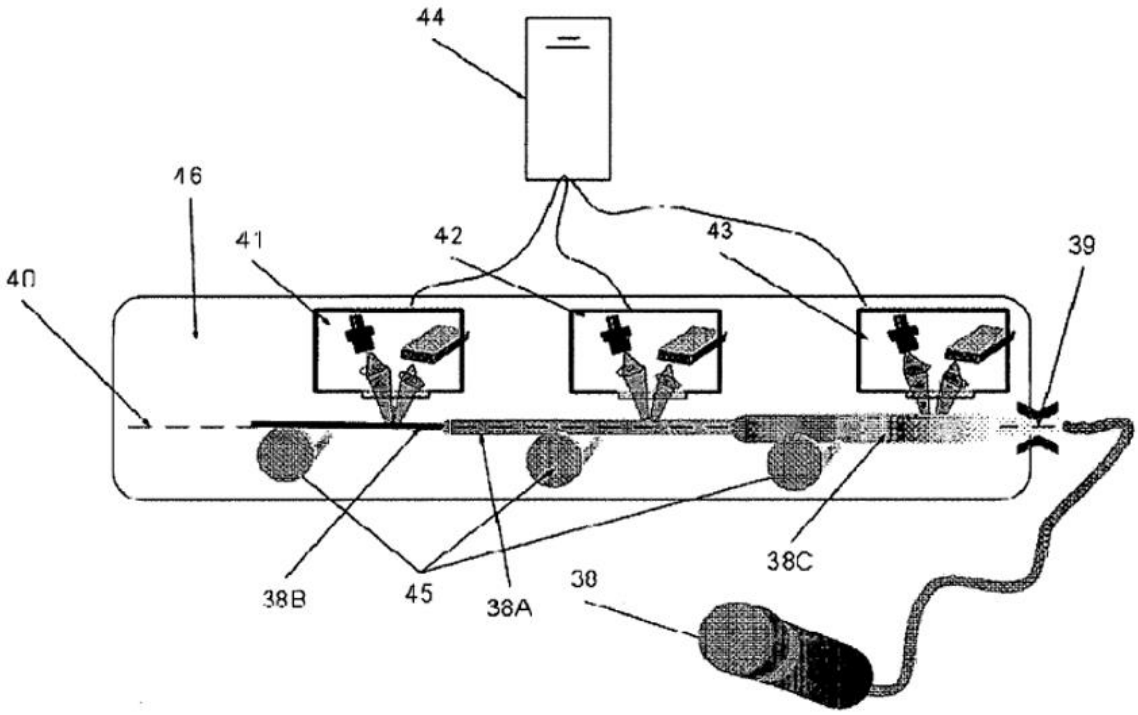


Figure 6. Illustration of simulation system by Simbionix Ltd. 1.Simulation System 2. Enclosure 3. Catheter or Guidewire, 3A. Cavity 4. CPU 5. Monitor 5A. Input Device [9]

The simulation system includes an enclosure(2), a simulated interventional instrument (3) such as catheter and guidewire, computer processor unit (4), monitor (5). An input device (5A) is connected to the processor for the configuration of the simulation. The interaction between an operator and the system occurs with the intervention simulator device. An operator manipulates inserts a catheter (3) into the cavity (3A) and manipulates it. The tactile and visual feedbacks are provided by the imaging simulation software module according to the position of the catheter within the enclosure. The visual feedback is provided through the monitor and the tactile feedback through the enclosure.

Figure 7 shows the intervention simulation device that includes 3 motion detectors(41, 42, 43) . Along the axis (40), the catheter(38A), the guidewire(38B) and the deflated balloon (38c) move after being inserted through the cavity (39). As the devices passes the motion detectors along the axis, the motion is detected by the reflection of the laser diode onto the detector photodiode detectors. Receiving

the navigation signals the processor (44) analyzes the catheter instantaneous position.

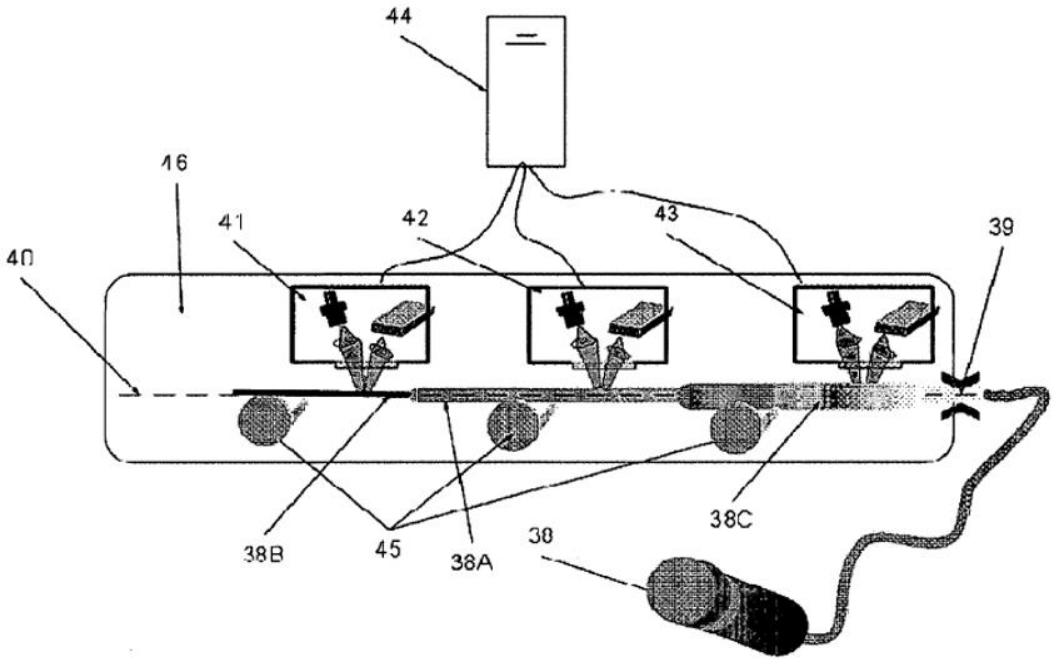


Figure 7. Intervention Simulation Device. 38A. Dummy Catheter 38B Guidewire 38C Deflated Balloon 39. Cavity 40. Axis 41,,42, 43. Motion Detectors 44. Processor 45. Resisting Force Generator 46. Enclosure [9]

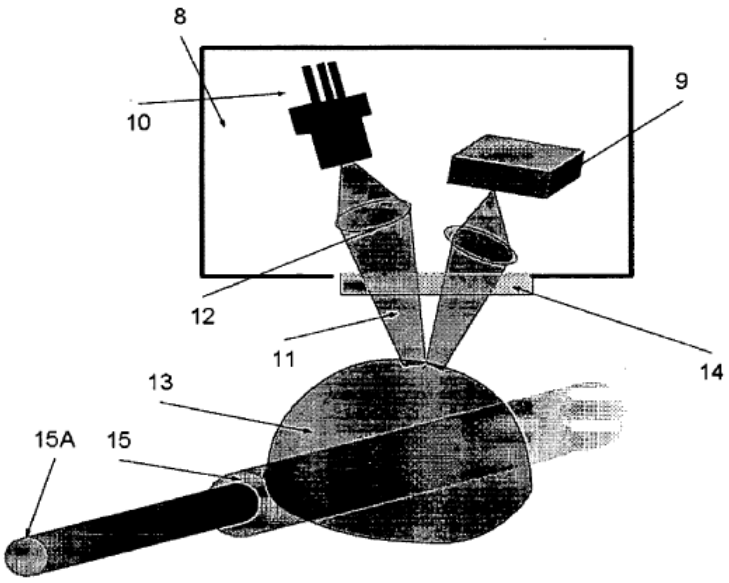


Figure 8. Motion Detector. 8. Motion Detector 9. Image Sensor 10. Laser Diode 11. Light 12. Lens 13. Predetermined bounded area 14. Window 15. Catheter 15A Guidewire

Figure 8 outlines a motion detector that is in the interventional instrument. The laser diode (10) emits light (11) through the lens(12) onto the predetermined bounded area (14). The light passes through the window (14) that is transparent and protects the motion detector from dust, dirt or other contamination. As the catheter moves, the light is reflected back onto the image sensor(9).

The resisting force generator simulates the tactile feedback of the procedure for the catheter and the guidewire in the simulated vascular tract. The resisting force generator gives different pressure on the catheter and the guidewire respectively according to the signals from the imaging software module. A wheel (45B) is positioned on the catheter advancement axis (45C). As the wheel (45B) rotates, the wheel edge (45E) puts different amount of pressure on the catheter or its guidewire.

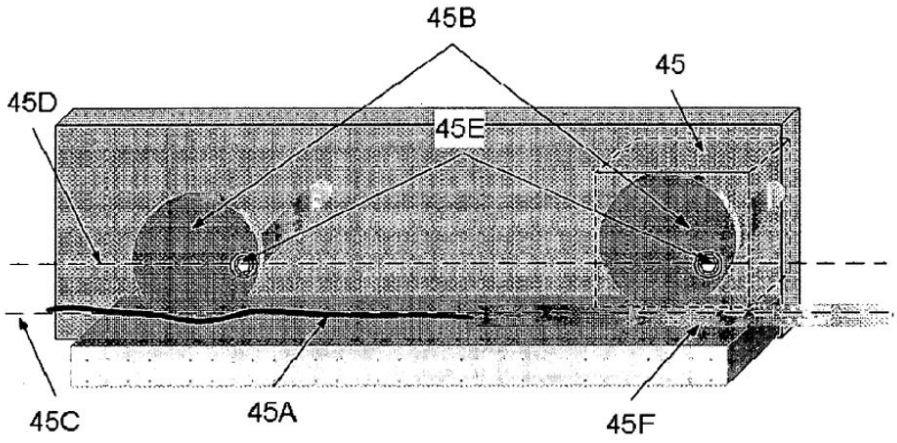


Figure 9. Resisting Force Generators. 45A. Guidewire 45B. Wheel 45C Axis 45D. Wheel Rotaion Axis 45E. Wheel Edge [9]

1.4.2 Mentice VIST®

The Mentice VIST® Simulator is a portable high-fidelity endovascular simulator that provides hands-on procedural training for medical professionals. This simulator is manufactured by Mentice AB, located in Gothenburg, Sweden.

The Mentice simulator system consists of computer unit(110) and interface device(120). The computer unit can be a PC which includes a display unit(111), input device(112). The interface device (120) is also called an interventional simulation device or an interventional simulation control system. It has severa instruments (121, 122, 123).

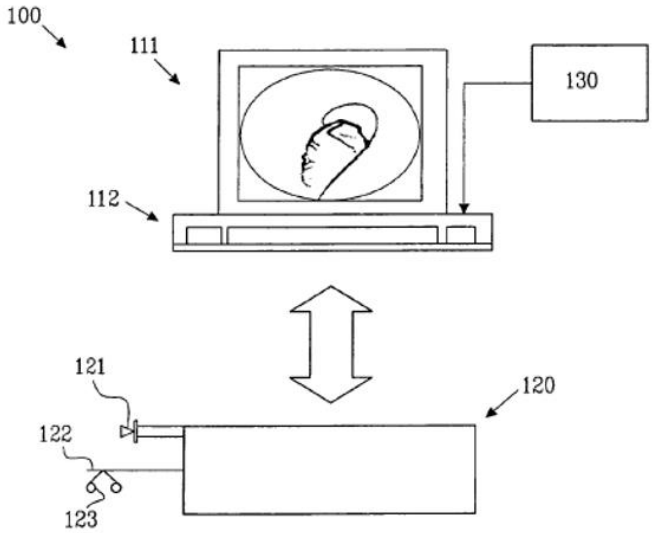


Figure 10. Components of Mentice Simulator 100. Apparatus 111. Display Unit 112. Input Device 120. Interface Device 121-123 Instruments 130. Scans [10]

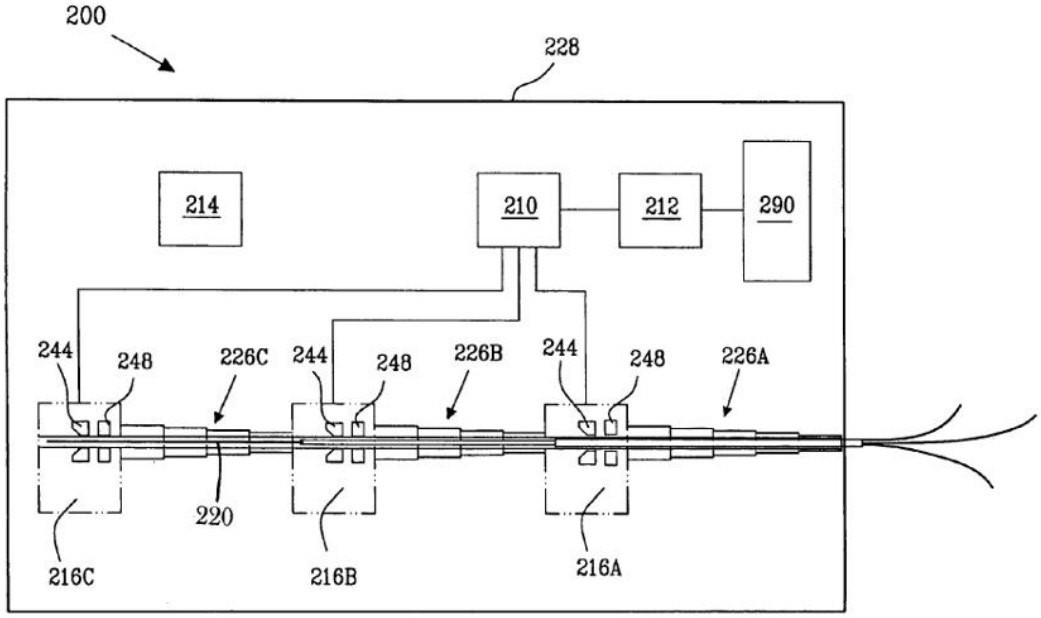


Figure 11. Mentice Interface Device (200. Interface Device 210. Processor Unit 212 Communication Unit 214 Power Supply 216A-216C. Carriage, 220. Track 226A-226C. Interconnecting Members 228. Cover 244. Locking Member 248. Detecting Member 290. Simulation Unit) [10]

The interface device (200) receives a number of instruments. It has a number of moveable carriages (216A-216C). The interconnecting member (226) interconnects the carriages serially. Each carriage receives instruments through an opening. As an instrument enter the carriage, the carriage locks the instrument and detects the type of the instrument(248). An optical sensor is used for the detection of the presence of an instrument in the carriage. The control unit (PC) measures the movement of each carriage and control the movement of the carriage that moves along the track (220) It also measures a longitudinal movement and a movement of rotation of the instrument and provides force-feedback in the logitudinal direction and in the direction of rotation according to the received force and torque.

1.5 Simulation Learning

As I mentioned in the introduction, the purpose of CA simulators is to train trainees so that they become competent in the performance of CA. It is very important for the trainees to transfer the effects of achieved skills in simulators to real world. The skills acquired through simulator trainings should reduce medical errors or complications. The evidence of transferability from simulator training to real life needs to be studied. Also, how to increase the transferability should be considered. However, unfortunately, the studies of transferability from simulator training to real coronary angiography procedure are scarce.

The learning process can be described in four stages of competence: Unconsciously incompetent, consciously incompetent, consciously competent, and unconsciously competent. At the unconsciously incompetent, the trainee does not know he is incompetent. He cannot recognize his deficit. At the consciously competent, the trainee is able to recognize his incompetence. At the consciously competent, the trainee understands and knows how to do the procedure though he depends on concentration following the procedural steps. Unconsciously competent trainees perform the procedure without reflecting over it.

A model of learning complex motor skills in three phases was introduced by Fitts and Posner in 1967 [11]. The three phases are cognitive stage, associative stage and autonomous stage. At the cognitive stage, the learner focuses on cognitive problems associated with the task. During this stage, the learner's performance varies and many errors are made by the learner. This stage is brief. In the associative stage, the learner associates different key skills to environmental circumstances and improves to detect errors related to their own performance and correct them. The final stage is the autonomous stage where the learner performs without a conscious thought and the skills become the learner's second nature.

In 1988, added the acquisition of cognitive skills to motor skills, Dreyfus and Dreyfus introduced a model [12]. The original model had 5 levels. A sixth level was added to the model. Novice is the first level where the learner sticks to the rules and plans with little ability to put the information into context. The next level is advanced beginner, where the learner has the ability to sort out relevant information and rules. However, he still has limitations to solve problems. Competent level is where the learner has the ability to see actions in terms of long-term goals. The learner at proficient level uses holistic views to analyze situations and has ability to identify the key elements in a situation. At the expert level, the learner can use analytic reasoning in new situations and does not depend on rules or guidelines. Mast level is where the learner has ability to see beyond the situation and uses unconscious practical wisdom to solve problems.

The efficiency of simulator training can be expressed in learning curves. A learning curve is the rate of learning over time. In the linear curve, a direct correlation exists between skill improvement and the number of time practiced. In the fast start curve, the beginner makes a fast progress in the beginning and slows down with on-going practice. In the slow start curve, the learner has hard time in acquiring the basic skill but later quickly progress. The stop-start-curve, which is probably the most common pattern of learning, shows alternate pattern of fast improvement and slow improvement. Between the two, plateaus exist.

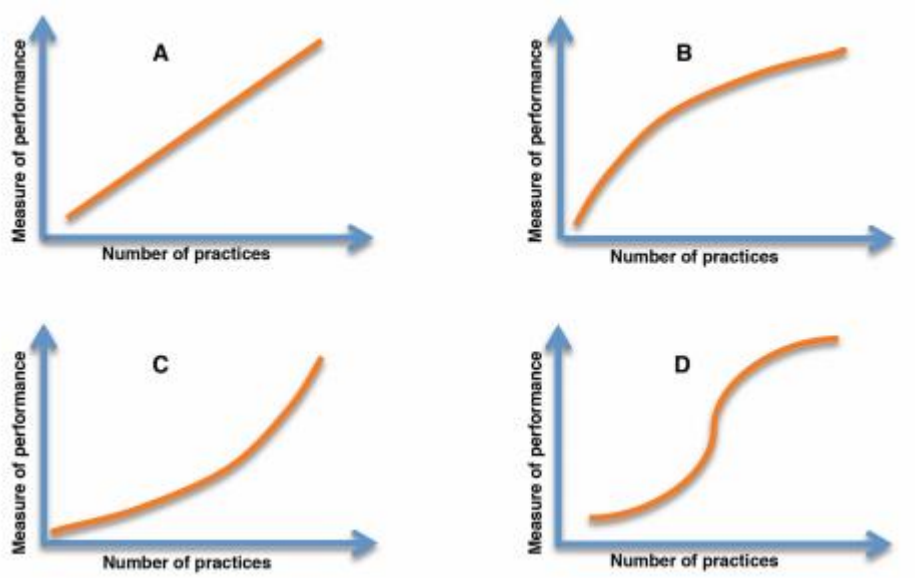


Figure 12. Characteristics of learning curves; A) the linear curve B) the fast start curve C) the slow start curve and D) the stop-start curve

The basic goals of simulator training were summarized into seven aspects [13].

1) Improve skills through interval practice
2) Improve consistency of performance
3) Decrease errors
4) Provide proximate and summative feedback
5) Allow for assessment of progress
6) Incorporate a standardized comprehensive curriculum
7) Optimize patient safety by accelerating the learning curve prior to patient exposure

In order to meet these goals, the simulator should include training and skills assessment. The simulator should have a way to measure the performance of the trainee. Metrics for the assessment of coronary angiography simulator training were studied which are total time (sec), Fluoroscopy (sec), Cath-LCA (sec), Cath-RCA (sec), Contrast (ml), and Cineloop (#) [13]. The study concludes that Fluoroscopy time was found the only sold marker for proficiency demonstrating a learning curve in the beginner group and the simulator trained residents performed better than the

conventional trained residents in quality and safety of the procedure and had shorter fluoroscopy time reflecting higher proficiency.

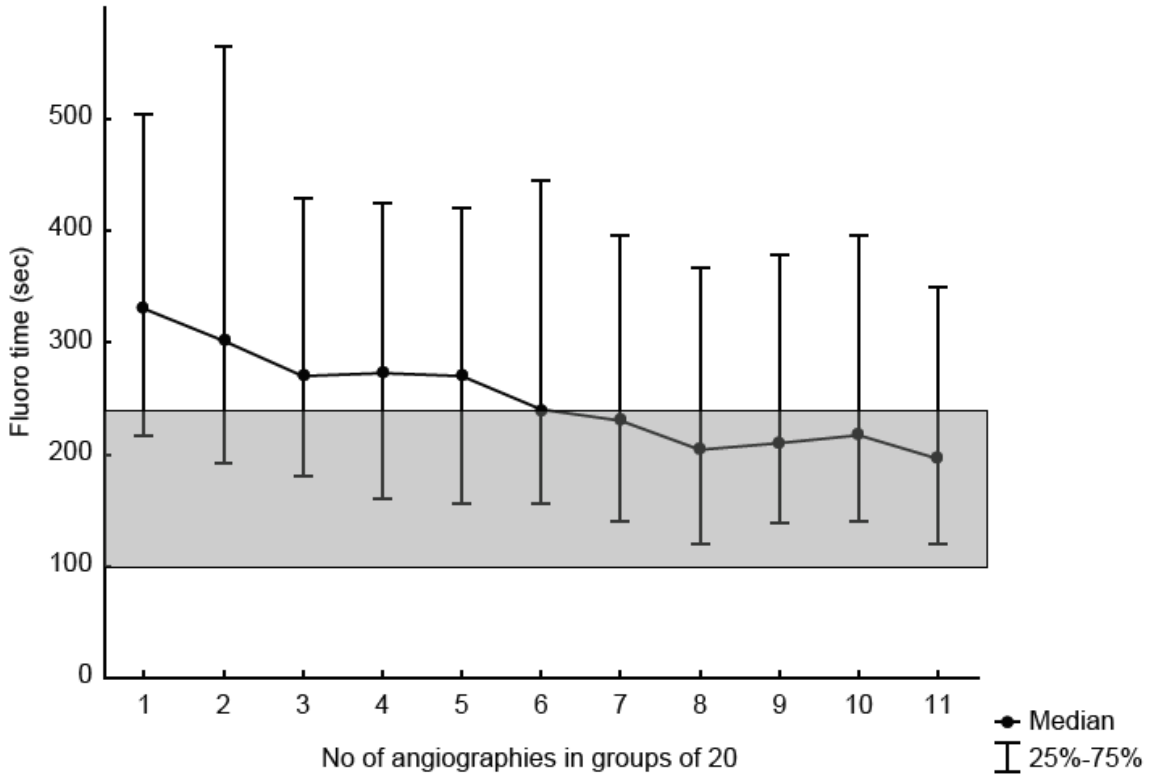


Figure 13. Median fluoroscopy time in beginners [13]

The Figure 13 shows the simulator training improves the performance of coronary angiography procedure regarding fluoroscopy time. This graph shows clearly how well the simulator trains trainees as the fluoroscopy time is the major metric for the assessment of proficiency.

