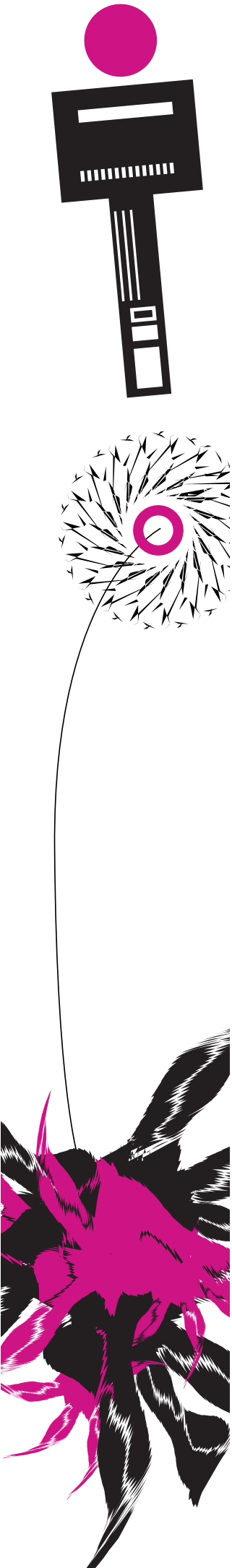


RAM ROBOTICS AND MECHATRONICS

ROBOT ASSISTED NEEDLE POSITIONING SYSTEM FOR LIVER BIOPSY

G. (Gaurang) Vashistha

MSC ASSIGNMENT

Committee:
dr. ir. J.F. Broenink
Y.X. Mak, MSc
dr. ir. M. Abayazid
dr. N. Strisciuglio

February, 2021

008RaM2021
Robotics and Mechatronics
EEMCS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

UNIVERSITY OF TWENTE. | TECHMED CENTRE UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

Summary

Intra-fraction motion is a major source of errors and inaccuracy in image-guided interventions. This motion can be induced by respiratory, skeletal muscular, cardiac, and gastrointestinal systems. Prior research at the University of Twente reports about the correlation between various surrogate signals and the liver tumor's respiratory motion. To prevent the issues linked to misplacement of the needle, due to inaccurate targeting of a liver tumor, estimation of the respiratory motion could be one of the promising solutions.

This research is done in collaboration with Philips to develop a robotic system for computed tomography (CT)-guided needle positioning for liver biopsy. The needle positioning system includes an interventional imaging system, a respiratory motion model, and a robotic manipulator. An already existing robotic phantom, that imitates the respiratory induced motion in the liver due to movement of the diaphragm was used. The phantom mimics the movement of the liver in cranio-caudal and anterior-posterior directions. The actual tumor location and the surrogate data (marker's positions) were detected using the Philips Medical system's Allura interventional imaging system. The Allura interventional imaging system uses CT scan images to locate the tumor. The system is also equipped with a camera system that detects the specially designed optical markers and provides their location in 3D space. The surrogate signal used was collected by placing external optical markers on the skin near the abdominal area. The machine publishes the generated CT-scan image and the surrogate data at a rate of 15 fps. ROS is used as the middleware to connect the system components. The motion of the tumor has been estimated using a learning algorithm that finds the correlation between the tumor location and surrogate signals. The response time of the system for processing the data and tracking the tumor position has been calculated to validate if the algorithm could be deployed in the real application. Also, the tumor's position estimation error has been calculated to determine the accuracy of the algorithm.

The system was trained and validated on pre-recorded CT-scan images and surrogate data of the liver's respiratory motion for four breathing cycles. The first three breathing cycles were used to train the system to find the correlation between the surrogate signal and the tumor's position. The fourth breathing cycle was used for validation. The results revealed that the system is able to track the live location of the tumor by processing the surrogate signal at a rate of 15 data points per second. The average processing time of surrogate data was around 0.7 ms. The estimation error for the learning algorithm for the cranio-caudal direction is 1.58 ± 1.54 mm and for the anterior-posterior direction is 0.3 ± 0.28 mm. As per the prediction results, the accuracy of the model is higher after full inhalation or full exhalation and is comparatively less during the transition from inhale to exhale or vice versa. The designed needle positioning system adapts to the patient's respiratory patterns and positions a robotic manipulator for performing diagnosis or treatments for the liver tumor in realtime. Further research is required to integrate the CT scan system and the robotic manipulator for demonstrating the clinical application.

Contents

1 Introduction

This report describes the research work done at the University of Twente in collaboration with Philips Medical Systems Nederland within the IMPACT project "Intelligence-based iMprovement of Personalized treatment And Clinical workflow supporT". Philips' line of business includes the manufacturing of medical, surgical, ophthalmic, and veterinary instruments and apparatus. They produce some of the powerful state of the art interventional X-ray systems that helps to precisely navigate to the target in real time, shown in Figure 1.1.

Figure 1.1: Philips Allura (interventional X-ray) system developed by Philips Healthcare [2014]

This research focuses on a feasibility study of a real time robot assisted needle positioning system for liver biopsy. The following sections of the chapter present the introduction to the current clinical procedures and challenges along with some common challenges in treatment and current solutions. This is followed by the objective of this research and the research questions.

1.1 Liver cancer and clinical procedures

Cancer is a major public health problem worldwide. A significant proportion of cancer can be cured, by surgery, radiotherapy, or chemotherapy, especially if they are detected early. With more than 3.7 million new cases and 1.9 million deaths each year, cancer represents the second major cause of death and morbidity in Europe. On a global scale, cancer accounted for 8.2 million deaths (around 13 % of the total) in year 2012 as reported by the World Health Organisation [2012].

1.1.1 Medical interventions

The treatment method of the liver tumor depends on the diagnosis. Some mild conditions can be treated by modifying the lifestyle such as exercising regularly, avoiding alcohol intake. Other serious problems may be treated with medication or with surgery. Some common diagnostic and treatment procedures are described below:

Diagnosing liver tumor

Blood test measures the level of a substance that is secreted by liver cancers called alpha fetoprotein (AFP). It is normally present in a fetus but fades away at birth. An increased level of AFP in adults indicates liver cancer as it is produced in 70% of the liver cancers as reported by Beneduce et al. [2003].

Imaging Medical imaging techniques are used to create visual representations of various internal parts of the human body. The breakthrough in medical imaging has eased the process of diagnosis and treatment-related image-guided interventions. Ultrasound could be used for detecting tumors as small as 1 cm (Beneduce et al. [2003]). High resolution CT scans and MRI scans are used to diagnose and stage these tumors.

Biopsy is a test used to diagnose liver conditions. Tissue samples are extracted from liver to check for signs of damage or disease, as shown in Figure 1.2. The most common type of biopsy is called *percutaneous liver biopsy*. In this a thin needle is inserted through abdomen into the liver and removing a small piece of tissue. The other types of biopsies are *Laparoscopic and Transvenous liver biopsy*. In *Laparoscopic biopsy*, laparoscope (thin tube) is put into skin through a tiny incision. The tube has a tiny camera attached to it. This camera helps to see inside of the belly, through another tube a needle is inserted to remove the sample. In *Transvenous biopsy*, an incision is made into the vein in neck. A tube is then inserted through the cut down to the liver. A contrast dye is used so that the vein show up more clearly on X-ray. A needle is then inserted through the tube to liver.

Treating liver tumor

Ablation is a treatment that destroys liver tumors without removing them. This technique is used when there are multiple small tumors present and surgery is not a good option. As per The American Cancer Society [2020], ablation is best used for treating tumors smaller than 3 cm. Since ablation destroys some of the healthy tissues near the tumor, it is not a good choice for treating tumors near major blood vessels, diaphragm, or bile ducts.

The work performed in this study is focused on liver biopsy procedure, specifically in conjunction with the use of motion estimation algorithms and robotic tools.

1.1.2 Existing procedure for liver biopsy

Image-guided needle interventions have become a dominant and important method for diagnosing liver conditions. The patient is positioned on the CT table, local anesthesia is applied to numb the path of the needle, and breath is held throughout the procedure. The target tumor is located through a CT check, the most secure needle insertion path is planned, and an interventional radiologist inserts the needle through the skin, pushes it forward towards the target, and samples the lesion. With the help of intraoperative CT scans, doctors affirm the places of the target and the needle. As mentioned by Waelkens et al. [2016], the surgical navigation system is similar to a GPS device to surgeons.

For an accurate biopsy, it is very important for the patient to stay still and repeat the breathing cycle exactly the same for the course of the biopsy. It is really challenging to perform biopsies for patients who have to struggle hard in holding their breath. If the patient does not comply with the process, due to the inaccurate needle placement the number of needle insertions and occurrence of complication increases. After a liver biopsy, there is a risk of internal bleeding and a leak of bile from the liver. Sometimes if the needle makes a hole in the chest wall, it collapses the lung and allows air to enter (webMD [2020]; Jacob [2020]).

To reduce the errors in locating the target tumor location, various attempts were made earlier to model the relationship between respiratory motion and the motion of the tumor. Most of the previous work on respiratory motion emphasizes that the respiratory pattern differs between

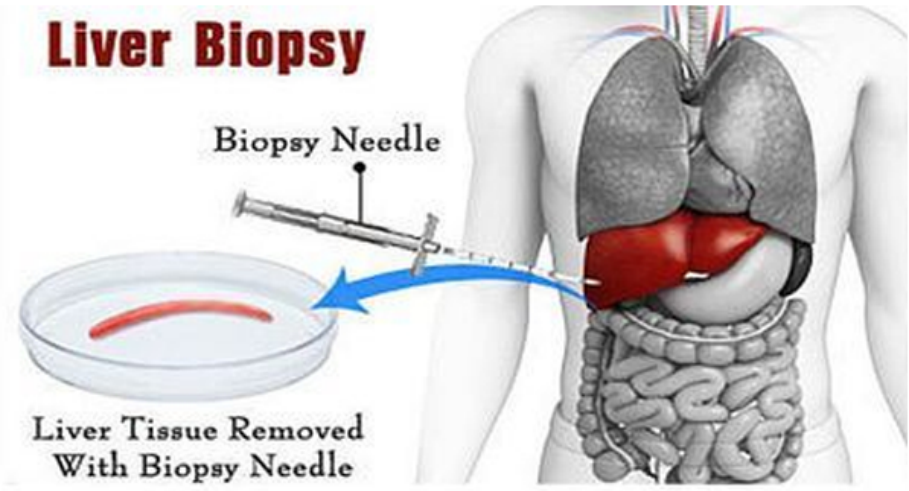


Figure 1.2: Liver Biopsy: collection of liver tissue for diagnosis (Barry [2015])

breathing cycle to breathing cycle (inter-cycle variations), from patient to patient (inter-patient variation), and between different sessions of diagnostic or treatments. As mentioned by Keall et al. [2006], the cycle to cycle variation also depends on two physiological causes of respiration that is, contraction of the thoracic diaphragm muscle and movement of the rib cage.

As per the previous studies at the University of Twente, the respiratory motion estimation and needle positioning system mostly depend on a few main components i.e., surrogate signals, motion model, fitting method, target motion prediction, a robotic manipulator for needle positioning as reported by McClelland et al. [2013]. These parts are discussed in detail in the next chapter.

1.2 Challenges and current solutions

Tumor localization in the thorax and abdomen remains a challenging task for treatments, mostly due to the respiration-induced liver tumor motion and limited tumor contrast against normal liver tissues. To take advantage of medical devices for surgery, it is crucial to have accurate information about the tumor during surgery. High-quality and realtime medical images would be required for determining the accurate position of the tumor. In the literature, several solutions have been proposed to handle respiratory motion-induced problems and minimise the inaccuracies. Keall et al. [2006] separated this into five major categories:

- **Motion-encompassing methods**, these methods comprise of imaging and treatment planning for the entire range of the tumor motion, i.e imaging complete breathing cycles.
- **Respiratory gating techniques**, involves administration of radiation for a specific portion of the patient's breathing cycle. Respiratory motion is divided into two parts, that is displacement and phase. Accordingly, the method of gating is either displacement gating or phase gating.

- **Breath holding techniques**, several breath-holding techniques like 'deep-inspiration breath-hold', 'active breathing control', 'self-held breath-hold' etc. could be used based on the patient's capability of holding breath and availability of necessary equipment.
- **Forced shallow breathing methods**, employs a stereotactic body frame with a plate attached to that is pressed against the abdomen. This helps in immobilising and positioning the patient correctly, the applied pressure permits limited respiration.
- **Realtime tumor tracking**, could be done using methods like: direct tumor imaging, tracking implanted fiducial markers, predicting tumor location based on surrogate signals.

The active and simplest approach is holding the breath for some time consistently at the same respiration phase for the duration of the needle insertion. In image-guided interventions like biopsies, if the target lesion is moving it may result in re-insertion of needle multiple times, the occurrence of motion artifacts in CT images. Multiple re-insertions decreases the patient's comfort and safety and might result in complications. According to Reisner et al. [2007], remaining still for a long time and holding breath will make the patient nervous and will degrade the compliance level. Insufficient patient attention can increase needle placement inaccuracy, the number of needle passes, and complications such as pneumothorax and bleeding. Various other passive approaches for motion tracking were discussed in the literature by Shirato et al. [2000], for example implanting markers in the region of interest. Therefore, patient training is required to allow the patient to familiarise with the breathing techniques and for evaluating their own ability to reproduce the respiratory cycles.

A literature review by Makary and Daniel using weighted analysis of events reports that nearly 210,000 to 400,000 deaths among hospitalised patients are associated with medical errors (Makary and Daniel [2016]). It is hard to eliminate human errors completely, but it can be reduced by having safer systems. For example, percutaneous interventions involve the use of long and rigid tools inserted into the patient via small incisions that can cause ergonomic challenges and limits the manual dexterity of the surgeon. In addition to this, a separate display is used to get visual feedback from the camera system which affects the hand-eye coordination of the surgeon.

The fundamental perceptual contrasts among people and robots lie in the capacity of preparing subjective and quantitative data. Robots can coordinate a lot of quantitative information decisively through various sensors, in this way having the option to perform monotonous tasks with great stability and positional precision. Whereas, surgeons are better than robots when it comes to qualitative decision making. Robot-assisted surgeries are part of a broader category of computer-aided surgery (CAS). The robot is just a single component of the multi-functional system designed to assist the surgery and to reduce errors.

The previous studies at the University of Twente were focused on modelling the respiratory induced motion on the liver and selecting a suitable surrogate signal for tracking the tumor (Fahmi et al. [2018]; Hoitzing et al. [2019]). In the previous work, Hoitzing et al. [2019] compares various learning algorithms and reports the suitable algorithm for modelling the respiratory induced motion in the liver. Hoitzing et al. [2019] discussed an approach to store the CT-scan images and then perform training and prediction to estimate the location of target tumor. The major limitation of Hoitzing et al. [2019] was that the algorithm was not capable of running in realtime and therefore cannot be directly used with a robotic manipulator.

1.3 Research question

Based on these limitations, the focus of this study is to bring the literature one step closer to clinical practice. The research question is formulated as follows:

- **How to develop a needle positioning system for compensating the respiratory induced liver motion in real time?**

Furthermore, to answer the research question, the main question is sub-divided into the following two sub-questions for validation:

1. How fast the designed system can process and respond to the incoming surrogate signals and position the robotic arm for needle insertion? Is it possible to compensate for respiratory induced motion in the liver and position the robot in realtime?

2. How accurate is the motion estimation for liver tumor with the selected surrogate signal?

1.4 Approach

This research focuses on developing a robot-assisted needle positioning system that can estimate the target tumor's location and position the robot in realtime for needle insertion.

The location of the tumor present in the liver changes with time due to the respiratory motion. Therefore, it is important that the robotic needle positioning system responds well in time during the motion estimation phase. Furthermore, as it is not required to position the robot during the learning phase, it is acceptable if the system does not perform in realtime.

In this study, a ROS-based software has been implemented that connects the various components of the system. The developed system uses image processing tools to detect the target tumor in the CT-scan images. The system also includes a method to filter the surrogate signal for false-positive data points. To find the correlation between tumor motion and surrogate data, the system initially stores the detected tumor position and the corresponding surrogate signal. Once enough data is collected, the tumor position and the surrogate signal are synchronized based on the timestamps. Next, a supervised-learning algorithm has been implemented that uses this synchronized data to find the correlation between tumor motion and surrogate data. Due to COVID-19 regulations, it was not possible to integrate and test the complete hardware setup. Therefore, software nodes are used to replace the imaging system that publishes the pre-recorded CT-scan images and the surrogate signal at the same rate as the actual CT-scan machine. The data processing node receives the CT-scan images and surrogate data and trains a motion model that is later used to estimate the tumor location. This estimated tumor location is then sent to the robotic arm.

The validation is divided into two major parts to support the research question. The first validation experiment reports about the feasibility of the system in terms of response time. The second validation experiment evaluates the accuracy and the performance of the learning-based model by comparing the estimated tumor motion with the measured tumor motion from the CT-scan images.

1.5 Thesis outline

The report is structured as follows: Chapter 2 describes the literature study performed on the respiratory motion model, the robotic liver motion phantom, robotic manipulator and Robot Operating System (ROS) . Chapter 3 mentions the system requirements and explains the various modules of the system. Followed by Chapter 4 that explains the experimental setup and the experiments performed. Chapter 5 presents the results of performed experiments. In chapter 6, a conclusion is given along with recommendations for future work.

2 Background

This chapter presents a theoretical background regarding medical interventions and advancements for diagnosing and treating liver tumors. The following section gives a brief description of anatomical directional references followed by a respiratory motion and its effects on the liver. Next, the available respiratory motion phantom is presented, followed by the details of surrogate signals, the motion estimation algorithm, details of the robotic setup used, and the ROS based interface.