According to the study, the simulator must have metrics to assess performance. It should train based on proficiency of trainees. The train level must be based on the trainee's level of performance. If the trainee does not reach a certain level of performance, he should not be trained further until he masters the current level.

The simulator of coronary angiography is not just about how well it simulates the real procedure including biofidelic images and realistic tactile feedback. The simulator training and the assessment of performance is also important to improve

the transferability of the acquired skills through the simulator to the real life coronary angiography procedure. Therefore, the simulator should have record the results of metrics from practices so that based on the proficiency, the trainee can go up to the next level of the practice until the trainee's proficiency level reaches an autonomous stage in which he can complete the coronary angiography procedure with unconscious competence.

Lastly, the simulator should provide incorporate realistic unexpected complications or difficulties in the training cycles so that the trainee can have the ability to cope with it safely and properly. The simulator has to generate sudden unexpected situations and circumstances to test the trainees whether they solve safely and properly.

Chapter 2. Coronary Angiography

2.1 Coronary Arteries

Heart supplies oxygenated blood to organs in human body through arteries. All the oxygenated blood exits the left ventricular chamber through ascending aorta. Heart supplies oxygenated blood to itself through coronary arteries.

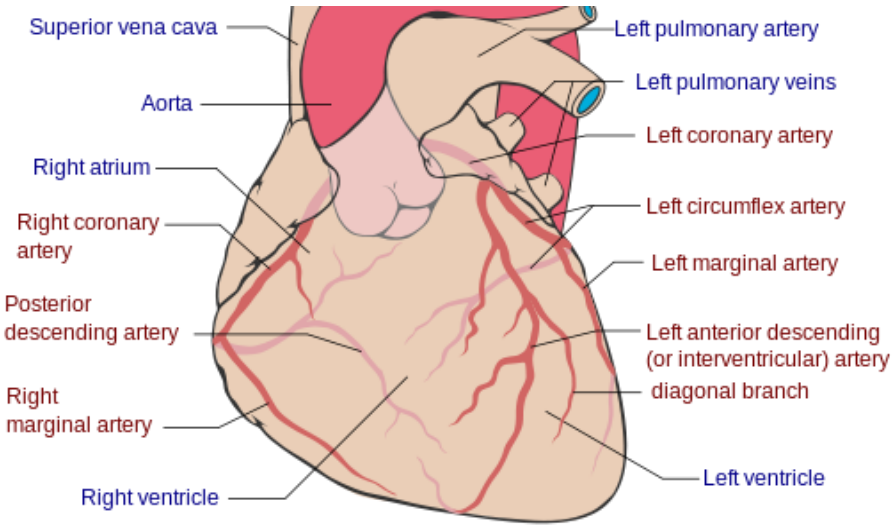


Figure 14. Coronary Arteries [14]

The left main stem (LMS) arises from the left aortic sinus of the ascending aorta. The LCA bifurcates into the left anterior descending (LAD) and left circumflex (Cx) coronary arteries. The LAD descends in the interventricular grove of the heart with diagonal branches and septal branches. The left Cx passes posteriorly down the left atrioventricular (AV) groove with obtuse marginal branches. The LMS supplies the left atrium, most of the left ventricle, part of the right ventricle.

The right coronary artery (RCA) arises from the right aortic sinus and runs down the right AV groove towards the inferior aspect of the heart with acute marginal branches, a conus branch, an AV node branch and a sinus node branch. The RCA gives off the posterior descending artery (PDA). Typically, the RCA supplies the right atrium, most of the right ventricle, part of the left ventricle.

Coronary dominance is determined by the vessel that gives off the PDA. The RCA supplies the PDA in 85% individuals and the left Cx does the PDA in 8% individuals. In 7% individuals, the RCA and the left Cx supplies the PDA, which is called co-dominance or a balanced circulation.

2.2 Contrast Agent Injection

To visualize the blood flow in coronary arteries, contrast agent is injected, which is a viscous, iodinated solution and has relatively low toxicity. Angiographic contrast agents are excreted by the kidney. There are two methods of injection of contrast agent: Hand injection and power injection. Typically, in the hand injection flow rates are usually 2 to 4 ml/sec with volumes of 2 to 6 ml in the right coronary artery (RCA) and 7 to 10 ml in the left coronary artery (LCA). Thanks to the advancement of technology, power injectors are more available providing the equal quality in safety to hand injection. Contrast agent can be delivered effectively and efficiently with power injector that can be programmed to deliver specific volume of contrast agent at specific flow rates.

Figure 15. ASCIST|CVi® Contrast Delivery System [15]

The four benefits of power injection are described: tight “Bolus” of Contrast Media, Precise Timing of Contrast-Media Delivery, Consistent, Reproducible, Patient-Specific Results, Time and Cost Savings [15]. The contrast agent can be concentrated in the area of interest during the scan. So, the tight “bolus” enhances the visualization of the final image without much waste. With power injector, timing of delivery can be controlled which is very important when scanning the heart. Power injectors make possible a consistent flow rate and volume from scan to scan which enables doctors to customize injection protocols for specific studies and specific patients. Automated features of a power injectors save time and money through efficient healthcare procedure.

Figure 16. X-ray image intensifier consists of X-ray generator, intensifier and C-arm

2.3 Angiographic Imaging

X-ray image intensifier is used to visualize the blood flow with contrast medium. It consists three parts: X-ray generator, intensifier and C-arm. The low intensity x-rays are converted to a bright visible light output by intensifier. The movable C-arm enables doctors to focus on various parts of coronary arteries.

I will describe the nomenclature for angiographic views and when the specific angiographic view is used for a specific coronary artery. In the anteriorposterior (AP) projection, the image intensifier is directly over the patient and the beam passes through the patient from back to front as the patient lies flat on the x-ray table. In this view, the left anterior descending (LAD) and left circumflex

coronary arteries branches overlap. In the cranial view, the image intensifier is tilted toward the head of the patient.

In the caudal view, the image intensifier is tilted toward the feet of the patient. In the right anterior oblique (RAO) projection, the image intensifier is on the right side of the patient. In the RAO caudal view is the best view for the visualization of the circumflex artery. It also shows the left main coronary artery bifurcation with the origin and course of the circumflex/obtuse marginal, intermediate branch and proximal LAD segment as well. The RAO view is chosen for the right coronary artery (RCA), which shows the mid RCA and the length of the posterior descending artery and posterolateral branches.

In the left anterior oblique (LAO) view, the image intensifier is to the left side of the patient. The LAO cranial view shows the left main coronary artery, LAD and diagonal branches. For the RCA, the LAO cranial view shows the origin of the artery and its entire length and the posterior descending artery bifurcation.

A left lateral view shows the mid and distal LAD well. The LAD and circumflex are well separated. For the RCA, in the lateral view, the origin and the mid RCA are well shown.

Left Coronary Artery	For Concentration on Vessel Segment
Straight AP or 5–10 degrees RAO with caudal	Left main
30-45 degrees LAO and 20-30 degrees cranial	LAD-circumflex bifurcation
30-40 degrees RAO and 20-30 degrees caudal	Circumflex and marginal branches
5-30 degrees RAO and 20-45 degrees cranial	LAD+diagonals
50-60 degrees LAO and 10-20 degrees caudal	LAD-circumflex bifurcation, circumflex, marginals

Table 1. Routine Coronary Angiographic Views – Left Coronary Artery

In a web site [16], tables show the best setup for visualizing the specific coronary artery. For left main artery, origin and bifurcation and course through body, AP view is suggested. For proximal circumflex segment, RAO cranial view is

suggested for its origin and bifurcation. During coronary angiography, there are common coronary angiographic views to see coronary arteries. In order to focus on the LAD-circumflex bifurcation, the angiographic view is suggested, 30 degrees to 45 degrees LAO and 20 degrees to 30 degrees cranial.

Right Coronary Artery	For Concentration on Vessel Segment
30-45 degrees LAO and 15-20 degrees cranial	Proximal, mid, PDA
30-40 degrees RAO	Proximal, mid, PDA

Table 2. Routine Coronary Angiographic Views – Right Coronary Artery

X-ray image mode sizes are normally less than 7 inch in diameter, which make impossible to scan the entire coronary artery course. It is necessary to pan over the heart to include the distal arterial or collateralized segments. Panning is necessary to see regions which are not visible from the initial setup positioning.

Chapter 3. Simulation method of Guidewire and Catheter – Mass-Spring Model

3.1 Introduction

Mass-spring model is one of the two most common methods that are used in surgical simulation [17]. The guidewire has been researched for the simulation with mass-spring model [18] [19]. The other common methods are finite-element model [20], elastic rod model [21] [22] and position-based dynamics. Mass-spring model was chosen because of its simplicity and availability of fast, efficient methods for solving the mass-spring system.

As guidewire moves in blood vessels, it interacts with blood vessels. There are many methods and algorithms developed for collision detection. Collision detection is the process that detects whether the guidewire collides with blood vessels or not. The collision detection methods will be described later.

I will discuss theoretical background first about particle system, mass-spring system for deformable models, and collision detection. After the theoretical background, I will describe how I applied the concepts of particle system and mass-spring system for modeling guidewires and the algorithms for collision detection.

3.2 Background

3.2.1 Particle System

For guidewire simulation, particle system was employed. Particle system is a collection of particles which obey physical laws. The virtual guidewire consists of particles connected by springs and dampers. Each particle has its own mass, position and velocity. The particles respond to forces. Particles are the easiest objects for simulation. The collection of particles is called particle system. Particle system dynamics are well explained by Andrew Witkin [23].

Each particle has its mass, position, velocity and force at a certain time, t. Velocity is time derivative of position.

Position: $\vec{x}$ = (x, y, z)

Velocity : $\vec{v} = \dot{x} = \frac{d\vec{x}}{dt} = (\frac{dx}{dt}, \frac{dy}{dt}, \frac{dz}{dt})$ $Eq.3.1$

Acceleration: $\vec{a} = \ddot{x} = \frac{d\vec{v}}{dt}$

From the above equations, 1st-order differential equation can be derived.

$$\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} \vec{v} \\ \frac{\vec{f}}{m} \end{bmatrix} \qquad Eq.3.2$$

3.2.2 Ordinary Differential Equation Solver

There are several methods to solve this ordinary differential equations (ODEs).

3.2.2.1 Euler's method

Euler's method is a first-order numerical procedure for solving ordinary differential equations. It is the simplest numerical method. In the method, an initial value is given. It computes an estimate of x after step size h.

$$x(t_0 + h) = x_0 + h\dot{x}(t_0) \qquad Eq.3.3$$

If we look at Taylor series, we can tell how Euler's method was derived.

$$x(t_0 + h) = x(t_0) + h\dot{x}(t_0) + \frac{h^2}{2!}\ddot{x}(t_0) + \cdots + \frac{h^n}{n!}\frac{\partial^n x}{\partial t^n} + \cdots \qquad Eq.3.4$$

Euler's method takes the first two terms on the right hand side. Euler's method is correct only if all the derivatives beyond the first are zeroes. The difference between Euler's method and the full Taylor series is dominated by $\frac{h^2}{2!}\ddot{x}(t_0)$. We can reduce the error by reducing the step size h. For example, by making h half, the difference becomes fourth.

3.2.2.2 Mid-point Method

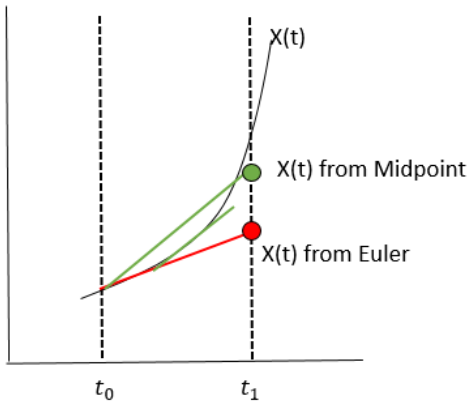


Figure 17 Mid-point method

Euler's method is simple but inaccurate and instable. To improve it, this method retain one additional term in Taylor Series.

$$x(t_0 + h) = x(t_0) + h\dot{x}(t_0) + \frac{h^2}{2!}\ddot{x}(t_0) + O(h^3) \qquad Eq.3.5$$

This equation can be simplified as written below.

$$x(t_0 + h) = x(t_0) + hf\left(x_0 + \frac{h}{2}f(x_0)\right) \qquad Eq.3.6$$

3.2.2.3 Runge-Kutta Method

Runge_Kutta method is the most used ODE solving method.

$$k_1 = hf(x_0, t_0)$$

$$k_2 = hf(x_0 + \frac{k_1}{2}, t_0 + \frac{h}{2})$$

$$k_3 = hf\left(x_0 + \frac{k_2}{2}, t_0 + \frac{h}{2}\right) \qquad Eq.3.7$$

$$k_4 = hf(x_0 + k_3, t_0 + h)$$

$$x(t_0 + h) = x_0 + \frac{1}{6}k_1 + \frac{1}{3}k_2 + \frac{1}{3}k_3 + \frac{1}{6}k_4$$

3.3 Mass-Spring Model

A simulated guidewire consists of particles, springs and dampers. Particles are connected with each other through springs and dampers. The elements obey Newton's Laws of Motion and Hooke's Law. Each particle can have forces applied upon it. The forces can be from gravity, force field, drag or combinations of these. These forces on a particle were grouped in to three broad categories: unary forces, n-ary forces and forces of spatial interaction [23]. Unary forces are independent forces upon each particle. They are constant or vary depending on particle position, velocity and time. The examples of unary forces are gravity and drag. N-ary forces are forces that are applied to a fixed set of particles. Springs are one of n-ary forces. Forces of spatial interaction such as attraction and repulsion are forces that act on any or all pairs of particles, according to their positions. I will discuss the forces below in detail.

3.3.1 Unary Forces

3.3.1.1 Gravity

The gravitational force in each particle is proportional to its mass (m) and gravity (g) which is a constant vector.

$$f = mg \qquad Eq.3.8$$

3.3.1.2 Viscous Drag

The viscous drag force resists motion and makes a particle to stop its motion gradually.

$$\overrightarrow{f_{drag}} = -k_d\vec{v} \qquad Eq.3.9$$

k_d is the coefficient of drag.

3.3.2 n-ary Forces

In Hooke's law, the force to extend or compress a spring by a distance X is proportional to the distance.

$$F = k_s X \qquad Eq.3.10$$

, which k_s is a constant. The spring constant k_s represents its stiffness.

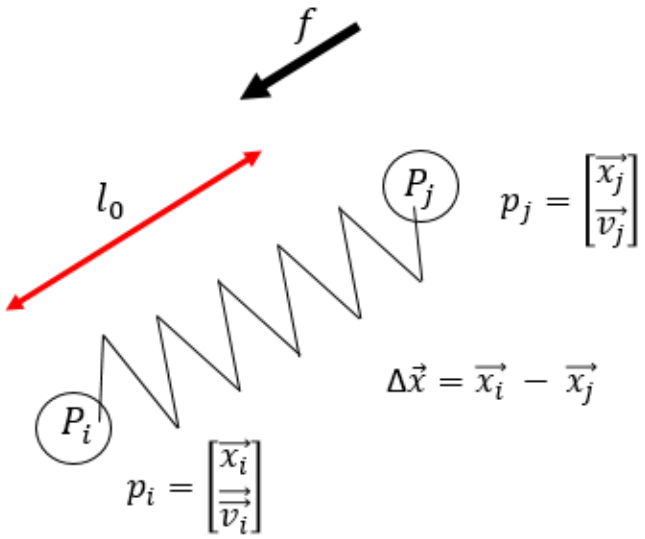


Figure 18 Particles with a spring

In Figure 18, the spring has l_0 rest length. F is force in the direction of the spring. As the spring is extended, it tends to return to its rest length. So, the force works towardP_i.

The force can be written as below.

$$F(P_i, P_j) = k_s\left(l_0 - \left\|\overrightarrow{P_iP_j}\right\|\right)\frac{\overrightarrow{P_iP_j}}{\left\|\overrightarrow{P_iP_j}\right\|} \qquad Eq.3.11$$

Combined with a viscous drag force and the internal spring force, we can obtain the overall force on a particle.

The force due to a spring,

$$f = -k_s(\Delta|\vec{x}| - l_0) \qquad Eq.3.12$$

Augmented with a viscous drag force,

$$f = -[k_s(\Delta|\vec{x}| - l_0) + k_d|\vec{v}|] \qquad Eq.3.13$$

The resulting combined n-ary force,

$$f_i = -\left[k_s(\Delta|\vec{x}| - l_0) + k_d \frac{\Delta\vec{v}\cdot\Delta\vec{x}}{|\vec{x}|}\right]\frac{\vec{x}}{|\vec{x}|} \qquad Eq.3.14$$

$$\overrightarrow{f_J} = -\overrightarrow{f_i}$$

According to Newton's law, the force works on P_i with the same magnitude but the opposite direction.

3.4 Collision Detection

Computer simulation involves movements of objects in 3D space. As the objects moves in the space, they may hit one another. The interaction between objects must imitate the interaction in the real world. In order to imitate the interactions between objects, different methods have been applied according to the objects' properties. The objects may be soft changing its shape or hard keeping its shape. Collision occurs as two objects hit each other.

There are methods developed for collision detection. I will introduce several methods for collision detection.

3.4.1 Bounding Volumes (BV)

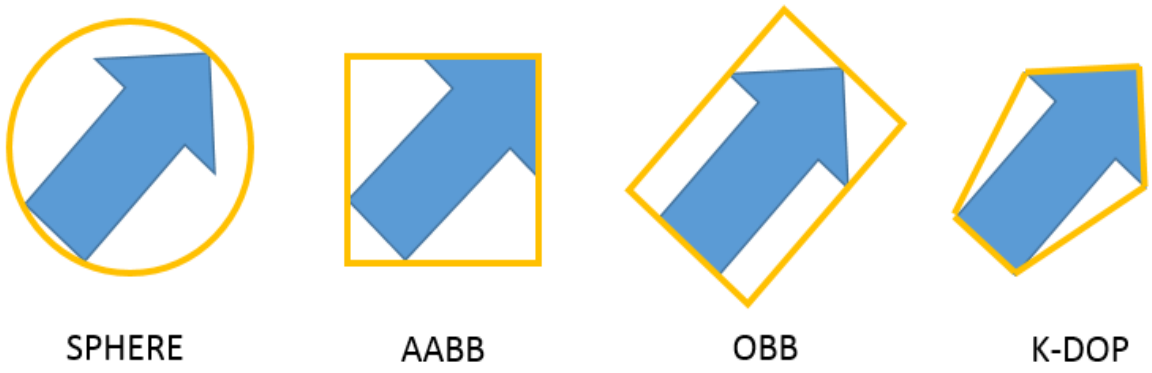


Figure 19 Common bounding volumes: Sphere, Axis-Aligned Bounding Box (AABB), Oriented Bounding Box (OBB), k-Discrete Oriented Polytope (DOP)

In this method, a complex shaped object is changed to a simpler geometric shape that surrounds the original objects. The simpler geometric shape can be a sphere, a cube or a cylinder. This simpler geometric shape is called a bounding volume. The bounding volume should be chosen carefully for efficiency. Desirable BV characteristics were listed [24].

BV must fit the object tightly. It has to be inexpensive to compute and easy to rotate and transform with minimum variance. It should take less resource of computer for intersection tests.

In Figure 19, k-DOP fits most tightly than other bounding volumes and sphere fits least. However, sphere is the simplest bounding volume that takes least memory and takes the least computation time for intersection test. Each bounding volume has its own pros and cons. For accurate medical simulations, k-DOP is preferred. But, for game that requires fast intersection tests, sphere has advantage over other bounding volumes. A type of bounding volume has to be selected accordingly.

3.4.1.1 Sphere

Sphere is the simplest bounding volume. Even though it gives low accuracy, it takes less computation time for intersection tests. If the distance between the centers of spheres is less than the sum of radii of spheres, two objects are considered to collide each other.

In Figure 5, d is the distance between the centers of spheres. R1 and R2 is the radii of the spheres. If the distance is equal to the sum of the radii of the spheres, the spheres touch each other (d = R1 + R2). If d > R1 + R2, the spheres do not collide.

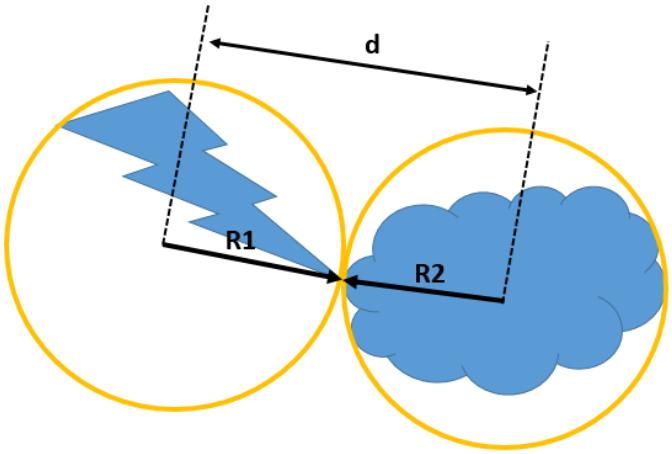


Figure 20 Sphere bounding volume collision detection

3.4.1.2 Axis-Aligned Bounding Box (AABB)

AABB is the most common bounding volume type. In 3D, the box is a rectangular six-sided box and in 2D, four sided. Its face normal are always parallel with axes of the coordinate system. For collision detection, AABB is projected onto axes. If the projected lines from AABBs are overlapped on all axes, the AABBs are considered to be interpenetrated.

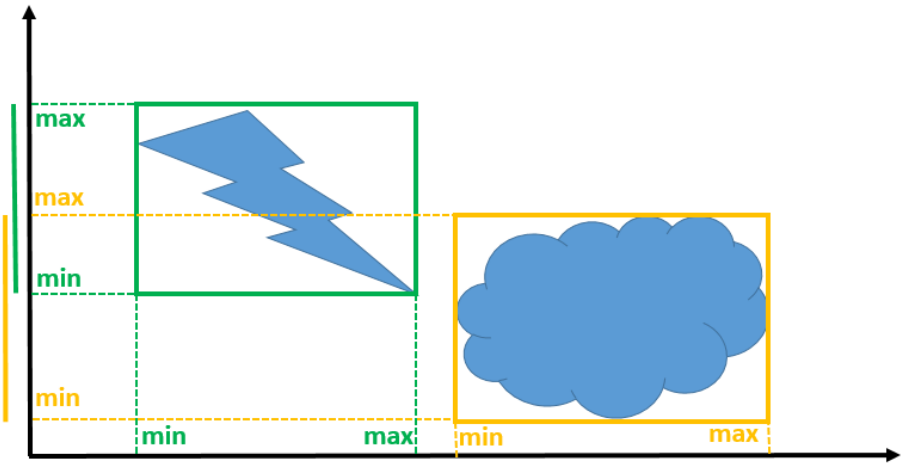


Figure 21 AABB Collision Detection

As seen in Figure 21, the maximum and minimum value on each axes are compared. If the maximum of one AABB is greater than a minimum of the other AABB, it is called overlapped. In Figure 6, the projected lines are overlapped on only one axis, so the AABBs do not collide.

3.4.1.3 Oriented Bounding Box (OBB)

An oriented bounding box (OBB) is also a rectangular box like an AABB, but its surface normals are not always parallel with the axes of the given coordinate system.

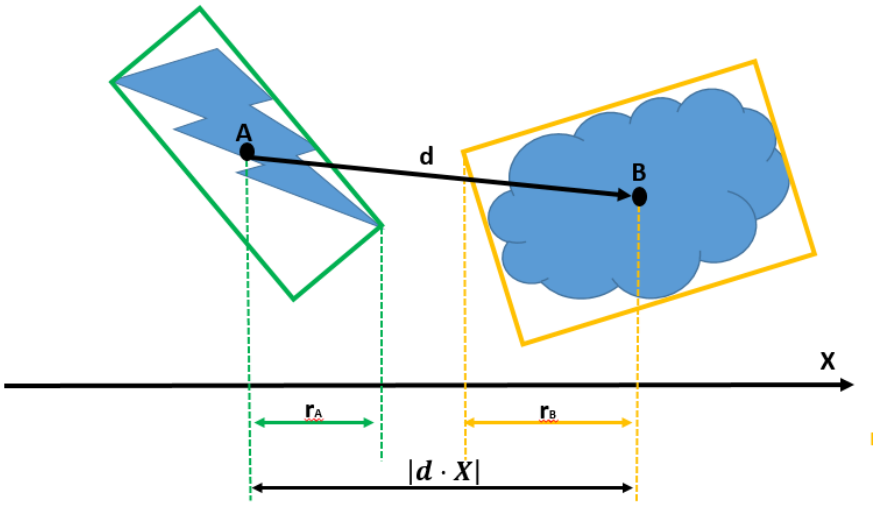


Figure 22 OBB Collision Detection

In Figure 22, the distance d between the centers of the OBBs is greater than its projection to X axis ($|\boldsymbol{d} \cdot \boldsymbol{X}|$). The OBBs do not collide. If d < $|\boldsymbol{d} \cdot \boldsymbol{X}|$, then, the OBBs are considered to collide each other.

3.4.1.2 Discrete-Oriented Polytope (k-DOP)

Discrete-Oriented polytopes (k-DOPs) form a bounding volume with fixed normal and minimum and maximum values. A closed 3D volume is formed with at least three slabs. AABB and OBB are formed with three slabs. Normal components are typically limited to the {-1,0,1} set and not normalized. The intersection test is similar to that of two AABB overlapping test. dnear and dfar of k-DOP are compared with those of other k-DOP to see whether they are overlapping each other.

k-DOP bounding volume is formed with normal, a plane with a distance d_{near} and a plane with a distance d_{far}. The region between two planes is called a slab.

3.4.2 Particle-Plane Collision Detection

The equation of a plane in 3D space can be expressed with normal vector and a point on the plane. The equation of a plane with a normal vector $\vec{n}$ through the point $P_1 = (x_1, y_1, z_1)$ is

$$\vec{n} \cdot (P - P_1) = 0 \qquad Eq.3.15$$

, where $P = (x, y, z)$. Plugging in gives the general equation of a plane,

$$ax + by + cz + d = 0 \qquad Eq.3.16$$

Normal vector $\vec{n}$ = (a, b, c)

, where $d = -ax_1 - by_1 - cz_1$.

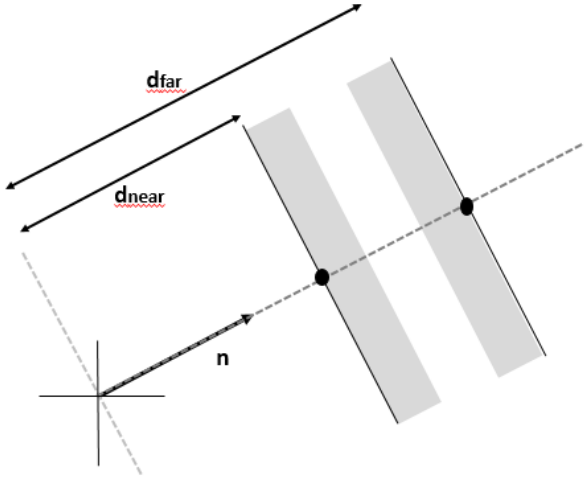


Figure 23 k-DOP bounding volume

This equation can be rewritten as

$$F(p) = ax + by + cz + d \qquad Eq.3.17$$

As seen in Figure 24 (b), a particle on the plane results in F(p) = 0. Particles that are located in the different sides show different signs, either positive or negative. Later, this properties are used for the collision detection based on the fact that the sign changes to the opposite sign as the particle crossed the plane to the other side.

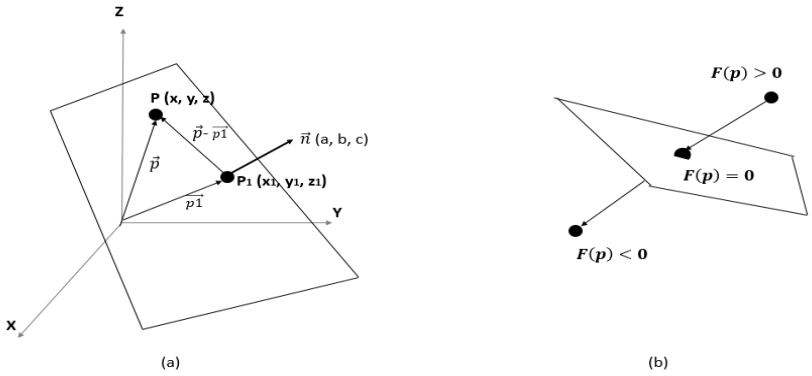


Figure 24 Particle Collision (a) plane with normal vector (b) particle positions

The shortest distance from a point to a plane can be calculated by projecting the vector $(\overrightarrow{P_2} - \overrightarrow{P_1})$ to the normal vector $\vec{n}$ of the plane. Figure 25 shows how to calculate the distance from a point to a plane. Basically, the distance is the dot product of the plane normal and particle vector $(\overrightarrow{P_2} - \overrightarrow{P_1})$.

$$D = \left|\overrightarrow{P_2} - \overrightarrow{P_1}\right| \cos\theta = \frac{\vec{n}\cdot(\overrightarrow{P_2}-\overrightarrow{P_1})}{|\vec{n}|} = \frac{\vec{n}\cdot(\overrightarrow{P_2}-\overrightarrow{P_1})}{\sqrt{a^2+b^2+c^2}} \qquad Eq.3.18$$

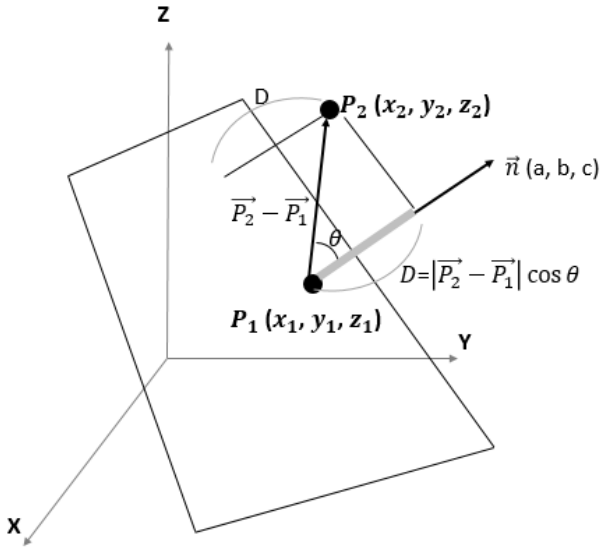


Figure 25 Distance from a point

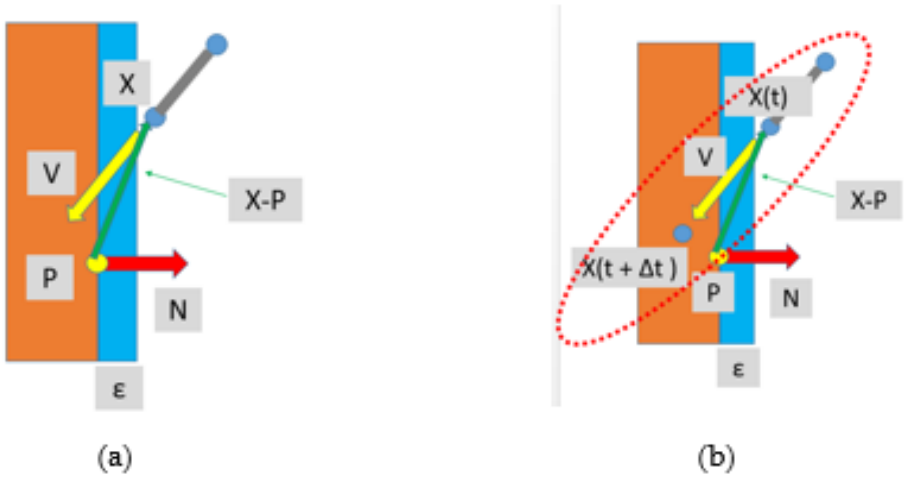


Figure 26 Collision Detection

In Figure 26, X is a vector that contains a particle position and V is a velocity vector. P is a point on the plane and N is a normal vector of the plane. ε is the minimum distance. If the distance from a plane is less than ε, then the particle considered to touch the plane. The product of the normal vector (N) and the velocity vector (V) shows the direction of the particle. If the product is less than zero, the particle moves toward to the plane.

If $(X - P) \cdot N < \varepsilon$, the particle X touches the plane.

If $N \cdot V < 0$, the particle heads to the plane.

In Figure 26 (b), a particle crossed the plane after a step size Δt. Whether a particle crossed a plane or not may be checked through the product of vectors before and after collision.

If $[(X(t) - P) \cdot N] * [X(t + \Delta t) \cdot N] < 0$, the particle crossed the plane. It means that collision occurs.

$$[(X(t) - P) \cdot N] * [X(t + \Delta t) \cdot N] < 0 \qquad Eq. 3.19$$

A plane is a flat surface extending forever. It does not have limit. But, in the simulation, a plane is limited, usually triangle. After collision detection, it should be checked whether the collision point is on the triangle or not. For this check, the collision point must be determined.

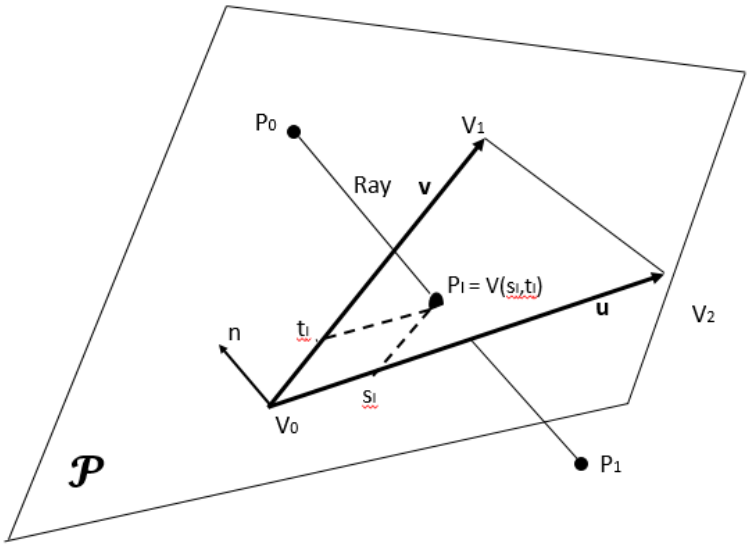


Figure 27 Collision point on a triangle

In Figure 27, a ray R from P0 to P1 goes through a plane P with normal n at a point PI. The ray can be written as

$$P(r) = P_0 + r(P_1 - P_0) \qquad Eq.3.20$$

If the intersection of the ray and the plane P occurs at a point P(RI), RI is obtained as below.

$$r_I = \frac{n \cdot (V_0 - P_0)}{n \cdot (P_1 - P_0)} \qquad Eq.3.21$$

The ray **R** goes through the plane P only with $r_I > 0$.

In Figure 12, a triangle **T** with vertices V_0, V_1, V_2 is on the plane **P**. The normal vector **n** can be obtained through a cross product.

$$n = (V_1 - V_0) \times (V_2 - V_0) = \mathbf{u} \times \boldsymbol{v} \quad Eq.3.22$$

The position of a point P can be determined with the coordinates $(\boldsymbol{u},\boldsymbol{v})$ on the triangle **T**[2].

$$P(\boldsymbol{u},\boldsymbol{v}) = (1-\boldsymbol{u}-\boldsymbol{v})V_0 + \boldsymbol{u}V_1 + \boldsymbol{v}V_2 \qquad Eq.3.23$$

$$P(\boldsymbol{u},\boldsymbol{v}) = V_0 + \boldsymbol{u}(V_1 - V_0) + \boldsymbol{v}(V_2 - V_0) \qquad Eq.3.24$$

$$P(\boldsymbol{u},\boldsymbol{v}) = V_0 + s\boldsymbol{u} + t\boldsymbol{v} = V(s,t) \qquad Eq.3.25$$

, where $s = (V_1 - V_0)\ and\ t = (V_2 - V_0)$.

If $P = V(s,t)$ is on the triangle **T** with $s \geq 0, t \geq 0\ and\ s + t \leq 1$. When $= 0, t = 0\ or\ s + t = 1$, P is on the edge of the triangle **T**. s and t can be obtained with Barycentric Coordinate computation.

From the equations above, we can rewrite an equation.

$$P(\boldsymbol{u},\boldsymbol{v}) - V_0 = s\boldsymbol{u} + t\boldsymbol{v} \qquad Eq.3.26$$

If we put $\mathbf{w} = \boldsymbol{P}(\boldsymbol{u},\boldsymbol{v}) - \boldsymbol{V_0}$,

$$\boldsymbol{w} = s\boldsymbol{u} + t\boldsymbol{v} \qquad Eq.3.27$$

$$\mathrm{s} = \frac{\boldsymbol{w}\cdot(\boldsymbol{n}\times\boldsymbol{v})}{\boldsymbol{u}\cdot(\boldsymbol{n}\times\boldsymbol{v})}\ and\ \mathrm{t} = \frac{\boldsymbol{w}\cdot(\boldsymbol{n}\times\boldsymbol{u})}{\boldsymbol{v}\cdot(\boldsymbol{n}\times\boldsymbol{u})}$$

This equation can be rewritten without cross product.

$$\mathrm{s} = \frac{(\boldsymbol{u}\cdot\boldsymbol{v})(\boldsymbol{w}\cdot\boldsymbol{v}) - (\boldsymbol{v}\cdot\boldsymbol{v})(\boldsymbol{w}\cdot\boldsymbol{u})}{(\boldsymbol{u}\cdot\boldsymbol{v})^2 - (\boldsymbol{u}\cdot\boldsymbol{u})(\boldsymbol{v}\cdot\boldsymbol{v})}\ and\ \mathrm{t} = \frac{(\boldsymbol{u}\cdot\boldsymbol{v})(\boldsymbol{w}\cdot\boldsymbol{u}) - (\boldsymbol{u}\cdot\boldsymbol{u})(\boldsymbol{w}\cdot\boldsymbol{v})}{(\boldsymbol{u}\cdot\boldsymbol{v})^2 - (\boldsymbol{u}\cdot\boldsymbol{u})(\boldsymbol{v}\cdot\boldsymbol{v})}$$

Chapter 4. Movement of guidewire and catheter

4.1 Introduction

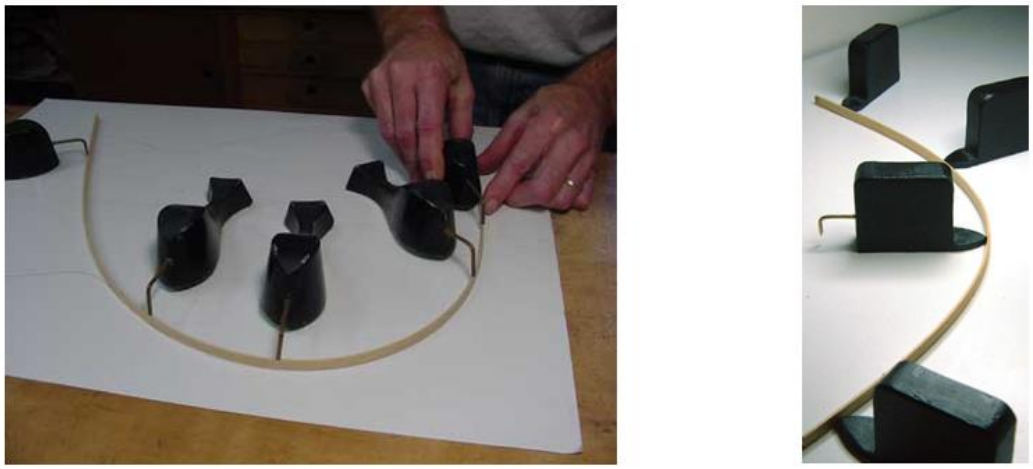


Figure 28. Splines made with weights

Spline is a Mid-1700s East Anglian dialect that means long thin piece of metal or wood, or a flexible strip of metal or other material. Before computers were used for design, spline was used with metal weights to make curves. They changed the shape of a curve by moving metal weights. For more control of the shape, they put more weights on spline. Splines also refer to a type of a curve. Splines can be expressed with mathematical functions. The first mathematical reference to splines was introduced in the paper by Schoenberg in 1946. It is believed that the paper used the word "spline" first time in connection with smooth, piecewise polynomial approximation. However, it was mainly for the aircraft and shipbuilding industries.

For the mathematical function of splines, polynomials are used because they are computationally efficient and easy to work with. However, often it is not possible to define a satisfactory spline with single polynomials. In order to define a satisfactory spline, the spline is broken into some number of pieces that are called segments. Each segment has its own polynomials. These segments form a piecewise polynomial curve. The splines are well presented in the book and most information on splines are obtained from the book [25].

The general form of polynomials is written as below.

$$y = a + bx + cx^2 + dx^3 + \cdots \qquad \text{Eq. 4.1}$$

The degree of a polynomial is the highest coefficient that is nonzero. For example, if c is nonzero, it is degree 2 that is called quadratic. The degree three polynomial, cubic polynomial , is the typical polynomial for splines in computer graphics because 1) it is the lowest degree polynomial that can support inflection points where the sign of curvature changes and 2) it shows smooth curves without sudden change of curvatures.

Since it is difficult to use single polynomials for a spline, the spline can be broken into segments which form a piecewise polynomial curve. For cubic piecewise polynomial curve, three continuities should be employed: C^0 continuity, C^1 continuity, and C^2 continuity. C^0 continuity is that the two segments have the same value at the join. With C^1 continuity, the two segments have the same slope at the join. C^2 continuity have the same curvature at the join. With these continuity conditions, the coefficients in the polynomial can be calculated.

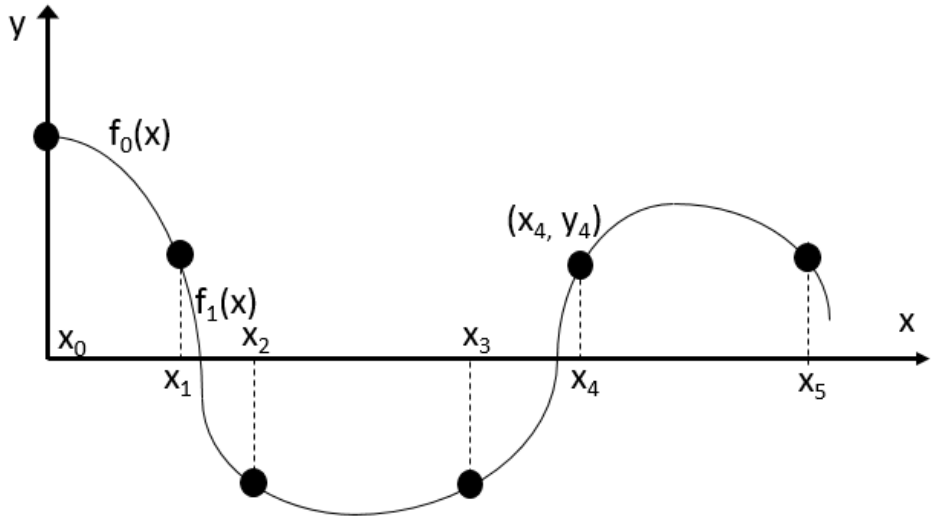


Figure 29. Polynomial curve

For example, the most generally used polynomial is a cubic polynomial.

$$f_i(x) = a_i + b_i x + c_i x^2 + d_i x^3 \qquad Eq. 4.2$$

Here, i represents a segment of a curve by the cubic polynomial. The segment is created between control points i and $i+1$. As discussed, we calculate polynomials segment by segment. $f_i(x)$ represents the curve between control points i and $i+1$.

With C^0 continuity, the cubic polynomial curve passes through all the control points.

$$f_i(x_{i+1}) = f_{i+1}(x_{i+1}) \qquad Eq.4.3$$

, where $x_{i+1}$is a joining point of the segments i and $i+1$.

C^1 continuity requires the same slope where two adjacent segments join together.

$$f'_i(x_{i+1}) = f'_{i+1}(x_{i+1}) \qquad Eq.4.4$$

Through C^2 continuity, one more equation can be produced to calculate the coefficients. The segments have the same curvature where they meet.

$$f''_i(x_{i+1}) = f''_{i+1}(x_{i+1}) \qquad Eq.4.5$$

Lastly, since there is not an adjacent segment at the end point, the slopes have to be assigned for the two end control points. As it is calculated with continuity conditions, the entire curve can be determined by one set of linear equation.

$$Ma = y \quad => a = M^{-1}y$$

, where the vector a is the set of coefficients to solve for, and the vector y is the set of known right-hand values of each equation and each row of the matrix M can be taken from the left-hand side of the linear equations. In this way, different types of spline can be solved easily.