2.1 Anatomical directional references

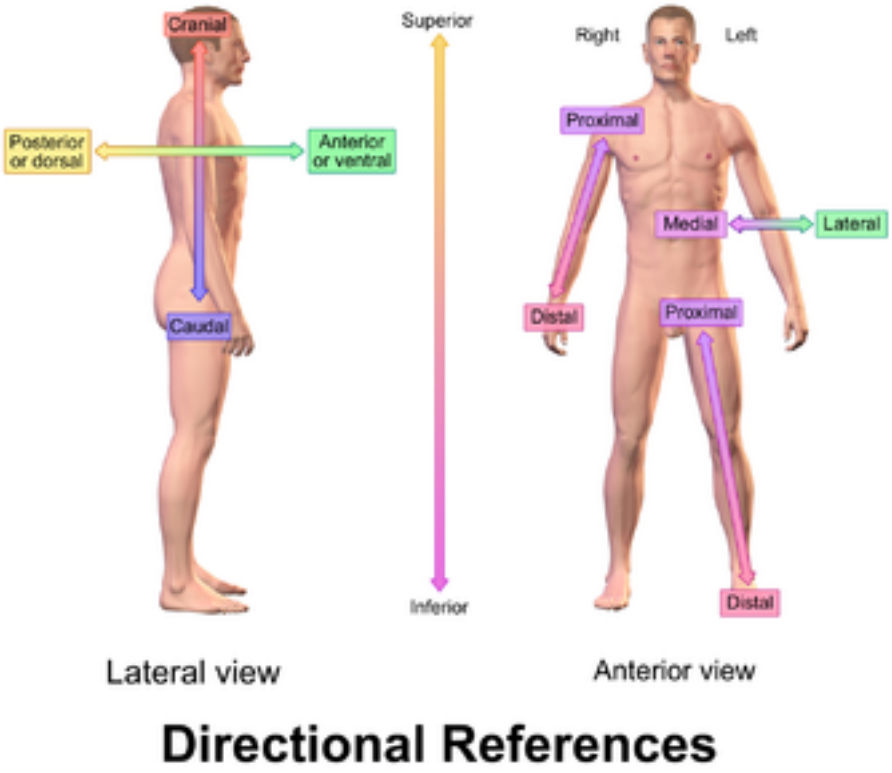


Figure 2.1: Anatomical directional references. In this research, the motion of the liver is considered only in two directions i.e. cranial-caudal direction and anterior-posterior direction as shown in the figure on the left. (Blausen.com staff [2014])

Figure 2.1 gives an overview of the directional references with respect to a human body. These anatomical positions are used to describe the body parts and position of patients regardless they are lying down on their face or their side or facing down. These directional terms describe body parts with respect to another. Anterior-posterior or AP direction where anterior indicates "front", posterior indicates "behind". Cranio-caudal or CC direction where cranial is upwards towards the head, and caudal in towards the feet. Medial-lateral or ML direction where medial is toward the midline of the body (example, the middle toe is located at the medial side of the foot). Lateral - away from the midline of the body (for example, the little toe is located at the lateral side of the foot).

2.2 Breathing mechanism and its effects

Breathing is an involuntary action in humans. The diaphragm, as the main respiratory muscle, and the intercostal muscles of the chest wall play an essential role by generating, under the control of the central nervous system, the pumping action on the lung. The muscles expand and contract the internal space of the thorax, the bony framework of which is formed by the ribs and the thoracic vertebrae. As shown in Figure 2.2, the liver is present directly below the diaphragm and hence it observes an induced respiratory motion.

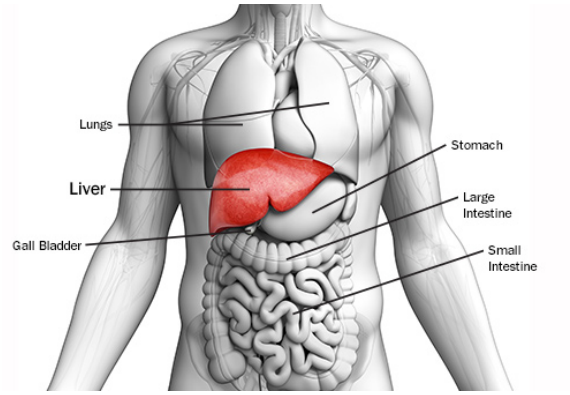


Figure 2.2: Position of the liver with respect to respiratory organs. Due to the motion of the diaphragm during respiration, an induced motion is observed in the liver. (Hoffman [2014])

The respiratory motion for each patient differs and it is also dependant on various parameters. For the research purpose, a respiratory robotic phantom is a good solution to mimic the patient breathing cycles. The respiratory motion could be easily adjusted and controlled by controlling the parameter of the phantom. It is also possible to repeat exact same breathing patterns multiple times for experiments.

2.3 Respiratory motion phantom

Advancements in robotic interventional technologies could help us in the precise treatment of tumors, the difficulty arises while validating these in practice. A robotic motion phantom was designed and developed at the University of Twente that can mimic the motion of the liver induced due to respiration. As the liver is present directly under the diaphragm, a considerable amount of movement is observed in the liver as well. This movement complicates the percutaneous treatment. Therefore, the motion phantom could be used as a substitute to validate the needle positioning system for such interventions.

This chapter describes the hepatic respiratory motion based on the literature and how the respiratory motion phantom works. The properties and patterns of the hepatic respiratory motion are presented according to the literature. The motion phantom used for this project was designed by Naghibi et al. [2018].

The motion phantom uses soft actuators to mimic the human-like behaviour of air operated diaphragm. As shown in Figure 2.3, the mechanical system is a second-order mass spring - damper system that is made of all nonmetallic parts. The amplitude of the respiratory induced liver motion for a normal breathing cycle is around 10 to 40 mm in cranio - caudal (CC) direction and 1-12 mm in anterior-posterior (AP) direction (Clifford et al. [2002]; Keall et al. [2006]).

As per the literature Miao et al. [2017], the average human respiration rate for normal breathing is 12 to 20 cycles per minute and 7 to 8 cycles per minute for deep breathing. That means the respiratory cycle for normal breathing repeats every 3-5 seconds and every 7-8 seconds for deep breathing.

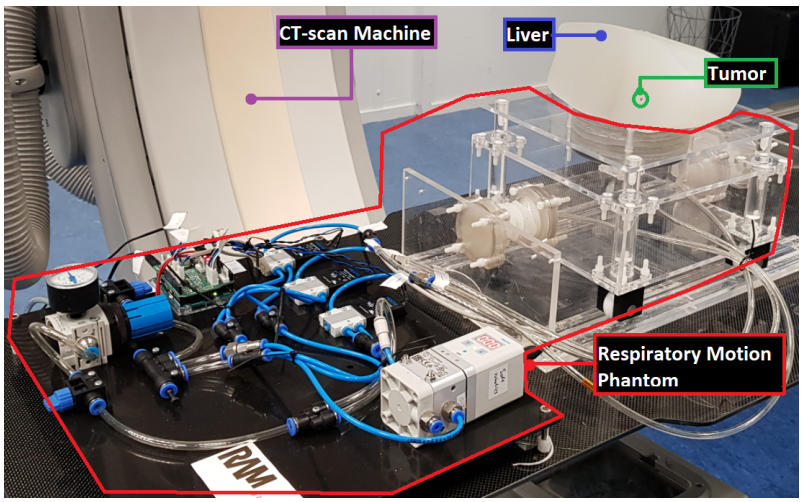


Figure 2.3: The respiratory motion phantom used for replicating the motion-induced on the liver. The robotic phantom (shown in red boundary) was developed by Naghibi et al. [2018] at the University of Twente. The white color liver, shown with the blue label is made of a flexible material called Ecoflex 00-30 (Smooth-on Inc., Pennsylvania, USA). A metallic bullet used as a target tumor is shown with the green label. The complete respiratory phantom is placed on the patient table of the CT-scan machine (shown with the purple label)

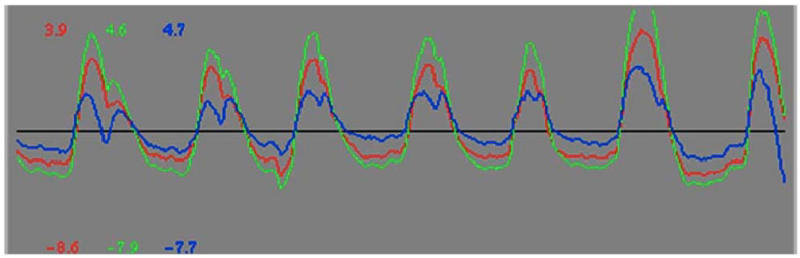


Figure 2.4: Breathing pattern of a patient as reported by Keall et al. [2006]. The three curves in each plot correspond to infra-red reflector measured patient surface motion in the superior-inferior, anterior-posterior, and medial-lateral directions, with each component arbitrarily normalized.

Patients' breathing patterns can vary in magnitude, period, and regularity during imaging and treatment sessions. Figure 2.4, show the respiratory pattern from a patient. The three curves in the plot show the measured surface motion in superior-inferior, anterior-posterior, and medial-lateral directions as mentioned by Keall et al. [2006].

The motion phantom works on the following assumptions:

- The robotic phantom was designed to mimic the two most dominant motion directions of liver motion i.e. the CC and AP directions. The motion in ML direction is negligible compared to the other two.
- The phantom replicates the normal respiration cycle.
- The movement of phantom in CC direction is 3 cm and in AP direction is 1 cm.
- The breathing frequency is 20 cycles per minute.
- the inhalation duration is 1 sec and for exhalation it is 2 seconds.