4.2 Types of Spline

4.2.1 Natural Cubic Spline

A cubic spline is third-order polynomials as mentioned before. It passes through a set of m control points. The cubic spline function $S(x)$ with degree k on $[a, b]$ can be described as below.

$$S \in C^{k-1}[a, b]$$

$$a = t_0 < t_1 < \cdots < t_n = b$$

$$S(x) = \begin{cases} S_0(x), & t_0 \le x \le t_1 \\ S_1(x), & t_1 \le x \le t_2 \\ & \vdots \\ S_{n-1}(x), & t_{n-1} \le x \le t_n \end{cases} \qquad Eq.4.6$$

, where $S \in \mathbb{P}^k$

From the polynomial equation Eq. 4.2,

$$S(x) = \begin{cases} S_0(x) = a_0x^3 + b_0x^2 + c_0x + d_0, & t_0 \le x \le t_1 \\ S_1(x) = a_1x^3 + b_1x^2 + c_1x + d_1, & t_1 \le x \le t_2 \\ \vdots & \\ S_{n-1}(x) = a_{n-1}x^3 + b_{n-1}x^2 + c_{n-1}x + d_{n-1}, & t_{n-1} \le x \le t_n \end{cases}$$

, which satisfies continuities, C^0, C^1, C^2.

C^0 continuity: $S_{i-1}(x_i) = S_i(x_i), i = 1,2,\cdots,n-1$

C^1 continuity: $S'_{i-1}(x_i) = S'_i(x_i), i = 1,2,\cdots,n-1$

C^2 continuity: $S''_{i-1}(x_i) = S''_i(x_i), i = 1,2,\cdots,n-1$

,where

$$S \in C^2[t_0, t_n]$$

If we set the second derivative to zero, which means the spline is natural cubic spline,

$$S''(t_0) = S''(t) = 0$$

Set $z_i = S''(x_i), i = 0,\cdots,n$. We can say $z_0 = z_n = 0$.

We know $S(x)$ is a third order polynomial, so $S''(x)$ becomes a linear spline that interpolates (t_i, z_i). We are going to construct this linear spline $S''(x)$ first and then integrate to get $S(x)$.

The linear spline can be written as below.

$$S''(x) = z_i \frac{x - t_{i+1}}{t_i - t_{i+1}} + z_{i+1} \frac{x - t_i}{t_{i+1} - t_i} \qquad Eq.4.7$$

If we set $h_i = t_{i+1} - t_i, i = 0, \cdots, n$, then we get

$$S"(x) = z_i \frac{x - t_{i+1}}{h_i} + z_{i+1} \frac{x - t_i}{h_i} \qquad Eq. 4.8$$

Through integration, we can get $S(x)$.

$$S_i(x) = \frac{z_{i+1}}{6h_i}(x - t_i)^3 + \frac{z_i}{6h_i}(t_{i+1} - x)^3 + C_i(x - t_i) + D_i(t_{i+1} - x) \qquad Eq. 4.9$$

From the integrated equation above, we can get

$$S_i(t_i) = \frac{z_i}{6} {h_i}^2 + D_i h_i = y_i, i = 0, \cdots, n \qquad Eq. 4.10$$

Continuity produces

$$S_i(t_{i+1}) = \frac{z_{i+1}}{6} {h_i}^2 + C_i h_i = y_{i+1} \qquad Eq. 4.11$$

We replace C_i and D_i in Eq. 1.9 with C_i in Eq. 4.10 and D_i in Eq. 4.11.

$$S_i(x) = \frac{z_{i+1}}{6h_i}(x - t_i)^3 + \frac{z_i}{6h_i}(t_{i+1} - x)^3 + \left(\frac{y_{i+1}}{h_i} - \frac{z_{i+1}}{6} h_i\right)(x - t_i) + \left(\frac{y_i}{h_i} - \frac{h_i}{6} z_i\right)(t_{i+1} - x) \qquad Eq. 4.12$$

The derivative of Eq. 4.12 becomes

$$S'_i(x) = \frac{z_{i+1}}{2h_i}(x - t_i)^2 - \frac{z_i}{2h_i}(t_{i+1} - x)^2 + \frac{1}{h_i}(y_{i+1} - y_i) - \frac{h_i}{6}(z_{i+1} - z_i) \qquad Eq. 4.13$$

We set $\frac{1}{h_i}(y_{i+1} - y_i) = b_i$

Then, we get

$$S'_i(t_i) = -\frac{1}{2} z_i h_i + b_i - \frac{h_i}{6} z_{i+1} + \frac{1}{6} h_i z_i$$

$$S'_i(t_{i+1}) = \frac{z_{i+1}}{2} h_i + b_i - \frac{h_i}{6} z_{i+1} + \frac{1}{6} h_i z_i$$

$$S_{i-1}(t_i) = \frac{1}{3} z_i h_{i+1} + \frac{1}{6} h_{i-1} z_{i-1} + b_{i-1}$$

$$S'_i(t_i) = S_{i-1}(t_i) \Rightarrow 6(b_i - b_{i-1}) = h_{i-1} z_{i-1} + 2(h_{i-1} + h_i) z_i + h_i z_{i+1} \qquad \text{Eq. 4.14}$$

We can calculate each variable as the following.

$$h_i = t_{i+1} - t_i, \qquad i = 0, \cdots, n-1$$

$$b_i = \frac{1}{h_i}(y_{i+1} - y_i), \qquad i = 0, \cdots, n-1$$

$$v_i = 2(h_{i-1} + h_i), \qquad i = 0, \cdots, n-1$$

$$u_i = 6(b_i - b_{i-1}), \qquad i = 0, \cdots, n-1$$

$$z_0 = z_n = 0$$

We get the tridiagonal system

$$\begin{bmatrix} v_1 & h_1 & & & & \\ h_1 & v_2 & h_2 & & & \\ & h_2 & v_3 & h_3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & \ddots & \ddots & h_{n-2} \\ & & & & h_{n-2} & v_{n-1} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_{n-2} \\ z_{n-1} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{n-2} \\ u_{n-1} \end{bmatrix}. \qquad \text{Eq. 4.15}$$

A standard method for solving linear equations is Gaussian elimination. For the tridiagonal system of equation, TriDiagonal Matrix Algorithm (TDMA) is used.

4.2.2 Hermite Cubic Spline

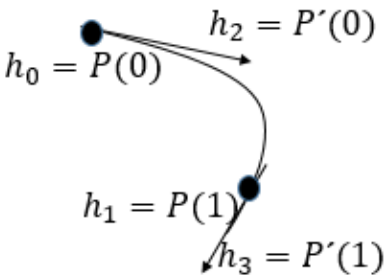


Figure 30. Hermite Spline

In order to draw Hermite Spline, a slope at each control point, $P(0)$ and $P(1)$ should be provided. The slope of at one control point is is tagent to the curve at control point, $P'(0)$ and $P'(1)$. From a cubic polynomial,

$$P(t) = at^3 + bt^2 + ct + d \qquad Eq.4.16$$

$$P'(t) = 3at^2 + 2bt + c \qquad Eq.4.17$$

We can get a matrix equation of Hermite Spline.

$$\begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \qquad Eq.4.18$$

Using inverse matrix properties,

$$\mathbf{a} = \mathbf{Mh}$$

$$\mathbf{h} = \mathbf{M^{-1}a}$$

$$\mathbf{I} = \mathbf{M^{-1}M} = \mathbf{MM^{-1}}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \qquad Eq.4.19$$

So, we get the inversed matrix representation.

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{bmatrix} \qquad Eq.4.20$$

Polynomials can be represented in a matrix form as below.

$$P(t) = [a \quad b \quad c \quad d] \begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix} \qquad Eq.4.21$$

$$P(t) = [a \quad b \quad c \quad d] \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix} \qquad Eq.4.22$$

$$\begin{bmatrix} H_0(t) \\ H_1(t) \\ H_2(t) \\ H_3(t) \end{bmatrix} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix} \qquad Eq.4.23$$

This matrix equation can be written in other way.

$$P(t) = [a \quad b \quad c \quad d] \begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix} = [h_0 \quad h_1 \quad h_2 \quad h_3] \begin{bmatrix} H_0(t) \\ H_1(t) \\ H_2(t) \\ H_3(t) \end{bmatrix} \qquad Eq.4.24$$

$$P(t) = \sum_{i=0}^{3} h_i H_i(t) \qquad Eq.1.25$$

4.2.3 Catmull-Rom Spline

The parametric cubic function is written as below.

$$P(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

With the two control points P0 and P1 and the slopes of the tangents P'0 and P'1 at each point, the parametric cubic function can be solved.

$$P(0) = a_0$$

$$P(1) = a_0 + a_1 + a_2 + a_3$$

$$P'(0) = a_1$$
$$P'(1) = a_1 + 2a_2 + 3a_3$$

From these equation, a_k values can be found.

$$a_0 = P(0)$$
$$a_1 = P'(0)$$
$$a_2 = 3[P(1) - P(0)] - 2P'(0) - P'(1)$$

$$a_3 = 2[P(0) - P(1)] + P'(0) + P'(1)$$

Putting these a_k values in the general polynomial form, we get as below.

$$P(t) = (1 - 3t^2 + 2t^3)P(0) + (3t^2 - 2t^3)P(1) + (t - 2t^2 + t^3)P'(0) + (-t^2 + t^3)P'(1) \qquad Eq.1.26$$

In a matrix form,

$$P(u) = [1 \quad t \quad t^2 \quad t^3] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \begin{bmatrix} P(0) \\ P(1) \\ P'(0) \\ P'(1) \end{bmatrix} \qquad Eq.4.27$$

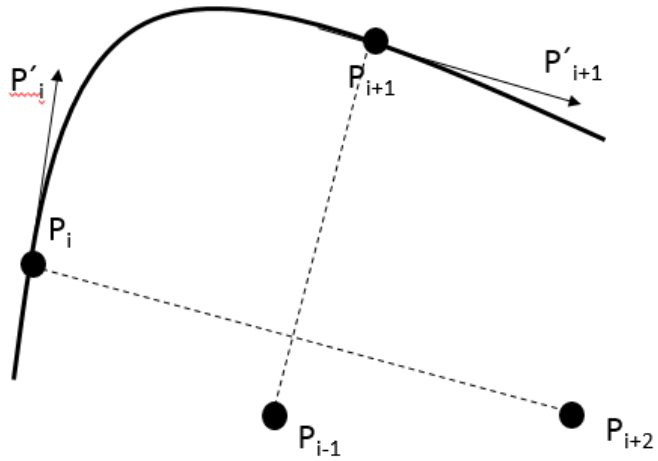


Figure 31. Catmull-Rom Spline Diagram

If we set the four control points as $P_{i-1}, P_i, P_{i+1}, P_{i+2}$ for a cubic polynomial curve, the tangent to the curve at each control point can be

$$\frac{P_{i+1} - P_{i-1}}{2} \; and \; \frac{P_{i+2} - P_i}{2}$$

Substituting these with $P'(0) and \; P'(1)$, we can get

$P(t)$

$$= [1 \quad t \quad t^2 \quad t^3] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} P_{i-1} \\ P_i \\ P_{i+1} \\ P_{i+2} \end{bmatrix} \qquad Eq. 4.27$$

Multiplying the two inner matrices, the linear equation is obtained.

$$P(t) = [1 \quad t \quad t^2 \quad t^3] M \begin{bmatrix} P_{i-1} \\ P_i \\ P_{i+1} \\ P_{i+2} \end{bmatrix} \qquad Eq. 4.28$$

, where

$$M = \frac{1}{2} \begin{bmatrix} 0 & 2 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 2 & -5 & 4 & -1 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

With this matrix linear equation, the cubic Catmull-Rom spline can be calculated. The cubic polynomial for the segment between control points P_i and P_{i+1} can be defined. Catmull-Rom spline passes through the control points, P_i and P_{i+1}.

4.2.4. Cubic B-Splines

The general form of cubic polynomial form can be written as below.

$$p(u) = c_0 + c_1 u + c_2 u^2 + c_3 u^3 = \sum_{k=0}^{3} c_k u^k \qquad Eq. 4.29$$

Each c_k is a column vector $\begin{bmatrix} c_{kx} & c_{ky} & c_{kz} \end{bmatrix}^T$. Different curves require different inputs to solve the cubic polynomial. For interpolation, 4 points are needed to solve for c_k.

To solve cubic B-Splines, m+2 control points are needed for m cubic segments. The computation is three times more expensive than simple interpolation. The cubic B-Splines are derived as follows.

$$C(u) = \sum_{i=0}^{n} N_{i,p}(u) P_i \qquad Eq.4.30$$

, where P_i denotes the control points. $n+1$ is the number of control points and $N_{i,p}(u)$ are B-spline basis functions of degree.

For example, a third order B-spline curve is defined as

$C(u) = \sum_{i=0}^{p} N_{i,3}(u) P_i \quad 0 \le u \le 1$

B-spline basis functions, $N_{i,p}(\mathrm{u})$, are defined recursively as:

$$N_{i,0}(\mathrm{u}) = \begin{cases} 1 & if\ u_i \le u \le u_{i+1} \\ 0 & otherwise \end{cases}$$

$$N_{i,p}(\mathrm{u}) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \qquad Eq.4.31$$

The basis function, $N_{i,p}(\mathrm{u})$, depend only on the value of p and the values in the knot vector.

$$\mathrm{U} = \{u_0, u_1, \dots, u_m\}$$

, where m+1 is the number of knots.

If the knots are equally spaced, they are called uniform. There are three parameters that change the shape of a uniform B-spline curve: the positions of control points, the positions of knots and the degree of the curve.

There are several different ways to change the shape of a B-spline curve. It can be achieved through the modification of the control parameters below.

- The positions of control points
- The positions of knots
- The degree of the curve

The forth degree function is used for uniform cubic B-spline.

$$N_{i,0}(\mathrm{u}) = \begin{cases} 1 & if\ u_i \le u \le u_{i+1} \\ 0 & otherwise \end{cases}$$

$$N_{i,4}(\mathrm{u}) = \frac{u - u_i}{u_{i+4} - u_i} N_{i,3}(u) + \frac{u_{i+5} - u}{u_{i+5} - u_{i+1}} N_{i+1,3}(u)$$

$C(u) = \sum_{i=0}^{3} N_{i,4}(u) P_i \quad 0 \le u \le 1$

With Cox-de Boor recursion formula, the basis functions can be calculated and the k-th segment can be written as

$$\mathrm{P(t)} = [1 \quad t \quad t^2 \quad t^3 \ ] M \begin{bmatrix} P_k \\ P_{k+1} \\ P_{k+2} \\ P_{k+3} \end{bmatrix} \qquad Eq.4.32$$

Where k = 0, 1, …, n-3 and 0 ≤ u ≤ 1, and where

$$M = \frac{1}{6} \begin{bmatrix} 1 & 4 & 1 & 0 \\ -3 & 0 & 3 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

4.2.5. Literature Review on Guidewire Model

A semantic guidewire model is introduced that consists of three parts: a catheter tip, a guidewire tip and a guidewire body [26]. There are challenges for tracking a guidewire in fluoroscopy. Guidewires are thin and consequently have low visibility in fluoroscopic images. A low dose of radiation in imaging makes even lower the visibility. Noisy images pose a challenge to track a guidewire. More challenges come from the shape deformation of a guidewire due to a patients' breathing and cardiac motions. Other wire-like structures make it difficult tracking a guidewire such as guiding catheters and ribs.

For guidewire tracking, a probabilistic framework is presented in this paper. The guidewire was modelled mathematically with a spline model. With M control points which are $x_i^c, i = 1, \ldots, M$, on a guidewire, the guidewire can be expressed with a set of points interpolated from the control points.

$$\Gamma(x) = \{x = (\gamma^x(\lambda), \gamma^y(\lambda)) | 1 \leq \lambda \leq M\}$$

, where $\gamma^x(\lambda)$ and $\gamma^y(\lambda)$ are cubic spline functions and $\lambda \in [i-1, i]$ means that x is interpolated between the control points x_{i-1}^c and x_i^c. This paper reports the spline representation reduces significantly the complexity of guidewire tracking.

A method is presented to extract and track the position of a guidewire during endovascular interventions under X-ray fluoroscopy [27]. The method improves guidewire visualization in the low quality fluoroscopy images. The guidewire was represented with a spline parameterization which is a third order B-Spline curve. It can be defined by:

$$\mathbf{C}(u) = \sum_{i=0}^{p} N_{i,3}(u)\mathbf{P}_i \qquad 0 \leq u \leq 1$$

, where $\mathbf{P}_i$ denotes the control points with p the number of control points and $N_{i,3}(u)$ are the third-degree B-Spline basis functions with the non-periodic knot vector

$$U = \{0, \dots, 0, u_4, \dots, u_{m-4}, 1, \dots, 1\}$$

, where m is the number of knots.

This paper suggests two step procedures to find the spline in frame $n+1$ with the position in frame n known. The first step is a rigid translation that captures the rough displacement of the spline. The second step is a spline optimization procedure for accurate localization of the guidewire.

A deformable B-spline tube model was presented, which represents the shape of a catheter and guidewire [28]. The B-spline tube model is different than other B-spline models used in other researches because it modifies knots instead of control points for B-spline representation.

The standard basis function of a B-spline curve of degree k-1 can be represented as:

$$b(t) = \sum_{i=1}^{n} C_i B_{i,k}(t)$$

, where $C_i \in \mathbb{R}^N$ are control points and $B_{i,k}(t)$ are basis functions that can be derived by the De Boor-Cox recursive formula for a total of n control points

A local cubic B-spline is a linear combination of local control points, $[C_{i-1} \quad C_i \quad C_{i+1} \quad C_{i+2}]$, that is defined based on the matrix representation for the De Boor-Cox formula.

$$b_i(t) = [t^3 \quad t^2 \quad t \quad 1] \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ \frac{1}{2} & -1 & \frac{1}{2} & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} & 0 \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 \end{bmatrix} \begin{bmatrix} C_{i-1} \\ C_i \\ C_{i+1} \\ C_{i+2} \end{bmatrix}$$

The sampling parameter $t \in [0,1]$ is uniformly distributed for interpolation. The knot points are determined as below with $t = 0$.

$$K_i = b_i(0) = \frac{1}{6}C_{i-1} + \frac{2}{3}C_i + \frac{1}{6}C_{i+1}$$

The cubic B-spline curve can be represented with the knot points instead of the control points.

$$\begin{bmatrix} C_1 \\ \vdots \\ C_n \end{bmatrix} = \begin{bmatrix} 1 & & & \cdots & & 0 \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & & & \\ & \ddots & \ddots & \ddots & & \vdots \\ & & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & \\ \vdots & & & \ddots & \ddots & \ddots \\ & & & & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ 0 & & \cdots & & & 1 \end{bmatrix}^{-1}_{n \times n} \begin{bmatrix} K_1 \\ \vdots \\ K_n \end{bmatrix}$$

II. Results

1. Overview of Coronary Angiography Learning System

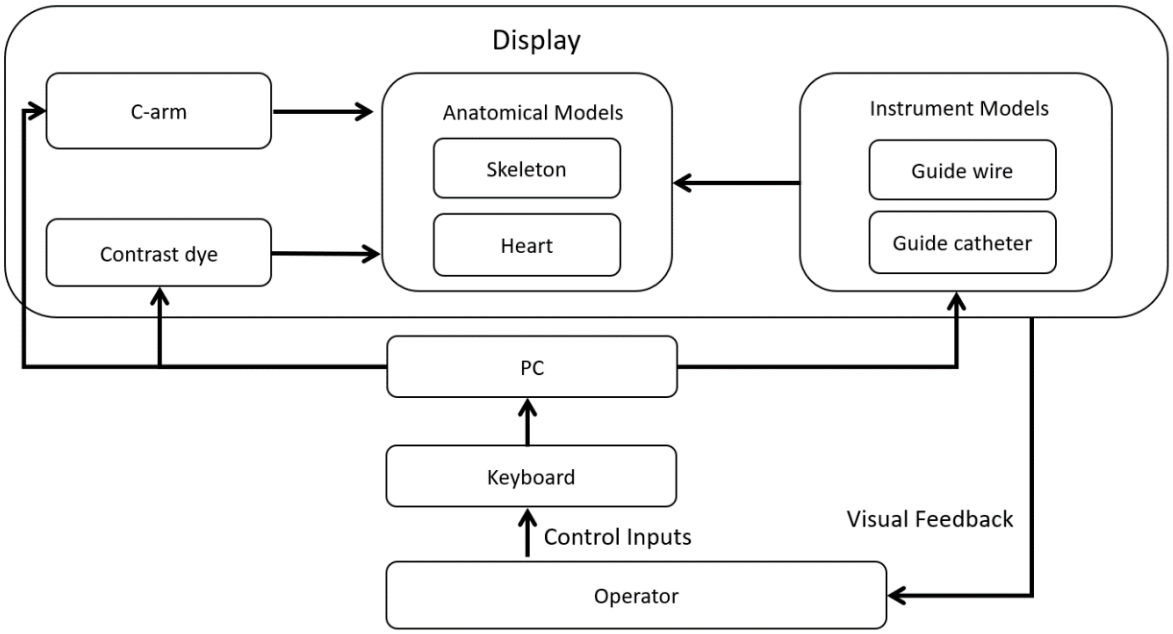


Figure 32. Schematic Diagram of Coronary Angiography Learning System

PC Specifications	
Processor	Intel® Core™ i7-4700MQ CPU @ 2.40 GHz
RAM	16.0 GB
System type	64-bit operating system, x64-based processor
Operating System	Windows 10 Home
Display adapters	Intel® HD Graphics 4600

Table 3. PC Specifications

Most CA simulators are composed of four components: Display, Controller, Interface and Computer. As it was mentioned before, this study focuses on the display and computer components. An operator interacts through a keyboard with the coronary angiography learning system. The keyboard delivers inputs from the operator to PC so that the system simulates C-arm, contrast dye injection, guidewire and catheter accordingly.

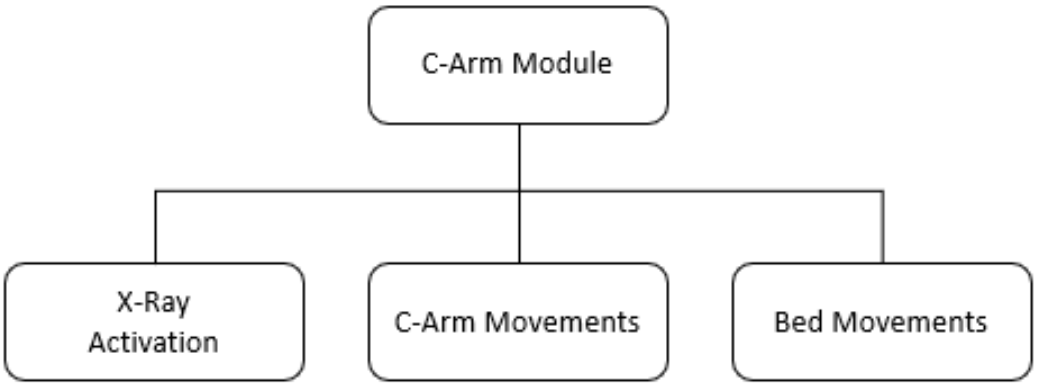


Figure 33. C-Arm module functionality

The C-arm module is for angiography imaging. By controlling the C-arm, the operator may have different angiographic views. As it was discussed before, depending on the coronary artery, a specific angiographic view is required. The C-arm module also simulates an activation of x-ray as the operator moves or manipulates guidewire or catheter and injects contrast dye to examine coronary arteries. During coronary angiography, the patient bed had to be moved for a better angiographic view. This bed movements were simulated by panning images.