Figure 2.5, shows the setup for the motion phantom as designed by Berijanian et al. [2018]. The motion was controlled by a micro-controller - Arduino UNO (Arduino, Turin, Italy) with the

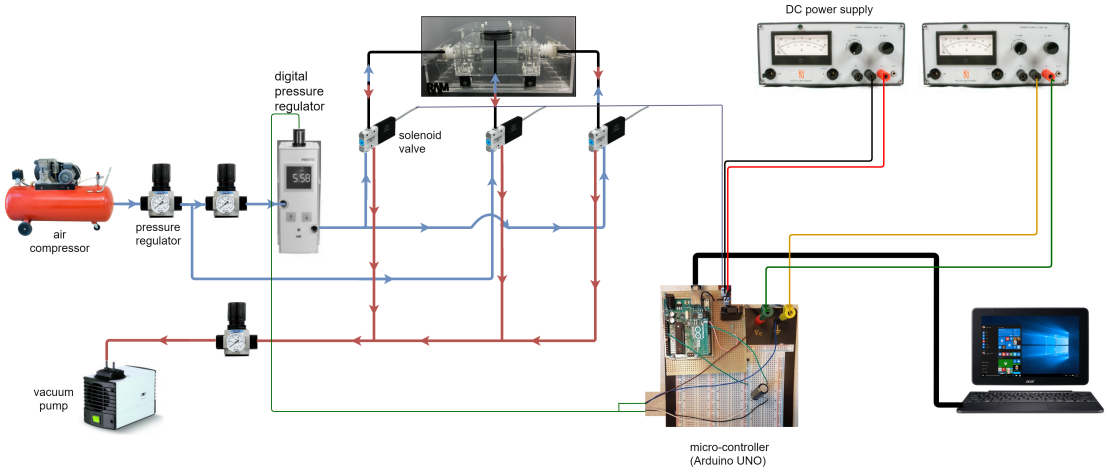


Figure 2.5: Electrical and mechanical setup for the respiratory motion phantom designed by Berijanian et al. [2018]

help of digital pressure regulators that connects the pneumatic actuators of the phantom to the air source.

2.4 Locating tumor

2.4.1 Surrogate signals

Surrogate signals help to track the position of the target lesion without directly tracking the lesion. The relationship between surrogate data and the motion model can be divided into two categories, direct and indirect correspondence. In direct correspondence the motion model is a direct function of the surrogate signals as shown in the equation:

$$M = \phi(s)$$

The above equation represents the motion model (M) as a function of surrogate data (s). Where as in the indirect correspondence model, the internal variables are used to define the motion model for example position in the respiratory cycle (McClelland et al. [2013]).

$$M = \phi(\hat{x})$$

where,

$$\hat{x} = \arg\max_{x} \mathrm{Sim}\left(F\left(T\left(I, \phi(x)\right)\right), s\right)$$

Here, x is vector of internal variables, $\phi(x)$ is vector of motion parameters, I is reference image, T is transformation function. The function F simulates the surrogate data from the transformed reference image. Sim is a measure of similarity between simulated surrogate data and the measured surrogate data. The basic requirement of surrogate signals is that they should have a high correlation with the target, they should be measurable easily, and most importantly they should have a high update frequency.

There are various types of surrogate signals used in literature for tracking respiratory motion for example inertial measurement unit (IMU), external optical markers, radio-opaque markers, fluoroscopic images, 4D CT, spirometry, abdominal surface motion, infrared emitters (Berijanian et al. [2018]). The most common one is the surrogate obtained from spirometry that measures the respiration air volume. the respiration air volume is highly related to the motion induced due to breathing but, due to air leakage, there is always inaccuracy in measurements (Fahmi et al. [2018]).

Another commonly used surrogate signal is measuring the displacement of the external markers placed on the abdominal area (Berijanian et al. [2018]). The displacement of external markers could be tracked by an imaging modality of choice, 2D or 3D. In a few of the previous studies, optical markers were placed on the abdomen skin and the motion was recorded by digital camera. Simultaneously the motion of the target tumor was extracted by MR images or CT images and the tumor motion was correlated with the target motion. The results concluded a high correlation between the surrogate data from the marker and the actual motion of interest (Fahmi et al. [2018]; Hoitzing et al. [2019]). As stated by Wasza et al. [2016], using external markers is advantageous due to its high temporal and spatial resolution. In this study, the external optical marker's position was used as the surrogate data shown in Figure 2.8. Various imaging modalities for tracking the target motion are discussed below.

2.4.2 Tumour imaging

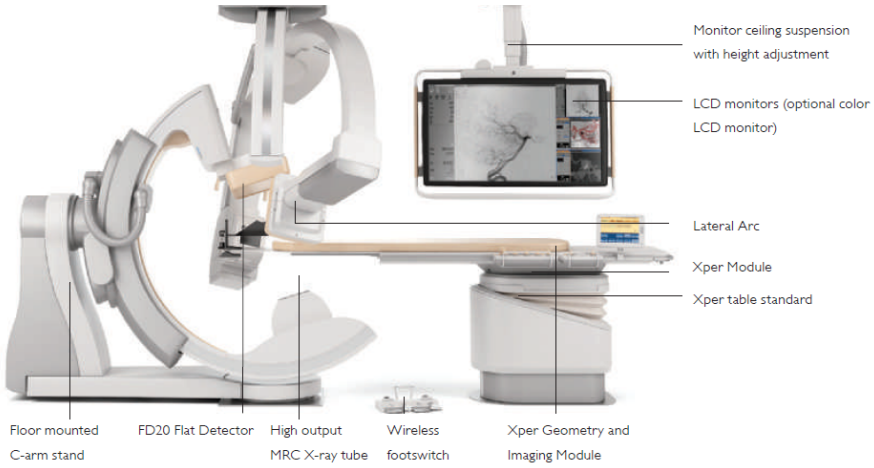
There are total six imaging methods available for clinical diagnosis and treatment of tumor inside human body:

- *Ultrasound*, It uses high frequency sound waves to produce images.
- *Magnetic Resonance Imaging (MRI)*, it uses combination of large magnets or radio frequencies to make images.
- *Computed Tomography Scan (CT scan)*, It uses CT scan images to produce cross-sectional images.
- *Single photon emission computed tomography (SPECT)*, it is a nuclear imaging test. It uses a radioactive substance and a spatial camera to create 3D pictures
- *Positron emission tomography (PET)*, a tracer injected in the body, it collect in the area of high chemical activity (area of diseases in the body.

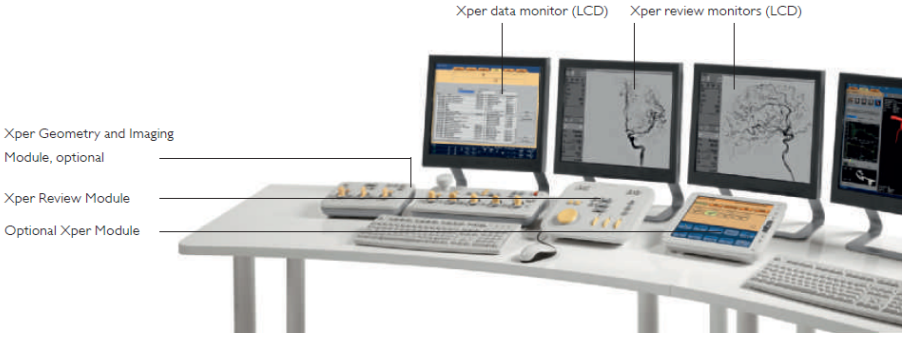
For this project, CT scan images were used to locate the tumor. The high contrast resolution of CT provides images with better details of hard and soft body structures. Philips Allura C-arm system has been used to acquire the tumor's location. The captured images could be collectively used to measure the motion of the liver and therefore the motion of the target tumor. The C-arm system captures images with a field of view of 48 cm and forms images of 1024×1024 pixels. The system is capable of capturing and publishing images at a rate of 15 fps. Philips Allura system is also equipped with two sets of four cameras as shown in Figure 2.7, one set of the camera is embedded in the C-arm, another set is embedded in the external movable OR lamp.

The 3D camera system of Allura is used to locate the optical markers shown in Figure 2.8. The medical-grade optical markers are specially designed by Philips for intraoperative 3D imaging for navigation and validation. The sterile, flat, adhesive circular markers could be randomly placed on the skin around the surgical site. A minimum of 5 markers is required to be in the field of view of the cameras for motion tracking.

The Philips' Allura system collects all the data coming in from the x-ray detector and the 3D optical camera system and process it. The stereo images coming from the set of 4 cameras are processed to detect the markers. The position of that point is then calculated in 3D space by the OR PC. These optical markers placed on the skin as the motion of the skin above the liver is highly correlated to the motion of the liver. The position of these markers in 3D space could be used as a surrogate signal to obtain the position of the liver tumor.



(a) C-arm setup with the patient table and LCD monitor



(b) PC and controls for CT scan machine

Figure 2.6: The two subfigures above show one of the example configurations of Allura X-ray system developed by Philips Healthcare [2014]

2.4.3 Target position detection using image processing

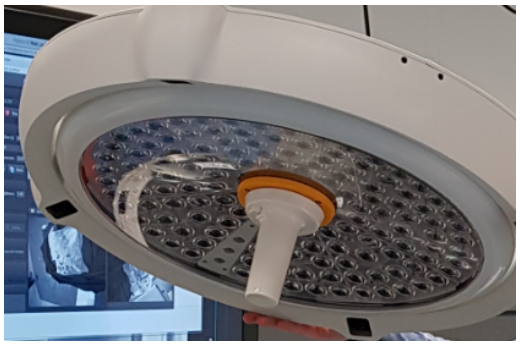
The target tumor could be detected using the imaging modalities available for diagnosis and guidance during treatment as mentioned in the previous section. For this research, CT scan images were used to extract the location of the tumor. The CT scan images contain data with object like characteristics such as high gradient edges, noise, artefacts generated due to the motion of the target (Bell and Shetty [2020]). Therefore, based on visual inspection of the image and using the available processing tools, the target tumor is correctly located for training the regression model. This section discusses the methods that could be used to detect tumors using image processing.

Hough circles

Hough circle transform is a feature extraction technique used to extract circles from an image. With the help of parameters, the target circular tumor could be detected even if there are multiple tumors present in the image.

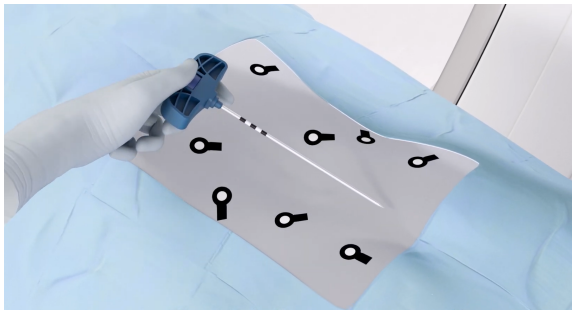
- *Distance between circles*, the minimum distance between the centers of circles could be set using this parameter.
- *Gradient value*, a gradient value for edge detection.
- *Accumulator threshold*, it is a threshold for the accumulator matrix in selecting the circles passing through the points.

(a) C-arm x-ray source with camera system

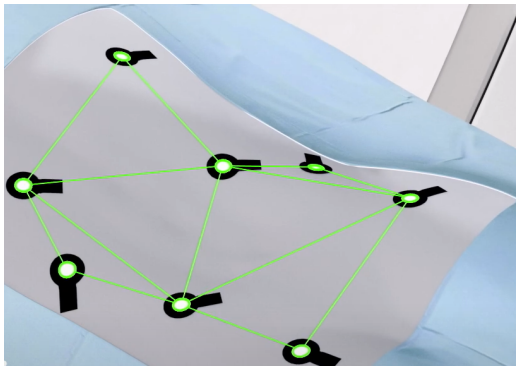


(b) Movable OR lamp with embedded camera system

Figure 2.7: Philips Allura optical camera setup. There are four cameras present on the c-arm x-ray source and four cameras on the external movable OR lamp. For this project, the cameras mounted on the external OR lamp are used. This camera setup is responsible to give the accurate 3-D location of the optical markers placed on the human body.



(a) Optical markers and needle designed specially for tracking by the camera system



(b) Mesh model generated by interconnecting the markers for motion tracking and compensation

Figure 2.8: Optical markers designed and developed by Philips [2020]

- *Radius*, minimum and maximum radius for circles could be defined for better detection of the target.

Due to the noise present in the x-ray image, Hough circle detection cannot be directly performed on the images. To avoid the detection of false circles, image enhancement and restoration must be done at the cost of increased processing time.

Blob detection

This technique is used to extract blobs from an image (Bradski and Kaehler [2008]). This method follows a series of steps starting from converting the grayscale image to a binary image. The image is converted into binary by applying parameters on the minimum and maximum threshold values. Contours are then extracted from the binary image by searching for the connected components. If there are multiple contours in a region, i.e. the distance between contours is less than the minimum distance parameter they are grouped as a single blob. From these groups of contours, the final blob center and radius are calculated. Various parameters that help the infiltration of blobs are as follows:

- *Color of blob*, this parameter is not useful in this case as we are filtering grayscale CT scan images.

- *Size of blob*, this parameter is used to locate the target tumor based on its size and ignore the remaining blobs in the image. The maximum and the minimum area could be defined in the parameters.

- *Shape*, the blob detector has three different filters for the shape of the target. They are circularity, convexity, inertia ratio. The inertia ratio of 0 will yield elongated blobs (closer to lines) and an inertia ratio of 1 will yield blobs where the area is more concentrated toward the center (closer to circles).

2.5 Robotic manipulator for medical intervention

With a clear potential for streamlined processes, repeatable safety measures, and more efficiency the collaborative robots - or cobots – are finding their place in the Industrial 4.0 automated revolution now propelling the healthcare industry. The condition of the patient is still diagnosed by a trained physician or a doctor, the robotic arms are just to assist them in surgeries. One of the most famous cobots in the surgery world is the da Vinci system (Intuitive Surgical Inc., California, USA), built with robot arms and high-tech cameras (highly-magnified 3DHD vision and true depth perception) to assist surgeons during operations.

For this research, Franka Emika's Panda robotic arm (Franka Emika GmbH, Munich, Germany) was used. The Panda robotic arm is designed to be versatile such that it can be easily used in various different environments like hospitals, production lines, laboratories, logistic platforms, etc. The Panda robot is suitable for research as it supports various C++ and ROS integration, and is easy to use. The robot has 7 degrees of Freedom and can carry a payload of up to 3 kgs. The Cartesian velocity limit of the end effector is up to 2 m/s. The robot has position repeatability of ± 0.1 mm. The control unit of the robot arm is connected to the PC through an Ethernet cable. To control the robot it is required that the sum of the following measurements is less than 1 ms (Franka Emika [2020a]) :

- Round trip time (RTT) between the workstation PC and FCI.

- Execution time of your motion generator or control loop.

- Time needed by the robot to process your data and step the internal controller.

The libfranka is the user side package that handles communication with the Franka Control Interface (FCI). Libfranka and FCI communicate via Ethernet cable enabling the controller to communicate with user commands:

- execute non-realtime commands to control the Hand and configure Arm parameters

- execute realtime commands to run your own 1 kHz control loops

- read the robot state to get sensor data at 1 kHz

- access the model library to compute your desired kinematic and dynamic parameters

Packet loss policy
The packet loss policy for the motion generator is constant acceleration. that is, the robot takes the previous waypoints and performs a linear extrapolation for the missed time steps. Similarly, if the controller command packet is dropped, a constant torque loss policy is used where the last received torque is applied. A message is dropped by the Franka Controller Interface if the *<1ms* constraint is violated for a cycle. If more than 20 consecutive packets are dropped, the robot will invoke fail-safe mode and will stop (Franka Emika [2020a]).

Packet losses generally happen due to poor performance of the PC or network card which as a result reduces the quality of the connection. Due to the poor performance of the PC, it is

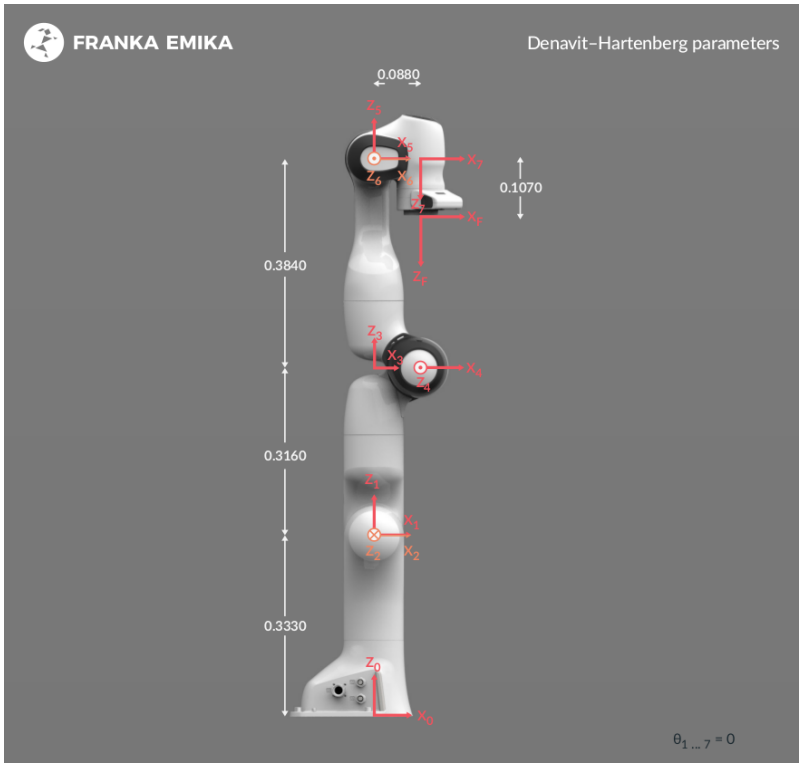


Figure 2.9: Panda's kinematic chain as designed by Franka Emika [2020b]

possible that the control loop takes too much time to do the computations. In this case, control assumes a constant acceleration model or a constant torque to extrapolate your signals.