The contrast dye module simulates the injection of contrast dye. The module visualizes coronary arteries. It also builds a coronary artery with stenosis according to input parameters of segment of coronary artery and severity of stenosis. As the heart beats, the coronary arteries move as the heart moves.

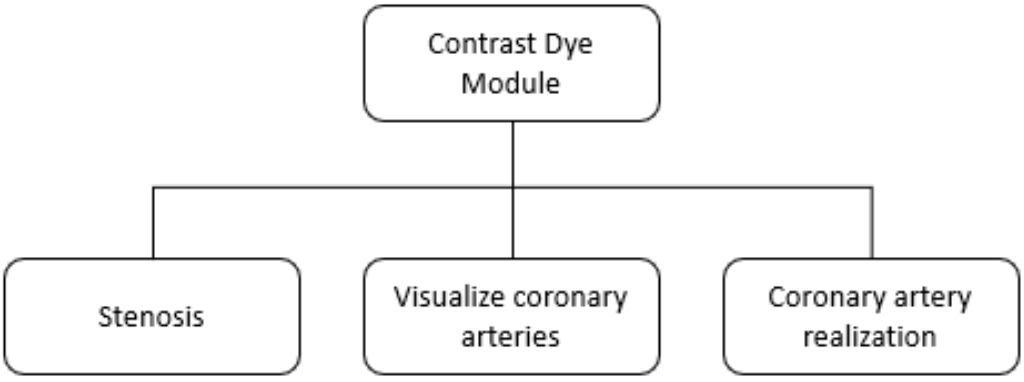
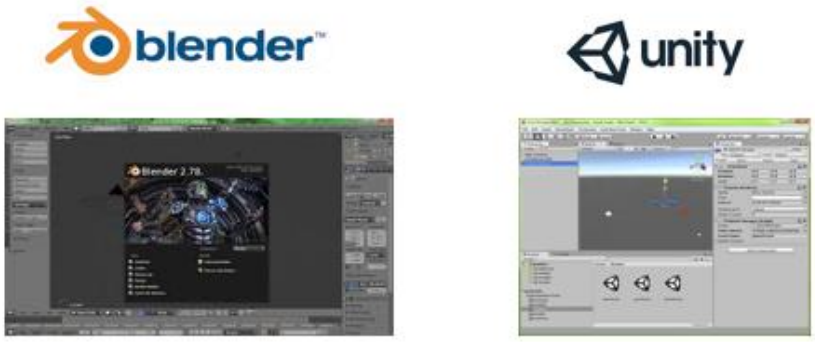


Figure 34. Contrast dye module functionality

Anatomical model module includes models of heart and skeleton. The heart beating is animated by this module.

Instrumental model module contains guidewire and catheter. It simulates guidewire and catheter in the aorta.



(a) (b)

Figure 35. Blender and Unity logos and their initial screenshots: (a) blender (b) unity

2. Implementation

2.1 Anatomical Model

In order to reconstruct coronary arteries graphically, Blender(v2.79) was employed. Blender is a free and open-source 3D computer graphics software. It can be used for 3D modeling, animations, visual effects, interactive 3D applications. It is based on Python programming language. For the implementation of the simulation, Unity was used. It is a cross-platform game engine which can be used for video games, simulations for computers, consoles and mobile devices. Scripts can be written in C# and JS for the game engine.

Unity is a multi-platform game engine. A game engine is the software that helps game developers with many features for the game environments. It means that game developers do not need to write programs from the scratch. And multi-platform means that games that is developed with Unity can be run in many platforms such as iPhone, Windows, iOS, Android and so on. With Unity, there are many possibilities to run the game. As the implementation of the learning system is

presented, you will recognize the great potential that Unity has for simulation training.

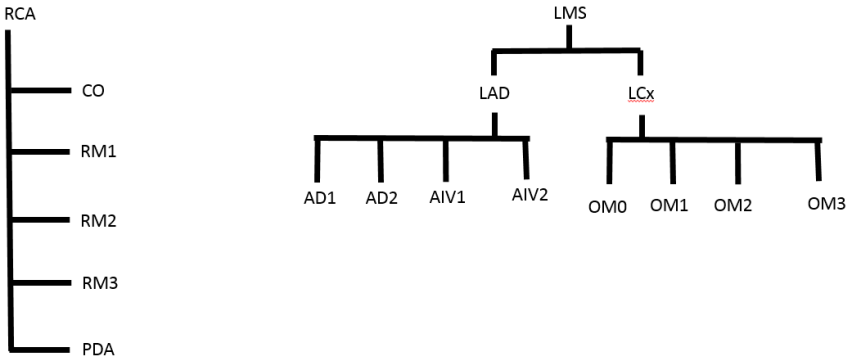


Figure 36. Coronary Artery Tree

A human heart anatomy file in a stl format was downloaded at a site, thingiverse.com [29]. The site provides digital design files. The file is licensed under the GNU General Public License under which end users are free to run, study, share and modify the file. This model was chosen because it shows coronary arteries on the surface.

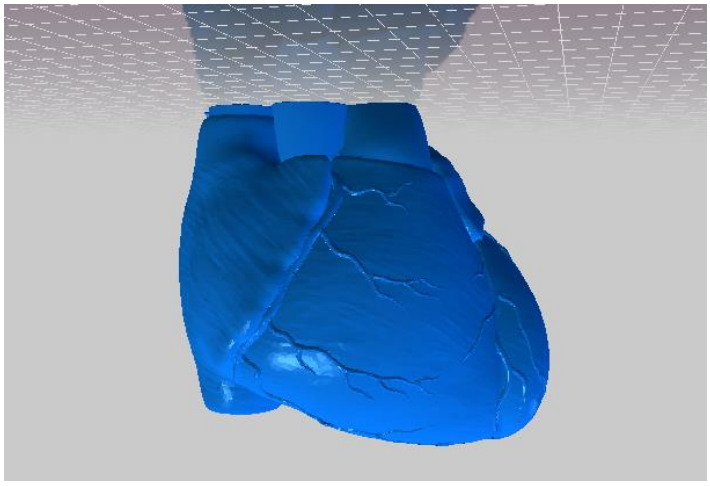


Figure 37. the heart anatomy file downloaded from thingiverse.com

After the file was imported to Blender, each coronary artery was identified. Each artery was connected according to the coronary artery tree. The centerlines of the coronary arteries were delineated manually in the middle of the coronary arteries on the surface of the heart. For this, several scripts were written in Python programming language in order to automate the manual process such as getting 3D coordinates of each vertex, saving them as an obj file, drawing curves and converting

curves into meshes. Since the arteries have different lengths, the curves contain different number of vertices. Each curve was interpolated with MATLAB. The MATLAB code was downloaded from a MathWorks forum [30]. The code interpolates the points of a curve in 3 dimensions. It produces evenly spaced points along the same curve. This code provides several options for interpolation: linear, pchip (for arc length), spline and csape (for a closed curve). The arteries were interpolated in spline to make them look smooth and it is the method that may be the most accurate according to the author. The interpolation makes easier to control the simulation and smooth curves without any sudden variation. The long coronary arteries such as LCX, LAD, and RCA have 50 points from the interpolation. Other coronary arteries have 25 points.

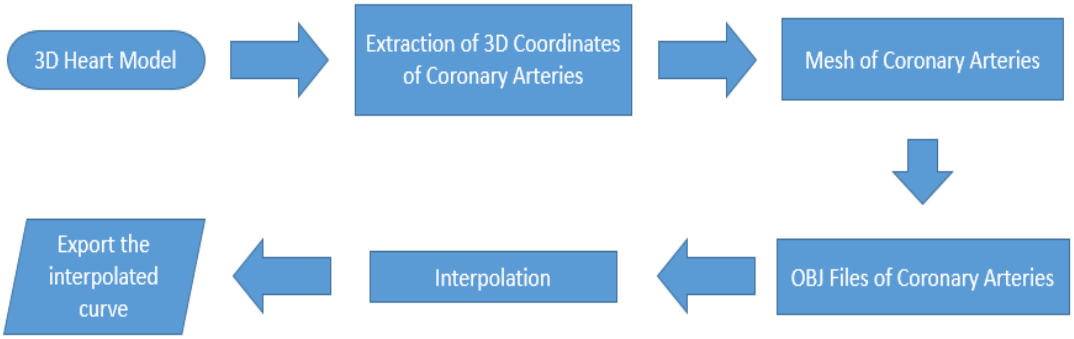


Figure 38. Flow Chart of 3D modeling

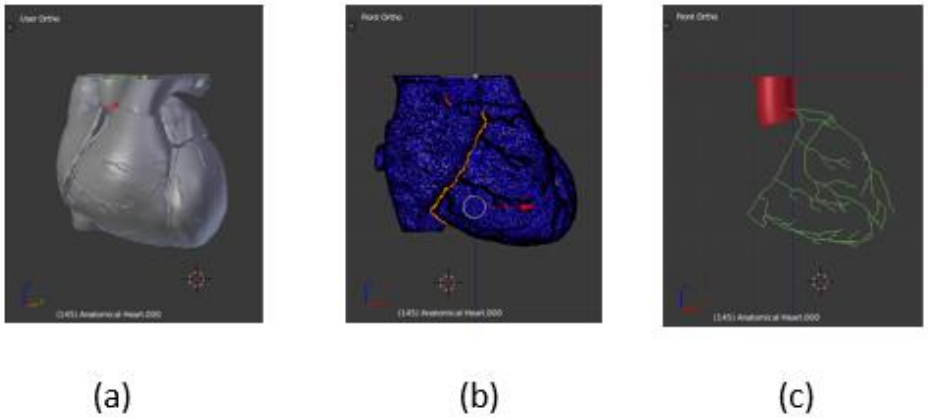


Figure 39. 3D model of a heart: (a) download 3d heart model (b) selected vertices for the right coronary artery(RCA) (c) meshed curves of coronary arteries

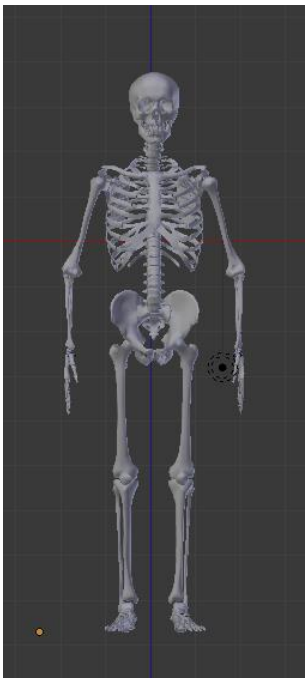


Figure 40. Skeleton 3D Model

Skeleton 3D model was obtained from a site, free3d.com [31].

2.2 Contrast Dye Module

For the implementation of contrast agent injection, unity was used. First, the 3D interpolated curves were imported. The curves function as arteries through which contrast agent flows. To operate the simulation, the simulator needs inputs. The position of the image intensifier can be set with the horizontal and vertical sliders. The horizontal slider moves the image intensifier laterally either toward the right of toward the left of the patient. Under the slider, the position of the image intensifier is indicated with an angle. For example, the position can be RAO 45.

The vertical slider tilts the image intensifier either toward the head of the patient (cranial) or toward the feet of the patient (caudal). On the left of the slider, the position of the image intensifier is indicated with an angle. For example, the position can be Cranial 20. The arrow keys pan the image as if the x-ray table moves laterally according to the arrow keys' directions.

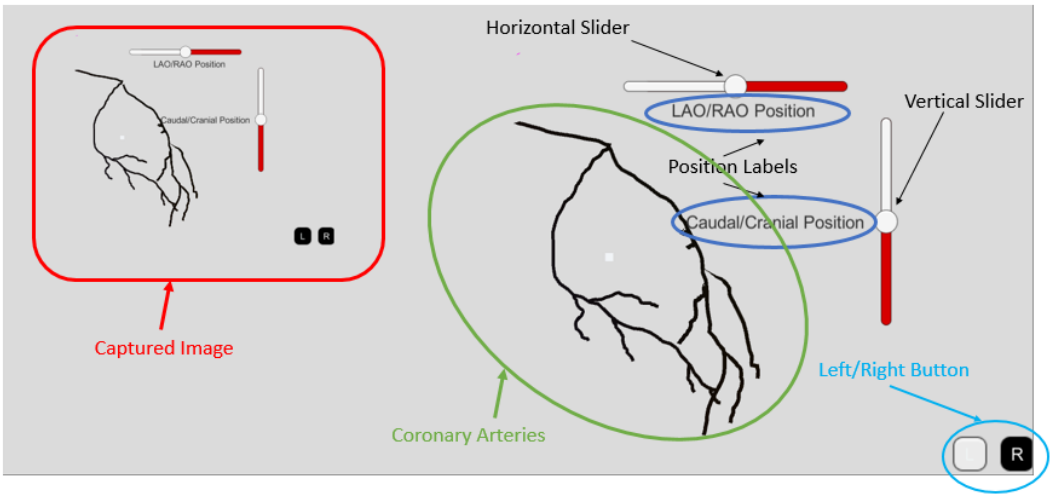


Figure 41. Simulator's objects

There are two buttons on the right bottom of the simulator that are labeled L/R individually. The L button, when clicked, simulates the contrast agent injection into the left main artery. The R button, when clicked, simulates the contrast agent injection into the right coronary artery. Once the injection button is pressed, the flow of contrast agent is simulated and the moving lines appear and grow longer. They appear on the screen for 5 seconds and disappear as the agent flows away. The left mouse click captures the screenshot and saves it in a file. The screenshot is displayed on the left top of the simulator.

As it was mentioned, the simulator is for the injection of contrast agent after the catheter is positioned for LMS or RCA. The L or R button starts the injection of contrast agent. From the origin, either RCA or LMS, the flow of contrast agent starts so black line appears and elongates as the contrast agent flow through either LMS or RCA. As the flow of contrast agent reaches bifurcation site, the additional line appears and starts a new flow of contrast agent in a new vessel. At the bifurcation site, in order to start a new flow of contrast agent, collision detection algorithm was employed.

Unity offers powerful rigid body physics engine. One of the physics features is collision detection. Sphere collision detection was used to initiate the new flow of

contrast agent for the new vessel. In unity, there are two variables for sphere collider: center and radius. Center is the center of the sphere that is located in the local space. Radius is the radius of the sphere measured in the local space. Sphere collision detection needs several parameters such as size of sphere, continuous or discrete detection.

As the contrast agent is injected to LMS, the contrast agent flow to the bifurcation point and the sphere from LMS collides with the sphere located at the beginning point of LAD and the sphere located at the beginning point of LCx. The collisions with LAD sphere and LCx sphere initiate the flow simulation of LAD and LCx simultaneously and individually. In this way, as the contrast agent flows in the left coronary arteries and the sphere collides with different spheres from different coronary arteries according to the coronary artery trees, coronary arteries whose sphere collided with the collide of a higher order coronary artery appear and grow longer. The sphere collision works in the same way for the right coronary arteries.

Objects	Functions
Horizontal Slider	Moves the image intensifier laterally (RAO/LAO)
Vertical Slider	Moves the image intensifier vertically (Cranial/Caudal)
Left Button	Injects the contrast agent into the left main artery
Right Button	Injects the contrast agent into the right coronary artery
Left Mouse Click	Captures a screenshot and saves a file of the screenshot
Arrow Keys	Pans the angiographic view

Table 4. Objects and their functions of the simulator

The radius of sphere has to be the same as the radius of corresponding coronary artery. If the diameter is too big, the new flow starts before the original flow reaches the bifurcation site, which leaves a discontinued line. If the radius is too small, the collision does not happen and the new flow does not start.

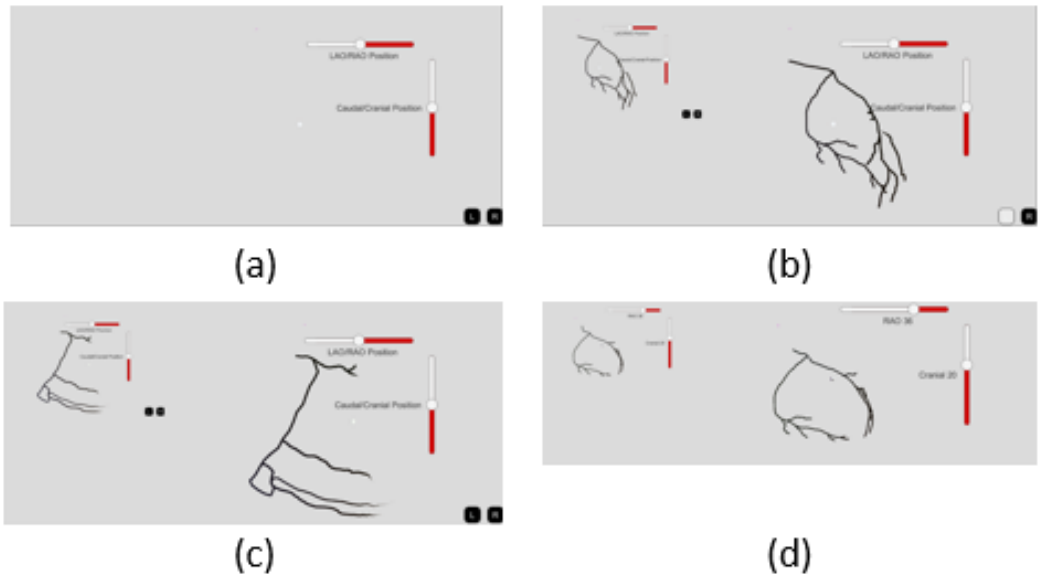


Figure 42. The Simulator at Work: (a) initial screen (b) Left Coronary Artery (c) Right Coronary Artery (d) Left Coronary Artery at RAO 36 and Cranial 20

This contrast dye injection is simulated statically, which means it does not simulate the beating heart. The beating heart is simulated through the periodic scale changes of the heart object. As the scale of the heart changes, the scales of the coronary arteries changes at the same frequency.

The number of the coronary arteries is not as big as the real image of coronary arteries. Since the purpose of the coronary angiography is to show the blood flow in the coronary arteries and to find any blockage in the coronary arteries. The blockage does not happen all the arteries. The most important arteries are large arteries such as LMS, RCA, LCX and LAD since they supply oxygenated blood to other sub arteries.

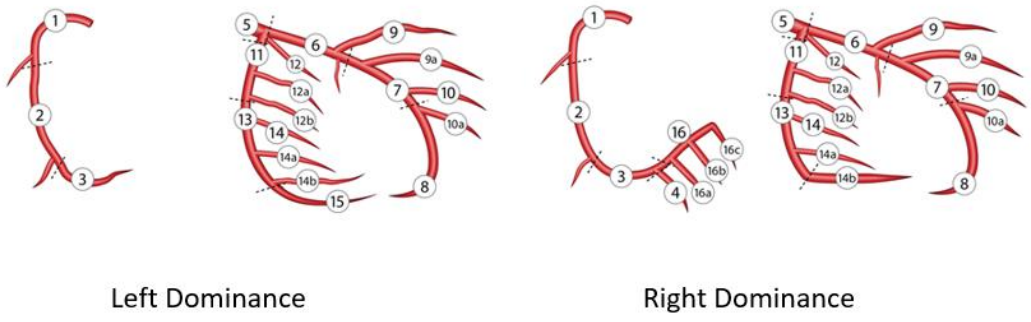


Fig. 43 Coronary Tree

According to the AHA classification, the coronary arteries are divided into 16 segments . For the assessment of coronary stenosis, there are four categories of lesion severity [32].

1. Minimal or mild CAD, narrowings<50%
2. Moderate, stenosis between 50% and 75%
3. Severe, stenosis between 75% and 95%
4. Total occlusion

In the website of SYNTAX score, the coronary tree and the assessment are explained very well [33].

1	RCA proximal	From ostium to one half the distance to the acute margin of the heart.
2	RCA mid	From end of first segment to acute margin of heart.
3	RCA distal	From the acute margin of the heart to the origin of the posterior descending artery.
4	Posterior descending	Running in the posterior interventricular groove.
16	Posterolateral from RCA	Posterolateral branch originating from the distal coronary artery distal to the crux.
16a	Posterolateral from RCA	First posterolateral branch from segment 16.
16b	Posterolateral from RCA	Second posterolateral branch from segment 16.
16c	Posterolateral from RCA	Third posterolateral branch from segment 16.
5	Left main	From the ostium of the LCA through bifurcation into left anterior descending and left circumflex branches.
6	LAD proximal	Proximal to and including first major septal branch.
7	LAD mid	LAD immediately distal to origin of first septal branch and extending to the point where LAD forms an angle (RAO view). If this angle is not identifiable this segment ends at one half the distance from the first septal to the apex of the heart.
8	LAD apical	Terminal portion of LAD, beginning at the end of previous segment and extending to or beyond the apex.
9	First diagonal	The first diagonal originating from segment 6 or 7.
9a	First diagonal a	Additional first diagonal originating from segment 6 or 7, before segment 8.
10	Second diagonal	Second diagonal originating from segment 8 or the transition between segment 7 and 8.
10a	Second diagonal a	Additional second diagonal originating from segment 8.
11	Proximal circumflex	Main stem of circumflex from its origin of left main to and including origin of first obtuse marginal branch.
12	Intermediate/anterolateral	Branch from trifurcating left main other than proximal LAD or LCX. Belongs to the circumflex territory.
12a	Obtuse marginal a	First side branch of circumflex running in general to the area of obtuse margin of the heart.
12b	Obtuse marginal b	Second additional branch of circumflex running in the same direction as 12.
13	Distal circumflex	The stem of the circumflex distal to the origin of the most distal obtuse marginal branch and running along the posterior left atrioventricular grooves. Caliber may be small or artery absent.
14	Left posterolateral	Running to the posterolateral surface of the left ventricle. May be absent or a division of obtuse marginal branch.
14a	Left posterolateral a	Distal from 14 and running in the same direction.
14b	Left posterolateral b	Distal from 14 and 14 a and running in the same direction.
15	Posterior descending	Most distal part of dominant left circumflex when present. Gives origin to septal branches. When this artery is present, segment 4 is usually absent.

Table 5. Segment Definitions

The simulator can provide variety of coronary angiography simulations. For example, each simulation can be run with parameters for the location of blockage and the severity of blockage. The parameters can be a segment number and a severity. With those parameters entered by an operator, abnormal coronary arteries may be created with a narrow diameter in a specified coronary artery. In this way, abnormal coronary arteries can be created also with other pathological

characteristics. The database of coronary arteries may be built easily and used for training to find the blockage and other anomaly due to the blockage. A training program can be developed so that trainees can pinpoint arteries to address and come up with treatments.

I would like to describe how I implement the contrast dye injection with Unity. The previous method used a linerederer that draws lines between static points. So, it does not simulate coronary arteries that supply oxygenated blood to a beating heart. First, the mesh of the heart is set as a child of four bones which are located at the centers of each atrium and ventricle. So, as the scales of the bones changes periodically, the scales of the parts of the heart also changes respectively with the same frequency.