2.5.1 Safety aspects of using robot

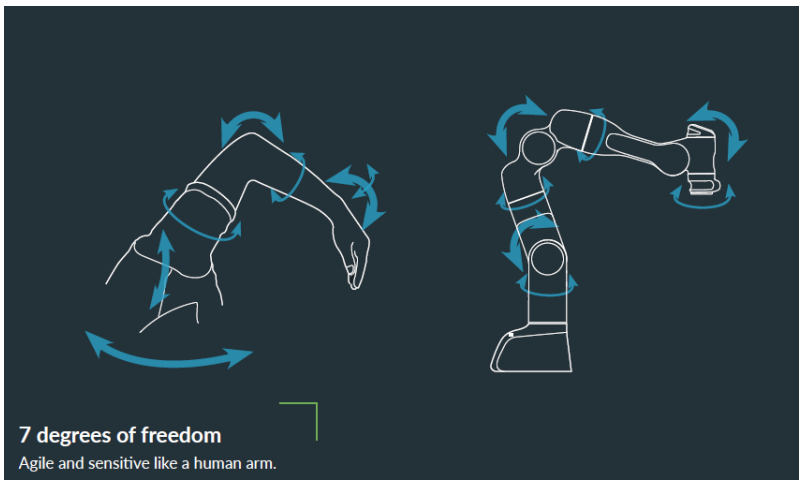


Figure 2.10: Panda robot: human-like kinematics Franka Emika [2020b]

The Panda robotic arm is a cobot (collaborative robot system), it is designed for direct human-robot interaction within a shared space where humans and robots are present in close proximity. The International Federation of Robotics (IFR), categorises robots in two major categories - industrial robots for automation, service robots for domestic and professional use. According to IFR, the cobots are designed with a variety of technical features that ensures the safety of humans that are in proximity or are in direct contact with the robots. These features include light-weight materials, rounded contours, padding, skins (padding with embedded sensors). The

embedded sensors measures and control the force and speed and ensure these do not exceed defined thresholds if contact occurs as per the International Federation of Robotics Frankfurt IFR [December 2018]. However, a safe robot does not guarantee a safe collaborative application in practice. A risk assessment is required by the end-user for example an assessment for the robot condition, the end-effector, workspace, and surrounding cables and tools.

Panda robotic arm for Human-Robot Interaction

Franka's Panda robot features the following three properties that are incorporated in its design and are desirable for a safe physical Human-Robot Interaction (Franka Emika [2020b]):

- Lightweight design, 18 kg
- Flexible actuator design
- Torque sensing, torque sensors in all 7 axes

2.5.2 Visualisation environment

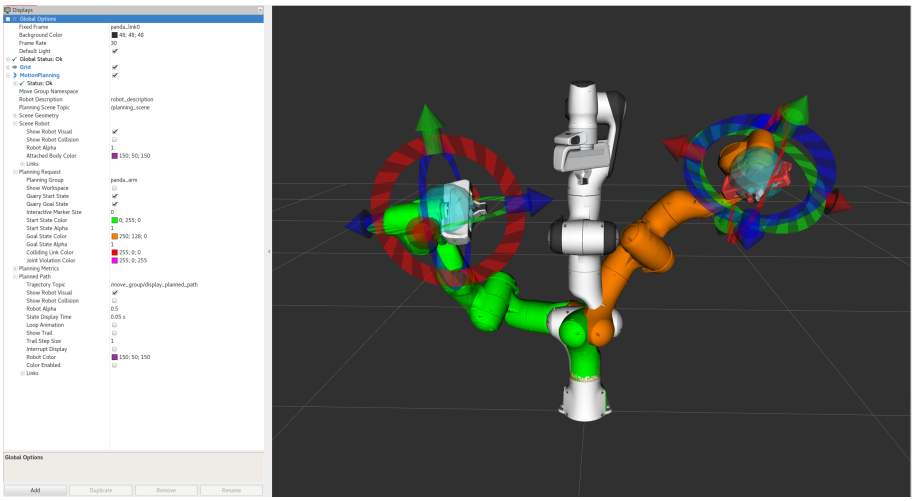


Figure 2.11: Robot visualisation example in RViz

To simulate the robot positioning, the visualisation tool from ROS, RViz can be used. "ROS visualisation" or "RViz" is a powerful tool developed by ROS for the 3D visualisation of robots and algorithms. With the help of the RViz simulation environment, we can visualise the robot's perception of the world (simulated). To let the robot know about its surrounding, sensors could be integrated into the system.

For representing the robot in the simulation world, Franka Emika provides the "franka_description" package with the description of the robot in terms of the kinematics, joint limit, visual surfaces, and the collision space (Franka Emika [2020a]). These descriptions are in the Unified Robot Description Format (URDF) format.

Figure 2.11 shows the Panda robot in RViz with different overlapping visualisations. The orange coloured arm shows the goal state of the motion planning, the green coloured arm shows the start state of the motion planning. The 6-DOF interactive markers, shown with red, blue, and green rings with arrows pointed outwards, are used to interact with the robot by changing their position or rotation. The rings and the arrows could be used to rotate and translate the end effector respectively.

2.6 Respiratory motion estimation

As mentioned in the previous chapter, there are various methods proposed earlier for overcoming the problem of respiratory-induced motion in the liver. Breath-holding being the simplest and most commonly practiced but this limits the intervention time to less than 30 seconds. Respiratory gating allows acquiring images for a limited time window, for example (end of expiration), this significantly increases intervention time (McClelland et al. [2013]). Another approach is to track the target motion for some time and then estimate the target position for a few respiratory cycles.

In previous studies, it was found that with the help of proper surrogate signal and corresponding target location could be used to develop models that can estimate and correct for the effects of respiratory motion. As defined by McClelland et al. [2013] a motion model refers to a process that takes some surrogate data as input and produces a motion estimate as output. Motion models are useful when it is not practically possible or feasible to directly track target motion during the procedure. In the scope of this research, the motion model plays an important role as acquiring and processing live tumor images is time-consuming. One of the main requirements of a needle positioning system is that it should be able to process the data in real-time and send target coordinates to the robot.

The motion model is based on the motion measurements made from imaging data during the training phase. The imaging data and the surrogate data are acquired at the same time. The model then estimates the motion of interest based on surrogate signals. Therefore once the model is ready, only the surrogate signal needs to be acquired.

The respiratory motion differs from one breathing cycle to another, also it varies from person to person. There have been a number of motion models present in the literature for modelling respiratory induced motion Fahmi et al. [2018]; Abayazid et al. [2019]; Hoitzing et al. [2019].

According to McClelland et al. [2013], there are four major components of the motion model:

- Choice of surrogate data
- Choice of motion representation
- Correspondence model
- Fitting method

As presented by McClelland et al. [2013] and Hoitzing et al. [2019], the direct correspondence model can be represented as:

$$\phi(s) = A_s + A_0$$

Where, ϕ is the direct correspondence model, s is the surrogate signal A_s is the matrix of input surrogate signals and A_0 is the vector of constants. A single surrogate signal limits the motion to be only in one direction. To model a complex motion, multiple surrogate signals could be used. The linear model could be expanded for a polynomial function of the surrogate signals to estimate the motion for multiple surrogate signals as follows:

$$\phi(s_1, s_2) = \sum_{i=0}^{n} \sum_{j=0}^{j-1} A_{i,j} s_1^i s_2^j$$

Here, s_1, s_2 are the surrogate signals, $A_{i,j}$ is a vector of polynomial coefficients. Other direct correspondence models that seen limited use in the literature are based on Fourier series, neural networks, fuzzy logic, least squares support vector machines, and support vector regression (McClelland et al. [2013]).

2.6.1 Fitting methods

Fitting is used for finding the best fit line of our predicted data that is closest to the actual values. There are various fitting methods reviewed in literature for describing the relation between the target motion and the surrogate data. The most common fitting method used for respiratory motion is the linear least squares (McClelland et al. [2013]). The following equation gives the cost function which is the norm of the estimation error vector:

$$J(\theta) = \frac{1}{m} \parallel y - \hat{y}(\theta) \parallel^2 \tag{2.1}$$

Here, m is the number of measured samples for vector y and the unknown parameter is found for the minimum value of the cost function. Various other methods are extensions of the linear least squares. For example, separate methods could be used for modelling inhalation and exhalation for acknowledging the intra-cycle variations in respiratory cycles. As presented by Klinder et al. [2009], the principal component analysis and the ridge regressions are the modification of linear least-squares that helps in a more robust fit to the data.

Berijanian et al. [2018] in her research compared various regression models, where different algorithms were employed to fit the correlation between surrogate signals and motion of interest. A comparison was made between ordinary multivariate linear regression, ridge regression, lasso regression, and 2^{nd} order polynomial regression. For an ordinary multivariate the output signal could be approximated by:

$$\hat{y}(\theta) = \sum_{i=0}^{n} \theta_i x_i \tag{2.2}$$

where $\hat{y}$ is a vector of approximations of the output. n is the number of input signals. θ_0 is the bias and x_0 is the constant term.

In ridge regression For generalising the above model, that is the estimation accuracy for unseen data could be added by introducing a regularisation parameter. This parameter helps in better estimation for the surrogate data that is highly correlated and is prone to over-fitting. The cost function for ridge regression is as follows:

$$J(\theta) = \frac{1}{m} \parallel y - \hat{y}(\theta) \parallel^2 + \alpha \sum_{i=1}^{n} \theta_i^2 \tag{2.3}$$

here α is the regularisation parameter. For a large α value, the penalty and the training error decreases. Whereas for a small value of α, the model acts as an ordinary linear regression (The MathWorks INC [2018]).

The lasso modification of linear regression is similar to the ridge, where the model is penalised for the sum of absolute values of weights. The following equation gives the cost function of lasso regression:

$$J(\theta) = \frac{1}{m} \parallel y - \hat{y}(\theta) \parallel^2 + \alpha \sum_{i=1}^{n} \theta_i \tag{2.4}$$

The only difference in ridge and lasso regression is in the regularisation. These regularisation techniques help in avoiding offer-fitting of the model. Ridge regression uses L_2-regularisation that adds squared magnitude of coefficient as penalty term to the loss function. Whereas lasso regression applies L_1-regularisation to the model which helps in reducing the big weights to small weights or reducing the smaller weights to zero. In other words, lasso regression helps in eliminating the less important features, this works well for feature selection in case there is a huge number of features.

2.7 ROS based system interface

One of the main components of the robotic needle positioning system is the interface that connects the robotic manipulator to the imaging modality and the surrogate sensors. In terms of deployment of robotics, having robot and imaging devices or sensors for surrogate data from different vendors is a challenge. For the real-time motion estimation and needle placement requirement of the system, the surrogate signals should have a fast communication interface between systems.

There are various open-source libraries available that could be employed to facilitate communication between the robot and the sensors or the image-guided therapy system. The robot operating system (ROS) incorporates features for computer vision, sensing, kinematics, simulation, and motion planning. Various research medical robots such as Raven II by Kazanzidesf et al. [2014], da Vinci research kit (dVRK) by Leonard et al. [2014], and the KUKA lightweight robot have used ROS as a software platform as reported by Frank et al. [2017].

ROS provides a set of libraries and tools along with reliable communication between different processes on localhost or across multiple machines. Andreas et al. Bihlmaier et al. [2016] has successfully shown in their work the capability of ROS based robot-assisted (minimally invasive) surgery.

As mentioned earlier, the Franka Panda robot is used in this project to demonstrate the needle positioning system. The main reasons for selecting ROS as middleware are:

- Wide usage of ROS in surgical robot's research [Frank et al. [2017]; Leonard et al. [2014]; Kazanzidesf et al. [2014]]
- Franka Panda's official support for ROS interface, "franka_ros". Figure 2.12 shows the Franka ROS packages as mentioned in the official documentation by Franka Emika.
- Strong community support and stable implementation

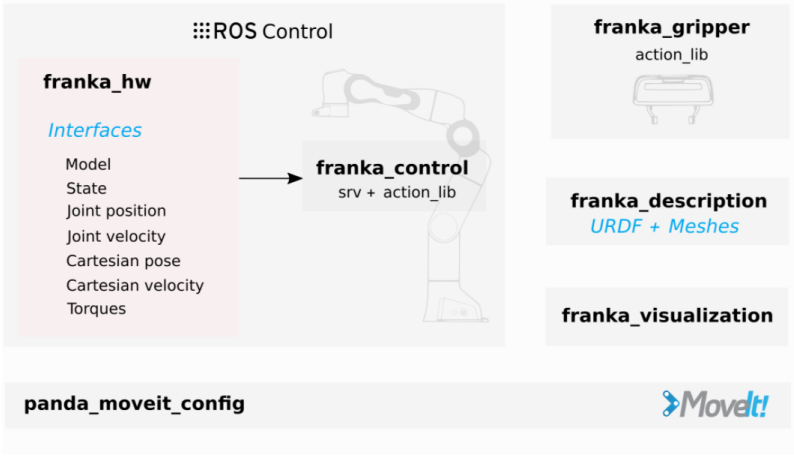


Figure 2.12: Franka ROS packages Franka Emika [2020b]

ROS implements the functionality of passing messages between processes, and package management. ROS provides a powerful tool, ROS-Bag to record and playback the recorded data. These bags can store one or more ROS topics in an efficient file structure. These bags are primarily used for offline analysis, visualisation, and storage. The recorded messages are published based on the rate at which they were recorded.

The ROS uses built-in packages for sending and receiving images and positions using image messages and PoseArray or PointStamped messages. For each message, the timestamp at which the message was recorded and the frame could be provided in the header.

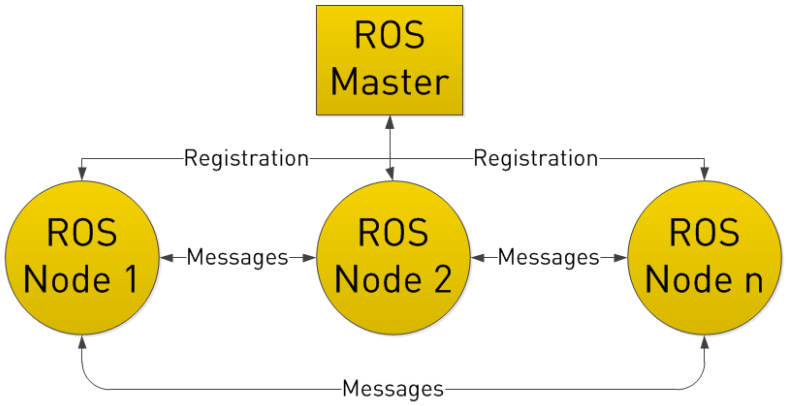


Figure 2.13: Figure showing an example of communication between various ROS nodes and their registration with the master node (Clearpath Robotics [2020])

As shown in Figure 2.13, the active ROS nodes connect to other nodes directly with the help of the master node. The master node keeps the information about all the active nodes. The subscriber nodes request connections from the publishing nodes and establish connections over a connection protocol. The most common protocol used in a ROS is TCPROS, which uses standard TCP/IP sockets.

3 Needle positioning system

This chapter starts with the system requirements followed by the system design and explaining the phases of the needle positioning system. Furthermore, in the later sections, the methods and the modules are discussed along with the system implementation.

3.1 System requirements

Based on the research goals and the available literature, the aim of the system of interest is to assist surgeons in needle placement for liver biopsy. The system should be capable of assisting the surgeons in placing the needle for the liver biopsy procedure. The system should satisfy the following requirements:

1. The system should be able to detect the location of the target liver tumor with the help of CT scan images.
2. The system should be able to predict the location of target tumor with the help of surrogate signals.
3. During the motion estimation and robot positioning phase, the system should be able to process the surrogate signal with a speed of at least 15 messages per second. The Philips Allura imaging system publishes the x-ray images and surrogate signal at a rate of 15 messages per second. Therefore, the system should be able to process the surrogate data in less than 67 ms to utilize the imaging system at its full capacity.
4. The system should be able to position the needle with better accuracy than the current clinical practice, that is the free-hand method. As per the literature, the needle placement error should be $\leq 5mm$ as reported by Lobbes et al. [2018].

3.2 System under consideration

This section gives an overview of the system under consideration. Keeping in mind the objective of the overall system, the system components can be divided into two: hardware components and software components. These components are briefly discussed here along with the system architecture and will be discussed in detail in later sections.

The major hardware components are as follows:

- **Imaging system:** One of the main component of the system is the CT scan machine that helps to locate the actual tumor location in liver.
- **Surrogate signal:** The 3D camera system integrated with the Philips Allura interventional imaging system. It is used to locate the position of the optical markers placed on the skin of patient.
- **Respiratory motion phantom:** The patient model that mimics the respiratory motion induced on the liver.
- **Robotic manipulator:** The robotic arm that is used to position the mechanical guide for needle insertion.

Along with the hardware, the software plays an important role in developing a system with good precision and better results. For the needle positioning system, it is crucial to have a short response time. The major software components that were implemented for this system are as follows:

- **Processing data:** The data received from the interventional imaging system cannot be used directly and need to be processed. The data processing is sub divided in two:
  - *Processing surrogate signal*
  - *Processing CT scan images*
- **Respiratory motion model:** A model is trained to find the correlation between the surrogate signal and the displacement of tumor.
- **Robot controller:** A controller is an important component that controls the movement of robotic arm. It sends torque commands to the joints.

Figure 3.1 shows the block diagram of the needle positioning system with the various components as discussed above.

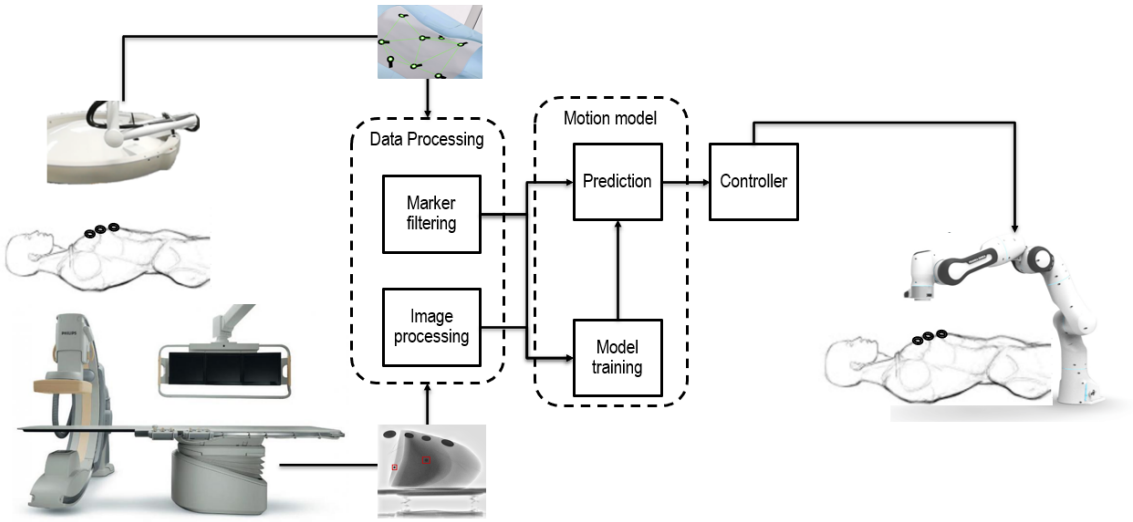


Figure 3.1: Needle positioning system with CT scan machine (bottom left), 3D camera system (top left), and robotic manipulator (right), the dots on the body of patient represents optical markers. The arrows show the flow of data through various processing nodes.