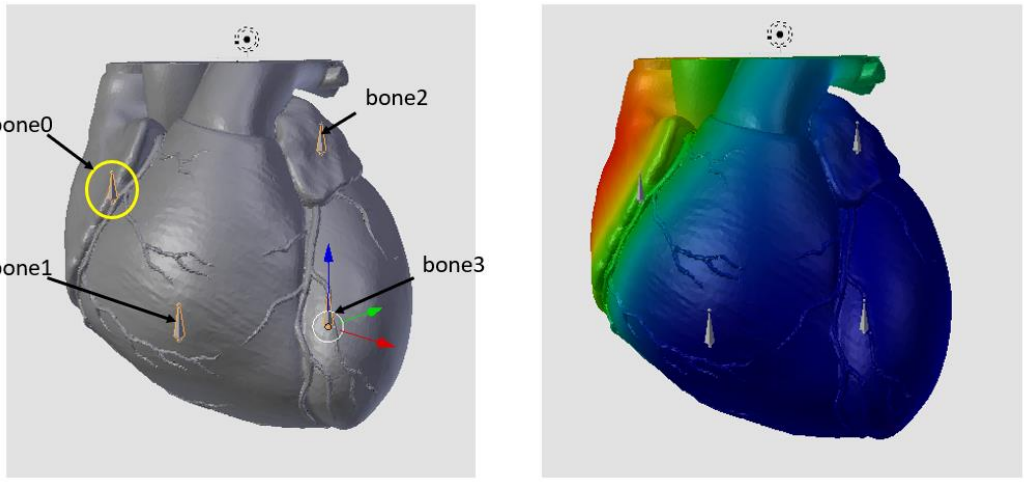


Figure 44. Making Heart Beat (a) Setup with bones (b) Scale Weight

In Figure 44, (a) shows how the heart was set up with bones. There are four bones and each bone control the scale change of each part. The bone0 controls the scale of the right atrium and the bone1 the right ventricle. The bone2 controls the scale of the left atrium and the bone3 the left ventricle. (b) in Figure 44 shows the scale weight. The color represents how much each bone's scale change affects parts of the heart. Red represents a weight 1 which means the part of the heart is affected 100% by the scale change of the bone. Blue represents a weight 0, which means no effect by the scale change of a bone. (b) shows the scale weight by bone1 that

controls the right atrium. The part around the right atrium is red so the part's scale change replicates the scale change of the bone1, but, other parts, left atrium, right ventricle and left ventricles are not affected by the scale change of the bone1. So, each bone changes its own scale with the same frequency can simulates a beating heart.

Figure 45 shows the images of a beating heart at specific frames. The number in the upper right of the image is the frame number. The beating heart animation consists of 100 frames. The animation is played at 24 fps.

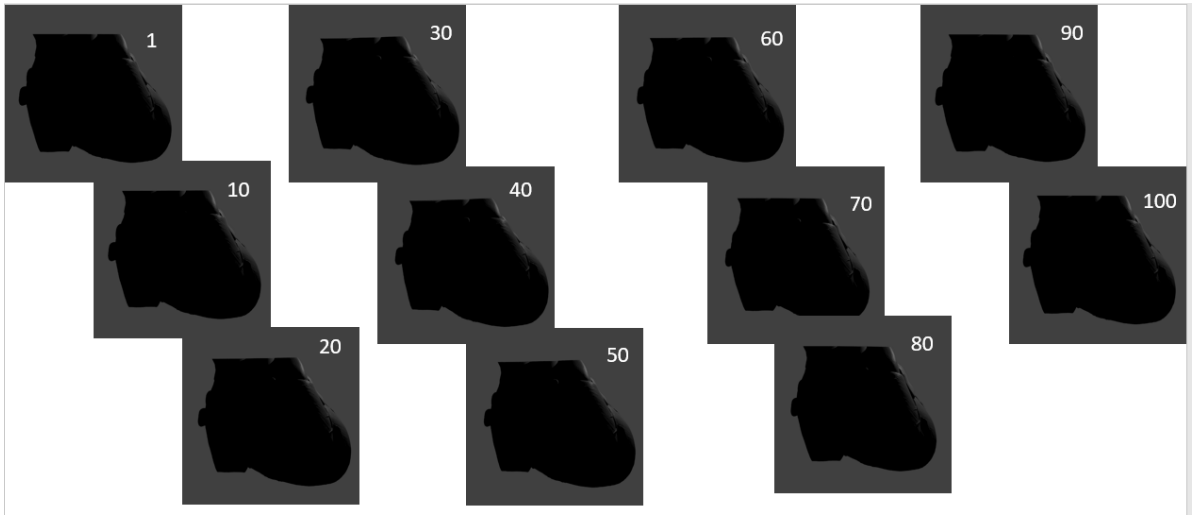


Figure 45. Beating Heart Animation

In the same method as the beating heart is animated, coronary arteries are animated with bones and scale weights. Each coronary artery consists of 25 or 50 empty objects. At each frame, the line render draws a line through the empty objects. Each coronary has its own line render that initiates contrast dye injection simulation by the activation input from a parent artery. For example, RCA has multiple sub-arteries such as PDA and right marginal branch.

The stenosis is created by manipulating animation keys for a line renderer of a specific coronary artery. Each artery has a line renderer as a component with properties such as width, positions, color. The line renderer is controlled by a script

that set parameters for the width of the line renderer. Those parameters are width, end width, percentage stenosis, and time stenosis. The width parameter sets overall width of a coronary artery that the line renderer draws. The end width parameter set the width of the end of the artery. It is usually zero. The percentage stenosis parameter sets the severity of stenosis. The time stenosis parameter sets the place of stenosis between 0 to 1. If the time stenosis parameter is 0, the stenosis is located in the beginning of the artery and if it is 1, the stenosis is at the end of the artery. Therefore, it should be between 0 and 1.

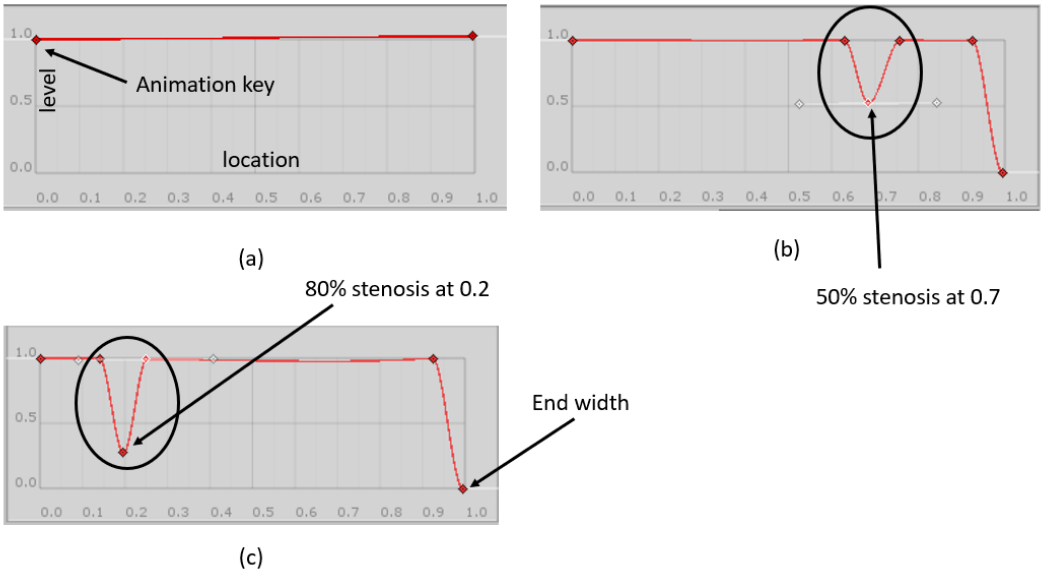


Figure 46. Creating stenosis in coronary arteries with animation keys

The width animation curve can have multiple number of animation keys. (b) in Figure 46 shows a width animation curve for an artery with 50% stenosis at 0.7. Half of the artery is blocked near the end of the artery. (c) shows 80% in the beginning of the artery.

2.3 Instrumental Model Module

For the simulation of guidewire and catheter, software Unity was used. As explained previously, a spline was used to simulate a guidewire and a catheter in an aorta during a catheterization procedure for coronary angiography.

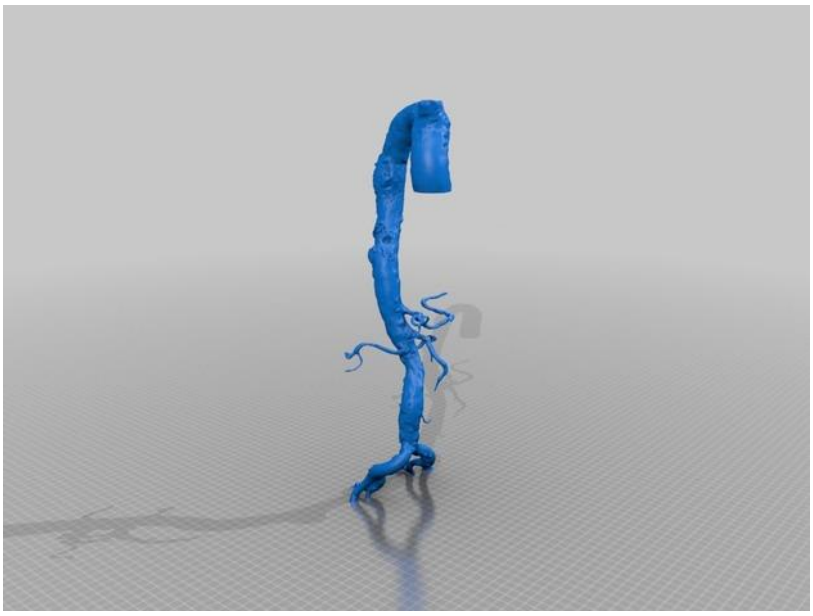


Figure 47. Aorta 3D model

I will discuss how a spline is constructed and simulates a guidewire with Unity. A 3D model of aorta was obtained in a stl format from a site, thingiverse.com [34]. The site provides digital design files. The file is licensed under the GNU General Public License under which end users are free to run, study, share and modify the file. From the model, a centerline of aorta can be obtained. In Figure 48, the centerline was drawn as a white line. Scripts for spline calculations were written in the C# script language.

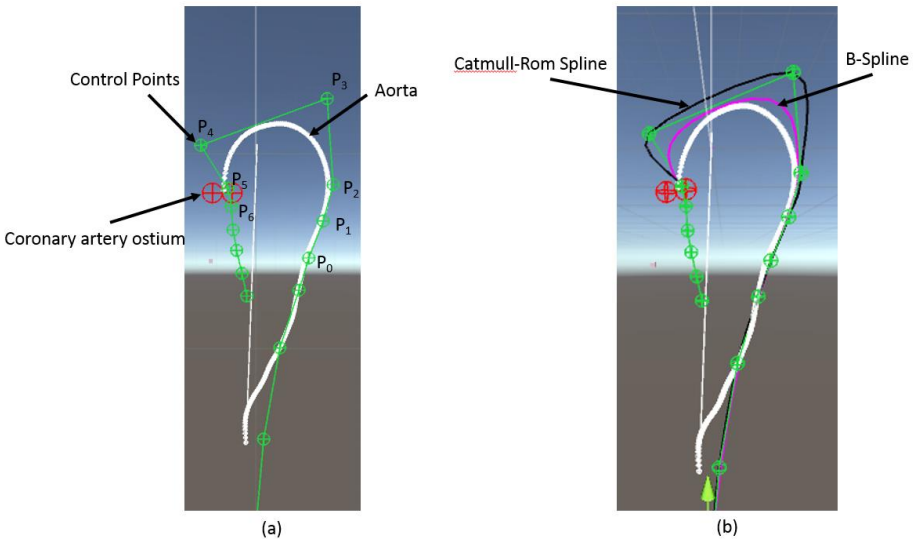


Figure 48. Simulation of Guidewire in Aorta (a) initial screen (b) simulated guidewire with Catmull-Rom spline and B-Spline

In Figure 48, there are control points that are green spheres. The white line represents aorta. The two red spheres are left coronary ostium and right coronary ostium. In the simulation, a guidewire moves along with the white line (aorta) and reaches red spheres (coronary ostium). For splines, 7 points were used to simulate the guidewire in the arch of aorta.

$$P = \{P_0, P_1, P_2, P_3, P_4, P_5, P_6\}$$

In order to move the guidewire, arrow keys are used. The up-arrow key represents an action of pushing the guidewire into the aorta. Down arrow key simulates an action of pulling the guidewire out of the aorta.

The following flowchart shows the algorithm of the spline calculation with those control points. Since splines require four control points to solve piecewise polynomials, the first check was to determine whether the current control point is either the first control point or the last control point.

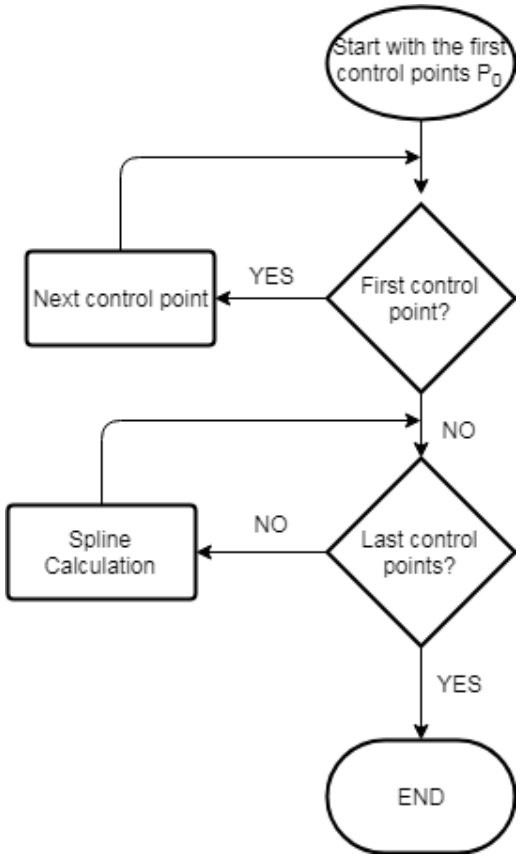


Figure 49. Flowchart of spline calculation

With 7 control points, there are 4 segments to calculate cubic polynomials for Catmull-Rom spline and B-Spline. Each segment requires four control points.

Segment0: $P = \{P_0, P_1, P_2, P_3\}$

Segment2: $P = \{P_1, P_2, P_3, P_4\}$

Segment3: $P = \{P_2, P_3, P_4, P_5\}$

Segment4: $P = \{P_3, P_4, P_5, P_6\}$

For each segment, a cubic piecewise polynomial can be calculated for Catmull-Rom spline with equation Eq. 4.28 and for B-Spline with Eq. 4.32.

$$P(t) = [1 \quad t \quad t^2 \quad t^3] M \begin{bmatrix} P_i \\ P_{i+1} \\ P_{i+2} \\ P_{i+3} \end{bmatrix}$$

For Catmull-Rom spline,

$$M = \frac{1}{2} \begin{bmatrix} 0 & 2 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 2 & -5 & 4 & -1 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

For B-Spline,

$$M = \frac{1}{6} \begin{bmatrix} 1 & 4 & 1 & 0 \\ -3 & 0 & 3 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

Each segment has a vector of coefficients for a cubic polynomial.

$$p(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3 = \sum_{k=0}^{3} c_k t^k$$

Each c_k is a column vector $\begin{bmatrix} c_{kx} & c_{ky} & c_{kz} \end{bmatrix}^T$.

Once a guidewire is positioned in the aortic root, a catheter is inserted to be located into the coronary ostium. Since a guidewire and a catheter have different mechanical properties, they move together in aorta differently than they move

alone. This behavior requires different spline for the path of a catheter. I will discuss the movements of a catheter along with a guidewire.

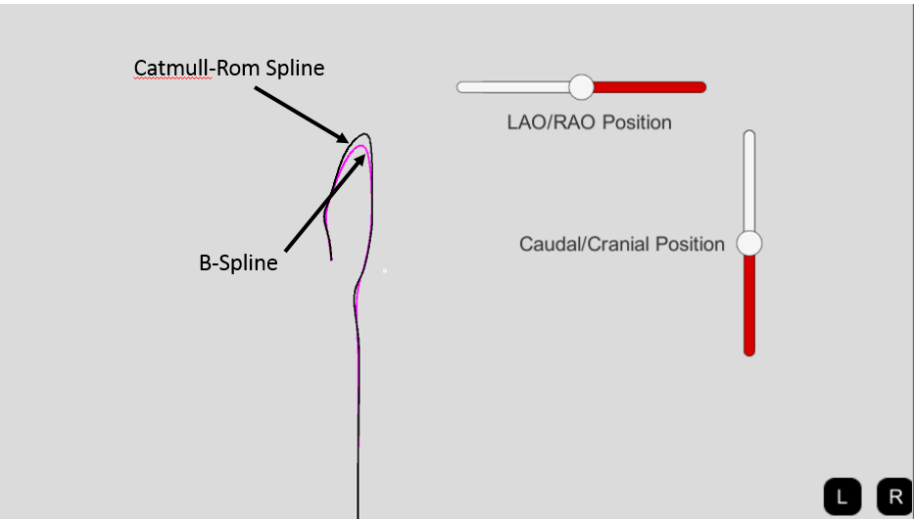


Figure 50. Simulated guidewire that reached the aortic root with Catmull-Rom spline and B-Spline

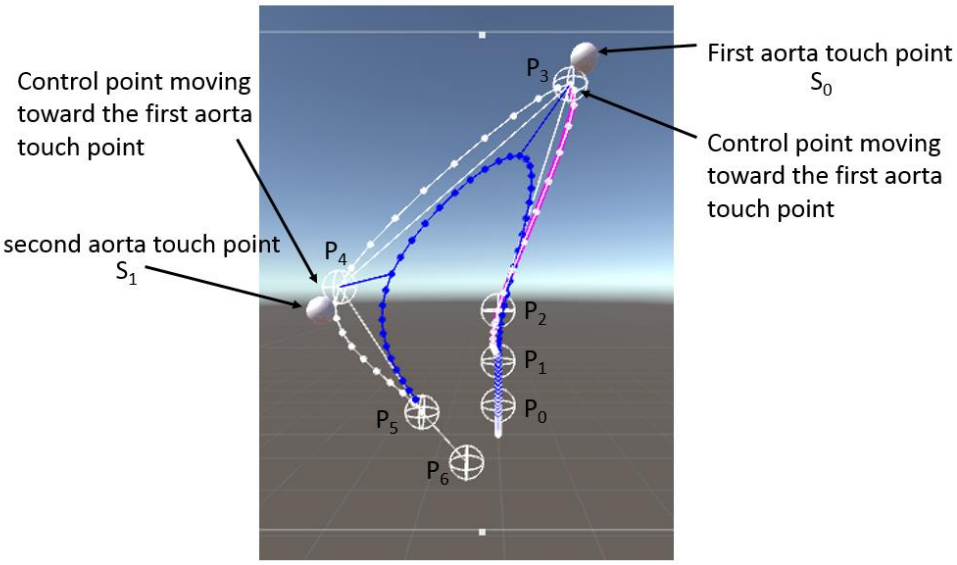


Figure 51. Splines for guidewire in arch of aorta

Figure 51 shows a guidewire which is simulated through Catmull-Rom spline. The white spheres are control points. The white line is Catmull-Rom spline with the control points. The blue line is B-Spline with the same control points. As a catheter advances into the arch of aorta along with a guidewire, the catheter and guidewire changes its shape of curve. To simulate the catheter bends at 45°, two

control points are used, P_3 and P_4. In Figure 12, there are two spheres, S_0 and S_1, that represents locations that the catheter touches the surface of the aorta. As the catheter touches the aorta surface, it bends at an angle. As the catheter advances to the aortic root, the control points move closer to those two spheres, S_0 and S_1. As the tip of the catheter touches the spheres, it makes an angle about 45°.

As mentioned, one way of modification of spline shape is to change the locations of control points. By moving the positions of the spheres, the shape of splines changes. Since Catmull-Rom spline passes through all the control points, it is relatively easy to make a desirable shape of spline. with the shape of spline modifications, the catheter movements can be simulated that moves over a guidewire.

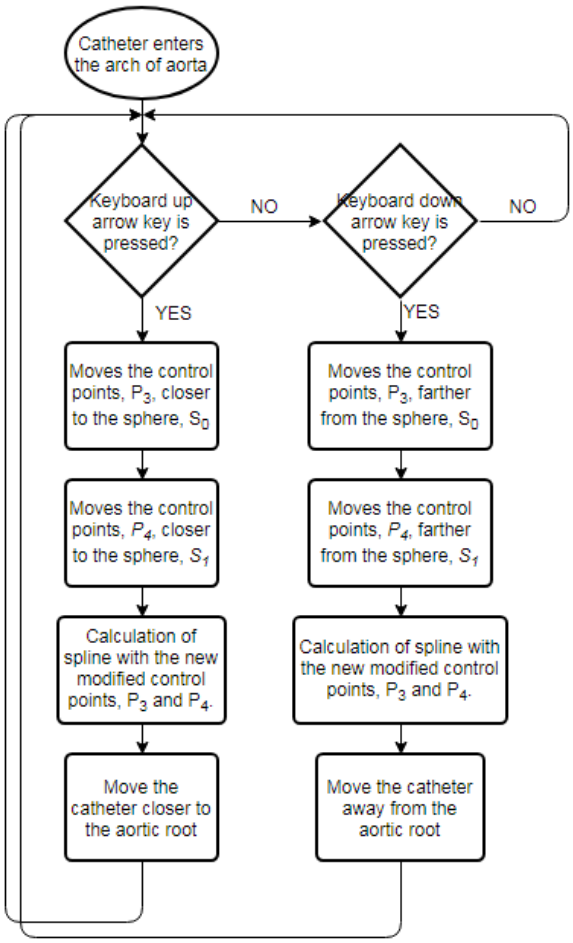


Figure 52. Flow Diagram for algorithm of spline for catheter movements in the arch of aorta

Figure 52 shows algorithm of simulation of catheter in the arch of aorta. The main control of the catheter and the guidewire is by the inputs from a keyboard, up or down arrow keys. It simulates pulling and pushing actions of an operator. The up-arrow key pushes the catheter and the guidewire into the aortic root and the down arrow key pulls away from the aortic root.

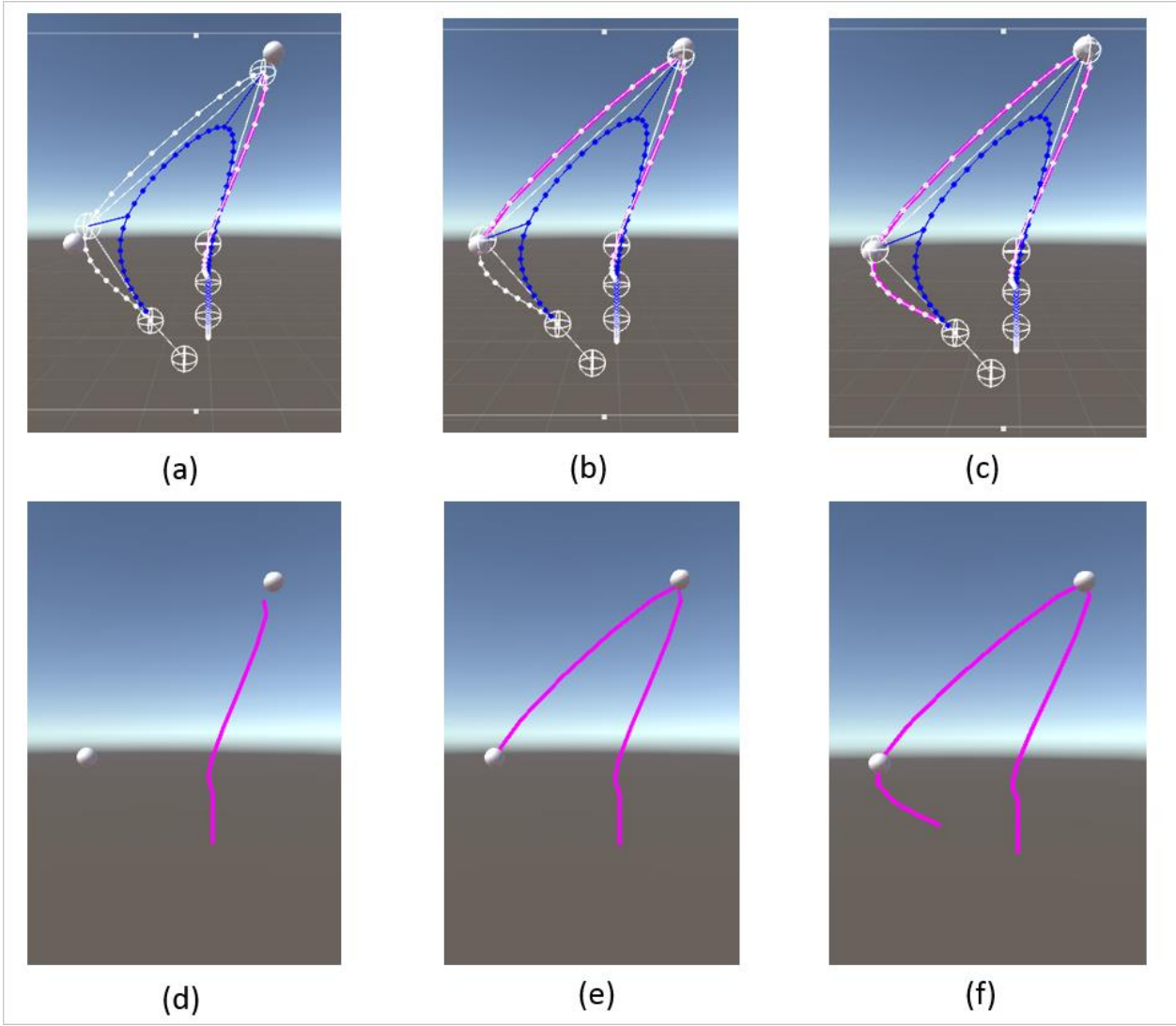


Figure 53. Simulated catheter in the arch of aorta. (a), (b), (c) are shown with control points. (d), (e), (f) are simulations of the catheter.

In Figure 53, the catheter, which is a pink line, advances to the aortic root from the descending aorta (a), through the arch of aorta (b) reaching to the aortic root (c). As the catheter moves closer to the aortic root, the control points, P_3 and P_4, moves closer to the spheres, S_0 and S_1. The movement of the catheter is

controlled with inputs from a keyboard. An up arrow key initiates the catheter movement toward the aortic root. It simulates pushing the catheter into the aorta. A down arrow key initiates the catheter movement away from the aortic root. It simulates pulling the catheter away from the aortic root.

As the guidewire is pulled out from the aorta, the catheter returns to the original shape, which is designed to locate the tip of the catheter into the coronary ostium.

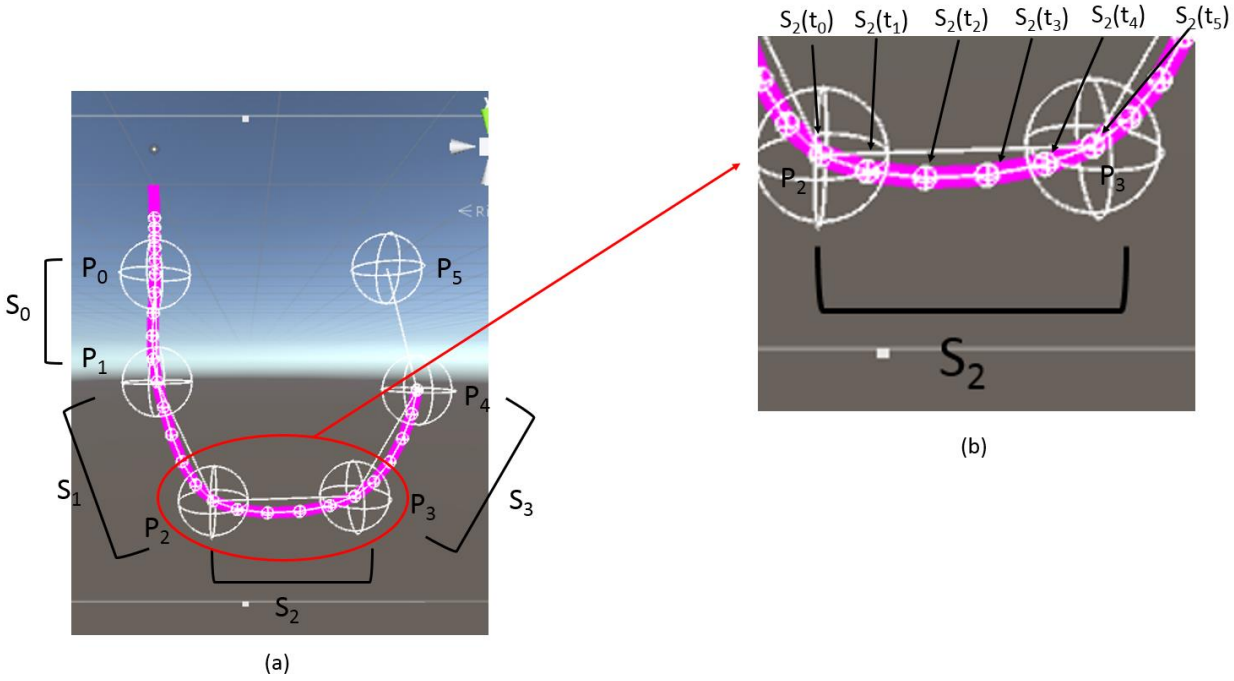


Figure 54. Guidewire and catheter spline diagram

In Figure 54, there are 6 control points that are used to calculate Catmull-Rom spline. I will discuss how piecewise polynomials are calculated. With the 6 control points, four segments were calculated. The Figure 15 (b) shows a segment, S_2, between the control points, P_2 and P_3.

The polynomial for S_2 can be written as below.

$$S_2(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \qquad 0 \le t < 1$$

The parameter t is discrete and increase by 0.2. That is, t_0=0, t_1=0.2, t_2=0.4 …

For the piecewise polynomial calculation, four control points are required. Using the matrix equation (1),

$$S_2(t) = [1 \quad t \quad t^2 \quad t^3] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix}$$

This matrix linear equation is simplified.

$$S_2(t) = [1 \quad t \quad t^2 \quad t^3] \begin{bmatrix} 2P_2 \\ -P_1 + P_3 \\ 2P_1 - 5P_2 + 4P_3 - P_4 \\ -P_1 + 3P_2 - 3P_3 + P_4 \end{bmatrix}$$

Since we know the vectors of those four control points, P_1, P_2, P_3 and P_4, $S_2(t)$ can be calculated at the point of t_0, t_1, t_2, t_3, t_4.

These days, games are not just 2D graphics with awkwardly moving objects. Thanks to the advancement of computer technology, games became more sophisticated and real in 3D graphics. Many simulation games simulate well the experiences of flying plane or the fight jets, driving cars. Unity has the physics system that simulations look real. The movement of an object and collisions between objects can be made in realistic ways by the physics system in the Unity game engine.

As the guidewire moves in the aorta, it does not collide with the surface as a ball hits the wall. Since the aorta is soft body, as the guidewire collides with the aorta surface, instead of bouncing off, it slides along with the surface of the aorta. It can be simulated by setting the parameter of bounciness to zero. In this way, the collision is made as the collision with a soft body which absorbs all the energy.

As it was mentioned, the guidewire was represented with multiple objects connected with spring joints. Basically, the guidewire was modelled with a mass-spring system. As it is shown previously, the movement of guidewire only and the movement of guidewire and catheter together are different. The guidewire moves along with cubic spline path, whereas the combination of guidewire and catheter

moves along with Hermite spline path. It can be simulated by changing the stiffness of the spring joints with high spring constant. With a low spring constant, the guidewire becomes more flexible and with high spring constant, it gets stiffer.

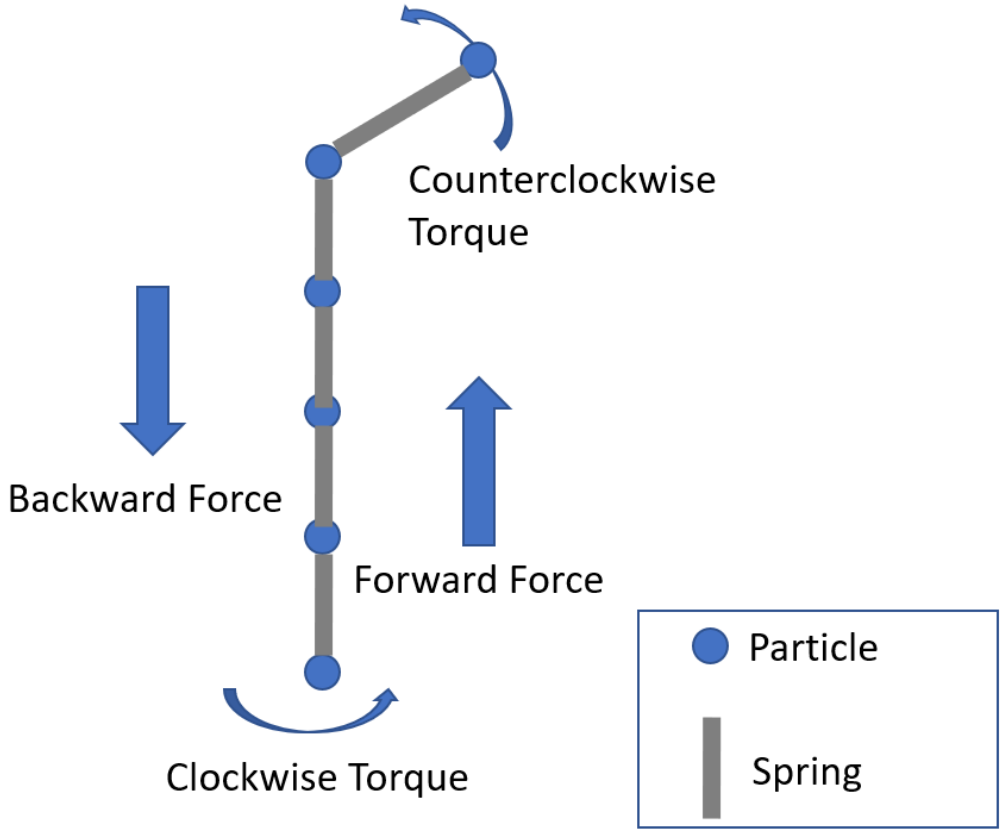


Figure 55 Guidewire with spring-mass model

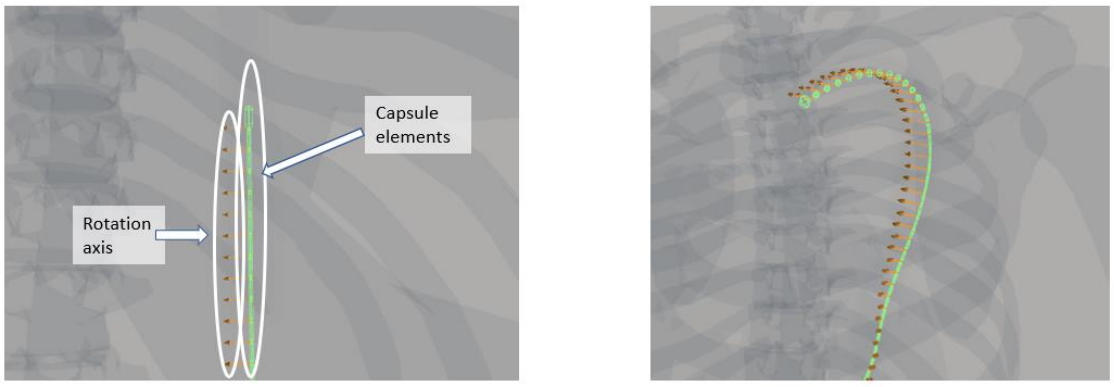


Figure 56 Simulated guidewire

In Figure 53, the simulated guidewire is shown. (a) shows the rotation axis around which each element rotates. The guidewire was simulated with capsule objects connected with springs. Each capsule objects were rigid bodies that follow the physics law according to the physics engine.

In the real coronary angiography, it is considered easy to engage catheter in the left coronary ostium. The process of catherization is shown in the Figure below.

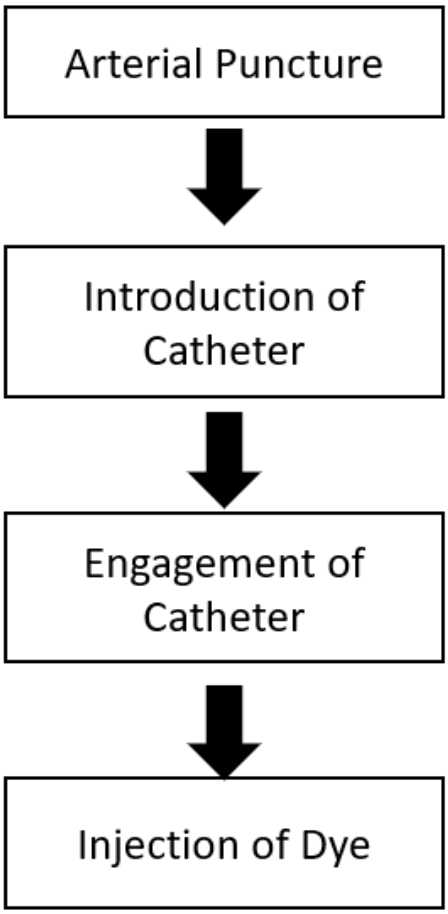


Figure 57. Catherization Process

The description of the catherization is based on the observations of coronary angiography at Chapidze Heart Center, which is located in Tbilisi, Georgia. The guidewire is introduced through an arterial puncture. And a catheter is introduced together with the guidewire which is already in an artery. The combination of the guidewire and the catheter advanced to the artic root. Once they confirmed through X-ray image the position of the guidewire and the

catheter, the guidewire is taken out. The catheter is manipulated to be engaged in the left or right coronary ostium.

The simulator is designed according to this process. The guidewire and catheter are controlled by inputs from a keyboard. The key “w” moves the guidewire and catheter toward the aortic root through the aortic arch. The key “s” moves backward the guidewire and catheter. The keys, “a” and “d” rotates the guidewire and catheter. These movements of the guidewire and catheter are simulated through the physics engine in Unity.

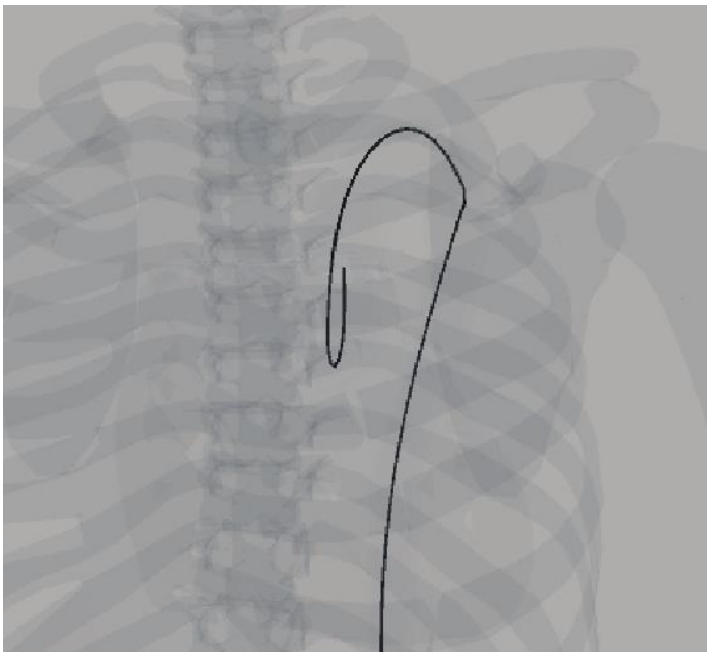


Figure 58. Simulated Guidewire in Aorta

The key “w” add force from one element of the guidewire to the other element that is joined through a spring. The key “s” add force in the opposite direction to the force applied by the key input “w”. The key “a” add torque in the direction of clockwise and the key “d” add torque in the counterclockwise direction. The movements of the catheter along the guidewire is controlled with the keys, “t” and “g”. in the image of X-ray, the guidewire is seen as a thick black line but a catheter is seen as a transparent tube. Only the longitudinal edges and the tip is shown. The key “t” moves the catheter toward the front tip of the

guidewire and the key “g” moves the catheter away from the front tip of the guidewire. Since a catheter has its own unique shape, as it moves along with the guidewire, it changes the shape of the line made by the guidewire. The combination of the guidewire and catheter becomes stiffer.

Inputs from Keyboard	Movements of Guidewire
W	Forward
S	Backward
A	Clockwise rotation
D	Counterclockwise rotation

Table 6. Control of guidewire movements

Inputs from Keyboard	Movements of Catheter
T	Forward
G	Backward
F	Clockwise rotation
H	Counterclockwise rotation

Table 7. Control of catheter movements

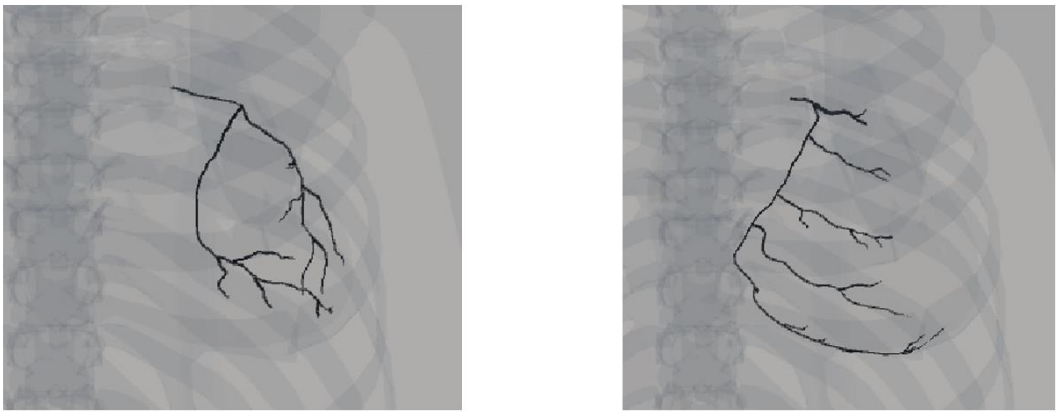


Figure 59. Simulated Dye Injection

Through the manipulation of the guidewire and catheter, catheter is engaged in the left coronary ostium or the right coronary ostium. The key “Shift” simulates the contrast dye injection.

Angiographic views can be controlled with inputs from a keyboard. The arrow keys simulates movement of C-arm and the keys “I”, “k”, “j”, “l” simulates moving the bed. And the keys “PageUp” and “PageDown” simulates zooming.

Inputs from Keyboard	Angiographic Views
Up Arrow	Cranial
Down Arrow	Caudal
Right Arrow	Clockwise rotation
Left Arrow	Counterclockwise rotation
PageUp	Zoom in
PageDown	Zoom out

Table 8. Control of C-arm

3. Serial Communication

With UNITY, the simulation of catheter and guidewire was implemented. They are controlled by keys from a keyboard. However, the control by inputs from a keyboard is not close to the control of a real coronary angiography procedure. It can give visual feedback and show the movements of guidewire and catheter in aorta, but it does not give tactile feedback. In the real coronary angiography, the inputs from an operator are analog inputs such as a moving catheter, a moving guidewire, a moving C-arm of X-ray image intensifier.

As previously mentioned, typical simulation system consists of four components: Display, PC, Control and Devices. The display unit displays simulated coronary angiography showing a catheter, a guidewire, x-ray images. The simulation is performed by PC unit. It is the unit that UNITY runs and simulates the moving

catheter and guidewire, x-ray images from x-ray intensifier. The inputs are from the control unit which will contain sensors for catheter, guidewire and an injection pedal. Devices are catheter, guidewire, syringe and so on.

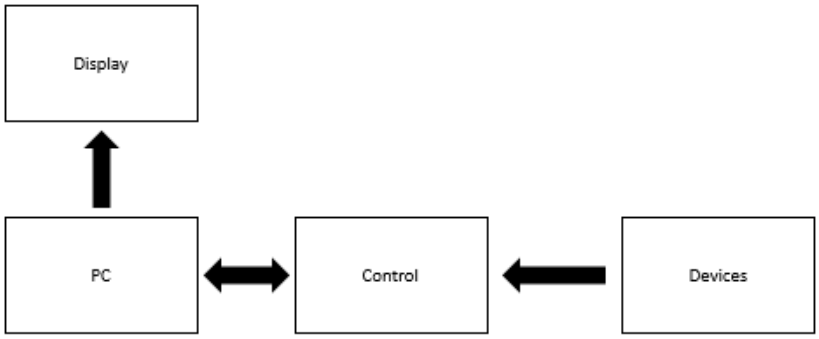


Figure 60 Simulation system schematic diagram

The coronary angiography learning system has two components: Display and PC. But, the system has capability of serial communication through which it can interacts with devices. The control component. It is ready for a haptic device that can give tactile feedback.

3.1 Arduino

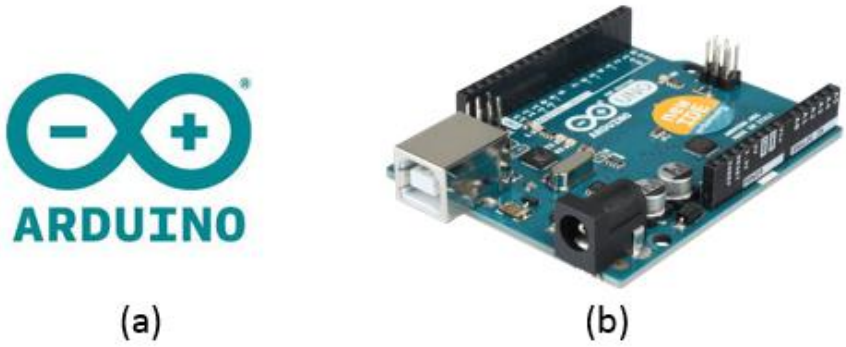


Figure 61. Arduino (a) logo (b) Arduino board

In order to connect the graphic simulation to the analog inputs, micro controller may be added as a control unit. Arduino was chosen for the control unit. Arduino is an open-source platform consisting of both a physical programmable circuit board and a piece of software for programming the physical board

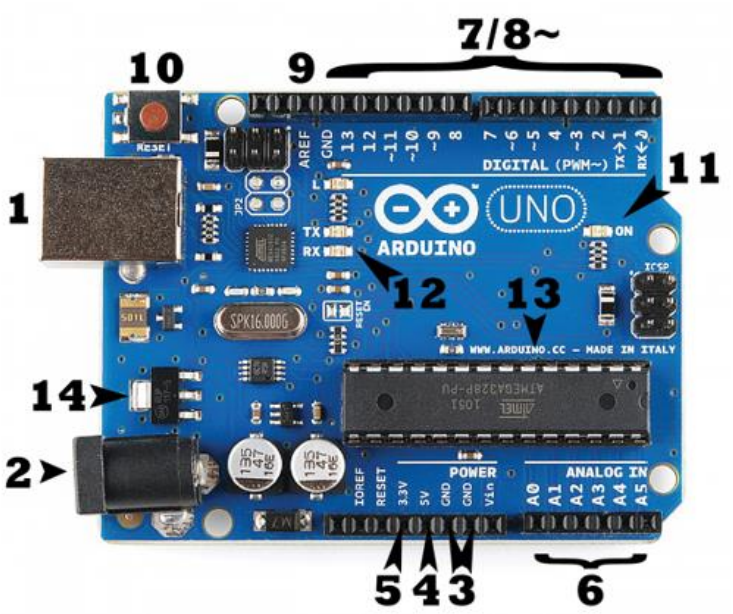


Figure 62. Arduino Uno structure: (1) USB connection (2) Power jack (3) Ground (4) 5 volts of power (5) 3.3 volts of power (6) Analog pins (7) Digital pins (8) PWM (9) Analog Reference (10) Reset button (11) Power LED indicator (12) TX RX LEDs (13) Main IC (14) Voltage Regulator

Figure 61 and 62 show Arduino circuit boards. The Arduino board needs to be connected to a power source. It can be powered from a USB cable coming from a USB cable through USB connection (1 in Fig. 59) or a power supply (2 in Fig. 59). Through USB connection, one can load code onto the Arduino board. There are several GNDs that are used to ground the circuit. The 5V pin supplies 5 volts of power and the 3.3v pins 3.3 volts of power. Arduino has analog pins that can read the signal from an analog sensor and convert it into a digital value. Digital pins can be used for digital input such as a button and output such as a LED. PWM pins act as normal digital pins but can also be used for Pulse-Width Modulation (PWM). AREF stands for Analog Reference. This pin is left alone most of time, but can be used to set an external reference voltage as the upper limit for the analog input pins. The reset button restarts any code that is loaded on the Arduino. It can be used when a code doesn't repeat. Power LED indicator tells you the Arduino is connected to a

power source. TX stands for transmit and RX for receive. TX RX LEDs gives visual indications of data transmission and reception from and to Arduino board.

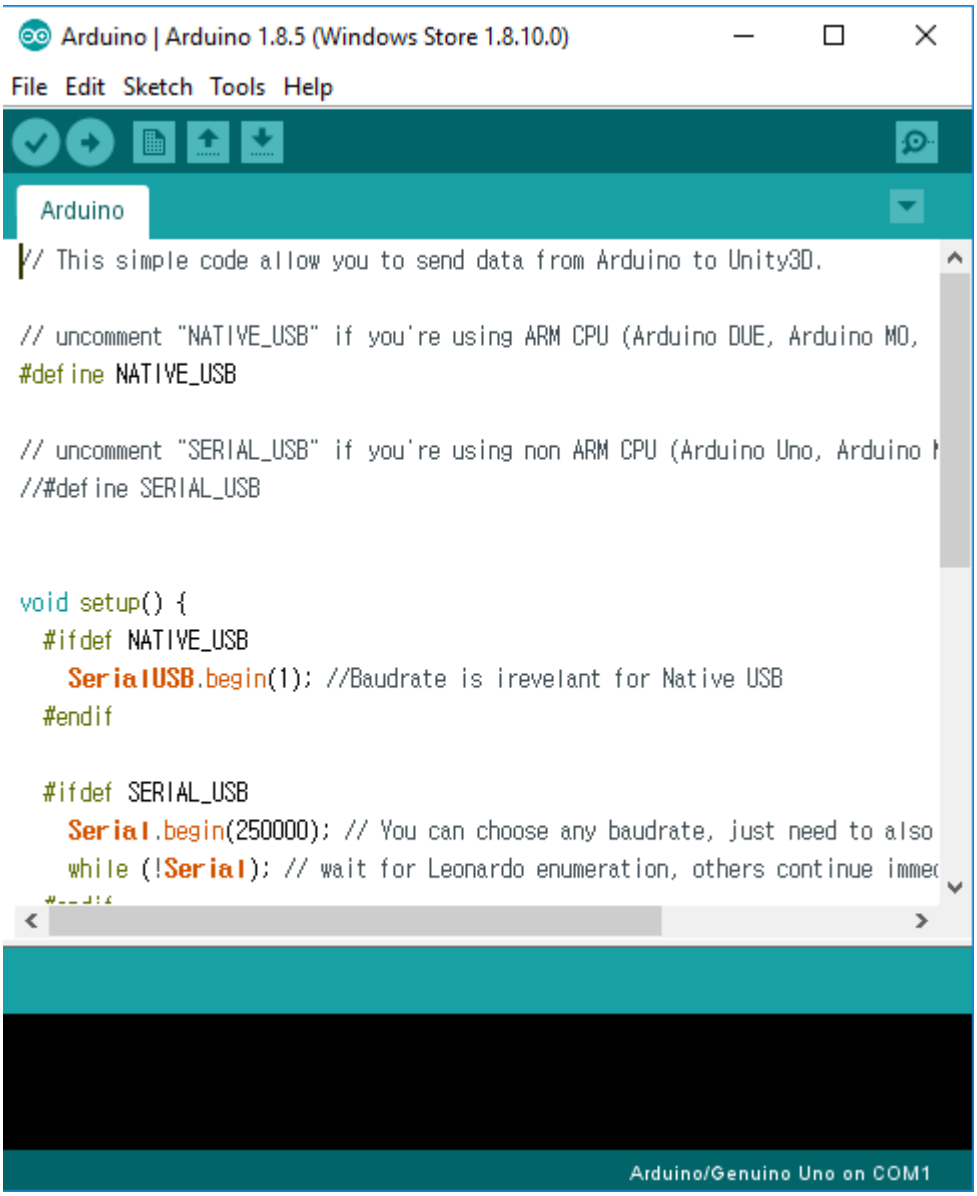


Figure 63. Arduino software screenshot.

The brain of Arduino is the main integrated circuit. The voltage regulator controls the amount of voltage that enter the Arduino board.

Arduino has its own open-source software (IDE) that makes it easy to write a code and uproad it to the board. Figure 60 shows the Arduino software for writing the code and uploading it.

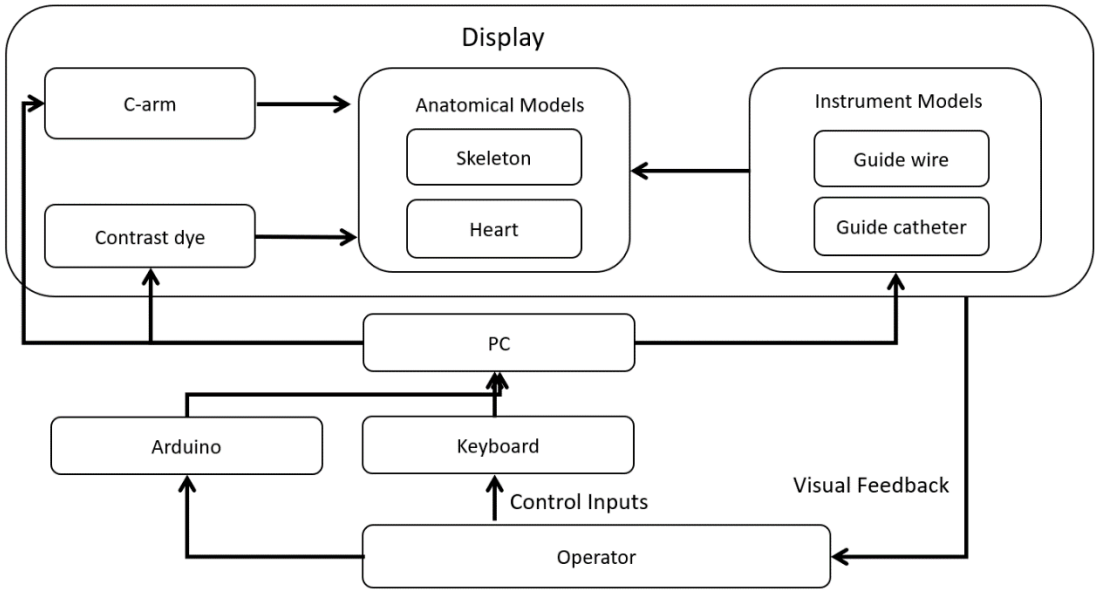


Figure 64. Schematic Diagram of Coronary Angiography Learning System with Arduino

Arduino controls the instrumental model module and the contrast dye module. Two 10k potentiometers move and rotate the instrumental model and a switch initiates the simulation of contrast dye injection.

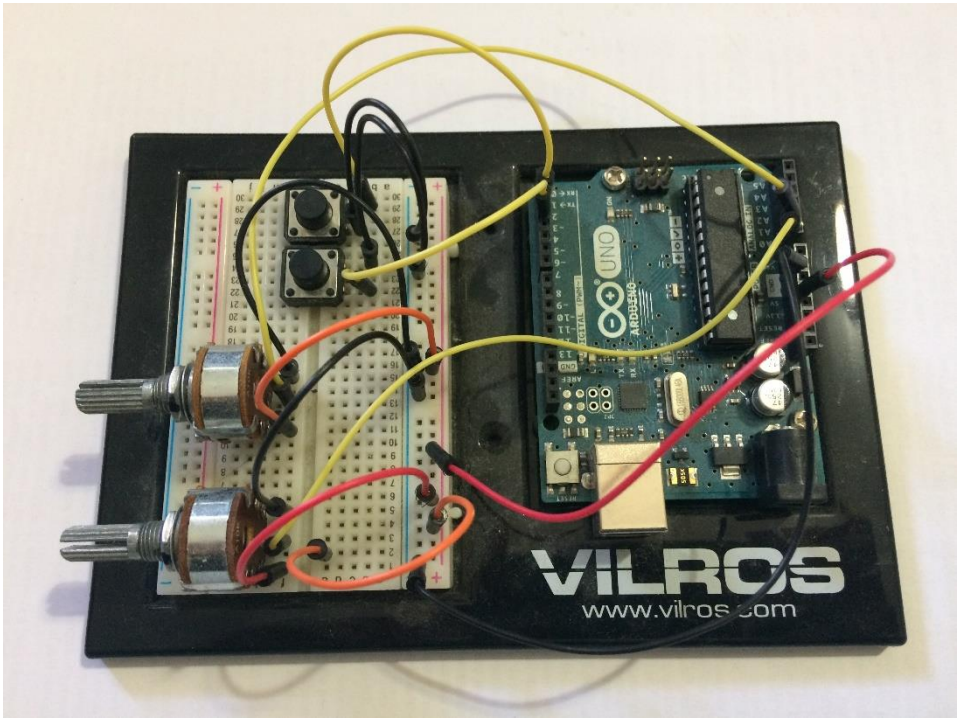


Figure 65. Arduino circuit for the control of the instrumental model module and the contrast dye module.

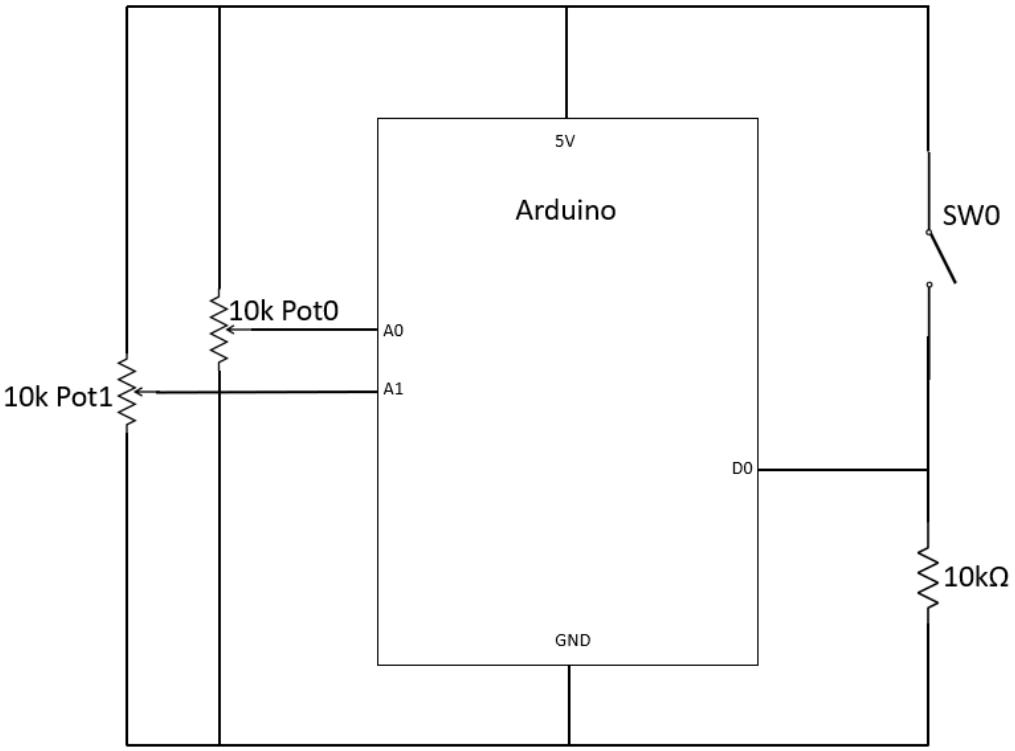


Figure 66. Arduino Circuit for Coronary Angiography Learning System

3.2. Integration of Arduino to Unity

For the communication between Arduino and Unity, an Unity package, wrmhl [35], was downloaded. The software package is under MIT license which permits commercial use, modification, distribution and private use.

The potentiometers give a number between 0 to 1023 to analog pins (A0 and A1). Based on the number from A0 or A1, the system controls the movements of the guidewire and catheter. And the digital pin (D0) receives the status of the switch (SW0) and if it is on, the system will activate the contrast dye injection.

With Arduino, the learning system works just same as with a keyboard. The catheter and guidewire movements and the contrast dye injection were simulated through the Arduino. So, it proves that this system can easily be expanded to external haptic device for tactile feedback. The commercial haptic device can be

added or the customized device that can detect movements of guidewire and catheter and give tactile feedback according to the movements.

III. Conclusion

In this paper, a coronary angiography learning system is presented. This system is designed and developed for simulation training. By the simulation of guidewire and catheter, contrast dye injection, and C-arm, the trainee can acquire skills for coronary angiography outside of a catherization laboratory.

The learning system was designed with a computer graphic software, Blender (2.79). And the simulations are implemented with a game engine, Unity3D, which also provide a physics engine for realistic movements of guidewire and catheter in aorta. Through the learning system, trainees will have more time to practice the procedure and quickly get familiar to the procedure.

The virtual experiences with the learning system will improve skills for coronary angiography and consistency of performance of coronary angiography and decrease errors. Consequently, it will decrease the time of training to reach a certain competent level in which a trainee can follow all the procedural steps unconsciously. The reduced training time will reduce the cost of training since the cost is proportional to the time.

However, the purpose of the simulation training is to transfer the effects of achieved skills through the simulation training to the real coronary angiography. The transferability from this coronary angiography learning system must be evaluated.

One way to increase the transferability of the skills is to add the capability of tactile feedback to this learning system. Currently, the learning system is capable of visual feedback only through display. For this reason, this learning system has capability of serial communication with external device. Once the control module, which can provide tactile feedback, is added to this learning system, the transferability would increase greatly.

Then, a question arises regarding the method of measuring the transferability and competent level. As it was discussed before regarding the assessment of coronary angiography simulator training, metrics have to be employed such as total procedure time, fluoroscopy time, catherization time for LCA and RCA and the volume of contrast dye. The metrics-based assessment of the skills must be combined together with the learning system.

Even though there are many researches and developments to be done to make the learning system better, it has still many applications for students, professionals and educators in various settings such as school, hospital, and simulation center.

References

[1] TOGNI M. Percutaneous coronary interventions in Europe 1992-2001. European Heart Journal 2004,25,1208–1213.

[2] http://vaughanheart.ca/coronary-angiography/ -03.05.17.

[3] iti. http://www.iti.com/blog/a-brief-history-of-simulation-training- -06.05.18.

[4] http://https://www.proflight.com/en/full-flight-simulatoren/historie.php - 05.06.18.

[5] Green SM, Klein AJ, Pancholy S, et al. The current state of medical simulation in interventional cardiology: A clinical document from the Society for Cardiovascular Angiography and Intervention's (SCAI) Simulation Committee. Cathet. Cardiovasc. Intervent. 2014,83,37–46.

[6] Jensen UJ, Jensen J, Olivecrona G, Ahlberg G, Lagerquist B, Tornvall P. The role of a simulator-based course in coronary angiography on performance in real life cath lab. BMC medical education 2014,14,49.

[7] Jensen UJ, Jensen J, Ahlberg G, Tornvall P. Virtual reality training in coronary angiography and its transfer effect to real-life catheterisation lab. EuroIntervention : journal of EuroPCR in collaboration with the Working Group on Interventional Cardiology of the European Society of Cardiology 2016,11,1503–1510.

[8] J. Kenneth. FDA Perpective on Medical Simulation-Based Training for Cardiovascular Devices 2005.

[9] Ran Bronstein SI, inventor. Medical Simulation Device with Motion Detector. US20070134637A1. 2007 Jun 14.

[10] Jan Grund-Pedersen, inventor. Interventional simulator system. US20060127867A1. 2006 Jun 15.

[11] Fitts, P. M., Posner, M. I. Human Performance. Oxford, England: Brooks/Cole 1967.

[12] Carraccio CL, Benson BJ, Nixon LJ, Derstine PL. From the Educational Bench to the Clinical Bedside: Translating the Dreyfus Developmental Model to the Learning of Clinical Skills. Academic Medicine 2008,83,761–767.

[13] Jensen U. The role of simulator training for skills acquisition in coronary angiography. Stockholm 2013.

[14] http://https://en.wikipedia.org/wiki/Coronary_arteries -31.03.16.

[15]http://www.radiologysolutions.bayer.com/products/ct/injection/powerinjection/ -15.05.17.

[16] http://thoracickey.com/angiographic-data-2/ -15.03.18.

[17] Liu A, Tendick F, Cleary K, Kaufmann C. A Survey of Surgical Simulation: Applications, Technology, and Education. Presence: Teleoperators and Virtual Environments 2003,12,599–614.

[18] Luboz V, Blazewski R, Gould D, Bello F. Real-time guidewire simulation in complex vascular models. Vis Comput 2009,25,827–834.

[19] Mi S-H, Hou Z-G, Yang F, Xie X-L, Bian G-B. A multi-body mass-spring model for virtual reality training simulators based on a robotic guide wire

operating system. In: 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO): IEEE; 2013 - 2013. p. 2031–2036
[20] Lenoir J, Cotin S, Duriez C, Neumann P. Interactive physically-based simulation of catheter and guidewire. Computers & Graphics 2006,30,416–422.
[21] Luo M, Xie H, Le Xie, Cai P, Gu L. A robust and real-time vascular intervention simulation based on Kirchhoff elastic rod. Computerized medical imaging and graphics : the official journal of the Computerized Medical Imaging Society 2014,38,735–743.
[22] Tang W, Lagadec P, Gould D, Wan TR, Zhai J, How T. A realistic elastic rod model for real-time simulation of minimally invasive vascular interventions. Vis Comput 2010,26,1157–1165.
[23] A. Witkin, editor. Particle System Dynamics 1999.
[24] Ericson C. Real-time collision detection. Amsterdam, Boston: Elsevier 2005.
[25] Bartels RH, Barsky BA, Beatty JC. An introduction to splines for use in computer graphics and geometric modeling. San Mateo, CA: Morgan Kaufmann 2006, 1987.
[26] Wang P, Chen T, Zhu Y, Zhang W, Zhou SK, Comaniciu D. Robust guidewire tracking in fluoroscopy. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition: IEEE; 2009 - 2009. p. 691–698 .
[27] Baert SAM, Niessen WJ, Meijering EHW, Frangi AF, Viergever MA. Guide Eire Tracking During Endovascular Interventions. In: Goos G, Hartmanis J, van Leeuwen J, Delp SL, DiGoia AM, Jaramaz B, editors. Medical Image Computing and Computer-Assisted Intervention – MICCAI 2000. Berlin, Heidelberg: Springer Berlin Heidelberg; 2000,727–734.
[28] Chang P-L, Rolls A, Praetere HD, et al. Robust Catheter and Guidewire Tracking Using B-Spline Tube Model and Pixel-Wise Posteriors. IEEE Robot. Autom. Lett. 2016,1,303–308.
[29] Drew TM. anatomical heart; 2013. http://https://www.thingiverse.com/thing:45830 -12.06.18.
[30] John D'Errico. interparc; 2012. http://https://www.mathworks.com/matlabcentral/fileexchange/34874-interparc -12.06.18.
[31] Kim Larsen. Skeleton 3d model; 1/10/13. http://https://free3d.com/3d-model/skeleton-81335.html -5/15/18.
[32] Kern MJ. The interventional cardiac catheterization handbook. 3rd ed. Philadelphia, PA: Saunders/Elsevier 2012, 2013.
[33] http://www.syntaxscore.com -01.05.18.
[34] Dario Baldi. Aorta; 4/27/17. http://https://www.thingiverse.com/thing:2277464/ -4/25/18.
[35] http://https://github.com/relativty/wrmhl -08.05.18.