For the needle positioning system shown in Figure 3.1, a software algorithm has been developed as shown in Figure 3.2. The software is developed using a combination of Python 2.7 and ROS Kinetic. All the software components were developed as the part of this study except the implementation for the controller for robot and the software for respiratory motion phantom.

Due to COVID-19 regulations, it was not possible to test the system by integrating with the CT-scan machine and the 3D camera system. Therefore, two separate software nodes were created to publish the recorded data locally.

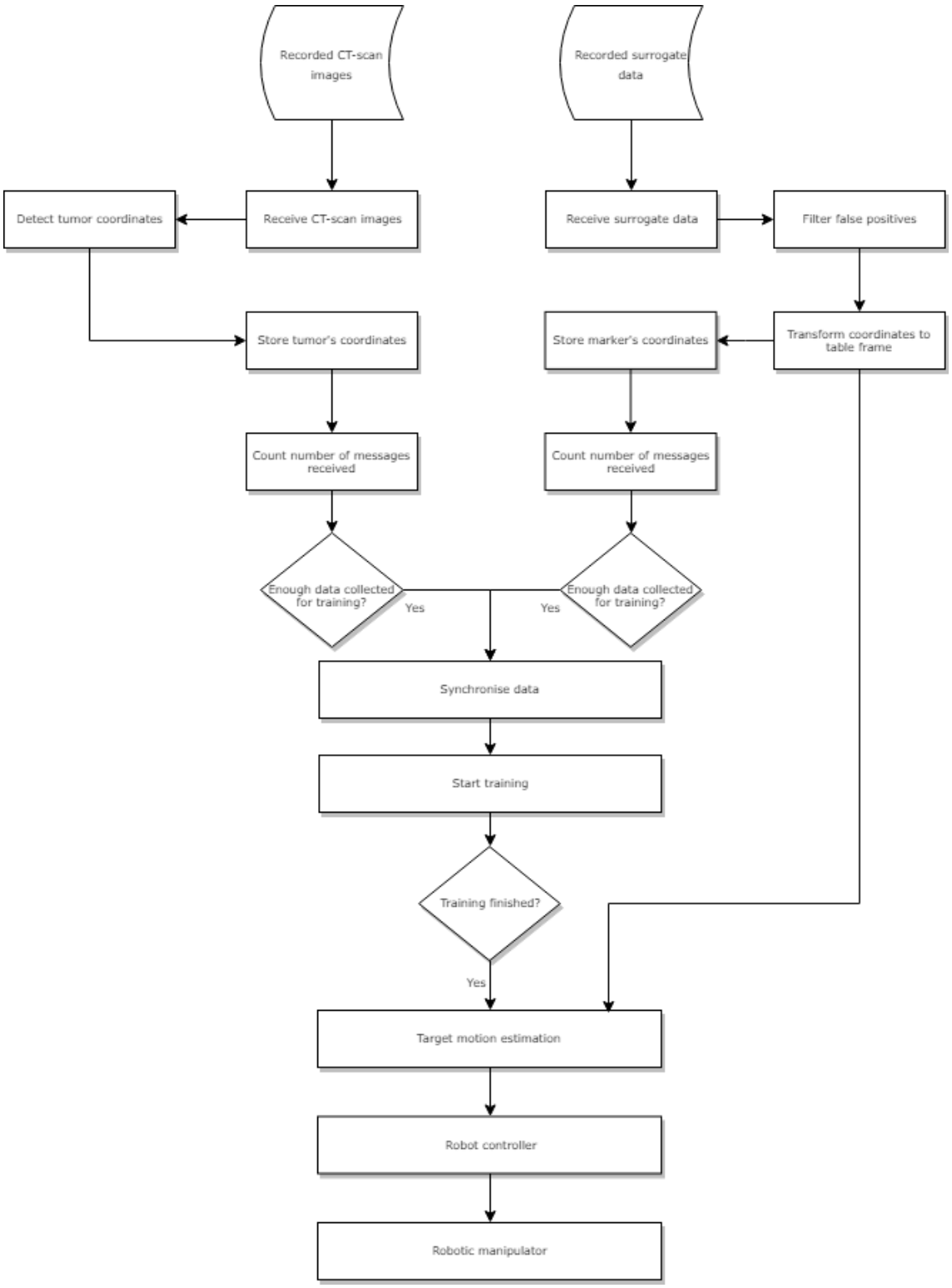


Figure 3.2: Flow of data. The pre-recorded CT-scan images and surrogate data are published from software nodes and are processed in two separate threads. Once the learning is finished, only surrogate data is used to estimate tumor motion and to position the robot.

3.2.1 Software layers

Figure 3.3 shows various software layers that operate are different frequencies. The robot controller is hard real-time and updates the state of the robot every 1 ms. As mentioned in Section

2.5, the robot communicates with the workstation PC through an Ethernet cable in real-time, that is at 1 kHz. On the workstation PC, the real-time loops of the motion generator are defined by a callback function that receives the robot state and the duration of the last cycle (1 ms unless packet losses occur) and returns the specific type of interface. Motion generators define the robot motion in joint or Cartesian space. The source of CT scan images and the surrogate data, the processing nodes are soft real-time in nature. These blocks on workstation PC pass the data to each other using a publisher-subscriber pattern of ROS.

As mentioned in the requirements above, the images and the surrogate data are published at a rate of 15 messages per second. That means a new message is received after every 67 ms. Therefore, the robot holds the last published target tumor position until the new coordinates are published by the system.

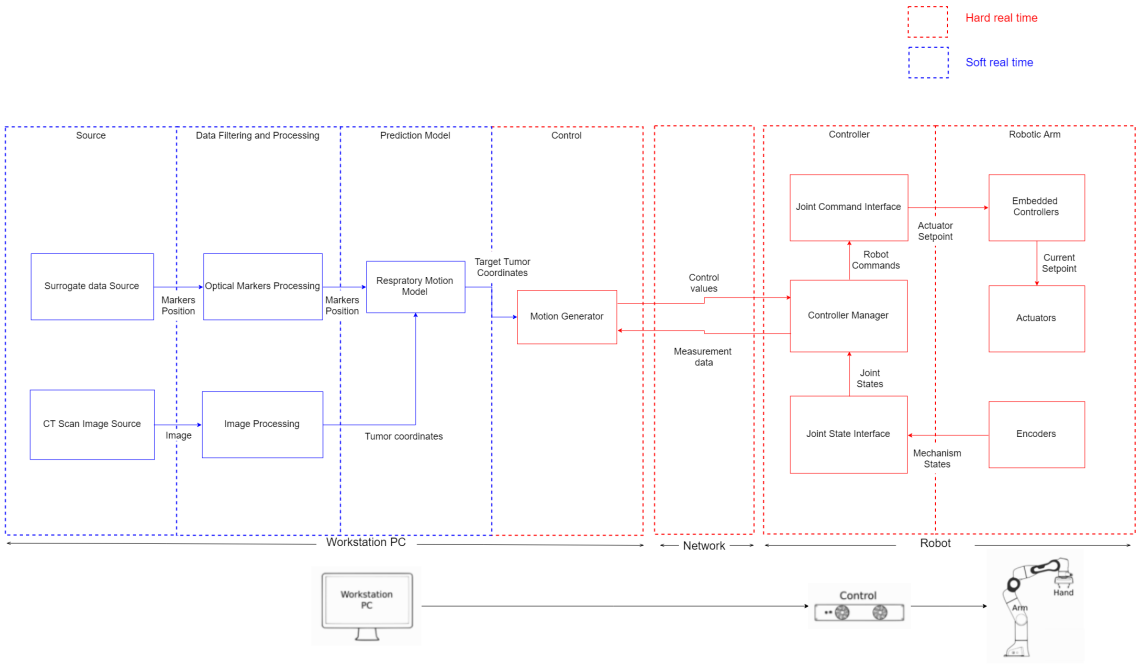


Figure 3.3: Software layers showing the soft real-time and hard real-time components of the system

3.3 System modules

3.3.1 Phases of the needle positioning system

As per the requirements mentioned above, the needle positioning system's functioning is divided into three phases: calibration phase, training phase, prediction phase. Figure 3.4 shows various phases of the over all system. The system starts with the calibration phase where all the individual components of the system are arranged and calibrated for the patient-specific environment. Followed by the training phase and prediction phase.

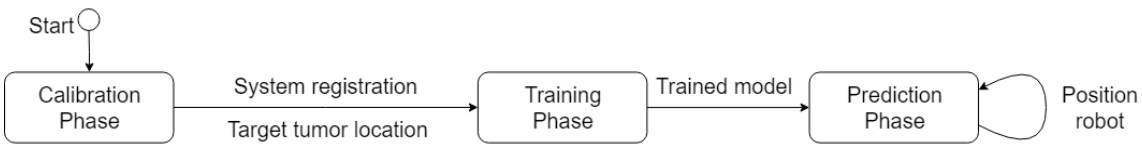


Figure 3.4: Phases of the designed needle positioning system.

During the training phase, the system will record the movement of the optical markers and the target tumor. As soon as enough data is collected for the training purpose, the correlation

between the surrogate signal and the target tumor's motion is obtained by training the respiratory motion model, as presented in Section 2.6.

Once the training is over, the prediction phase starts. In this phase, only the surrogate signal is used to locate the tumor's position with the motion estimation model. The system stays in the prediction phase after that until stopped manually.

3.3.2 System calibration

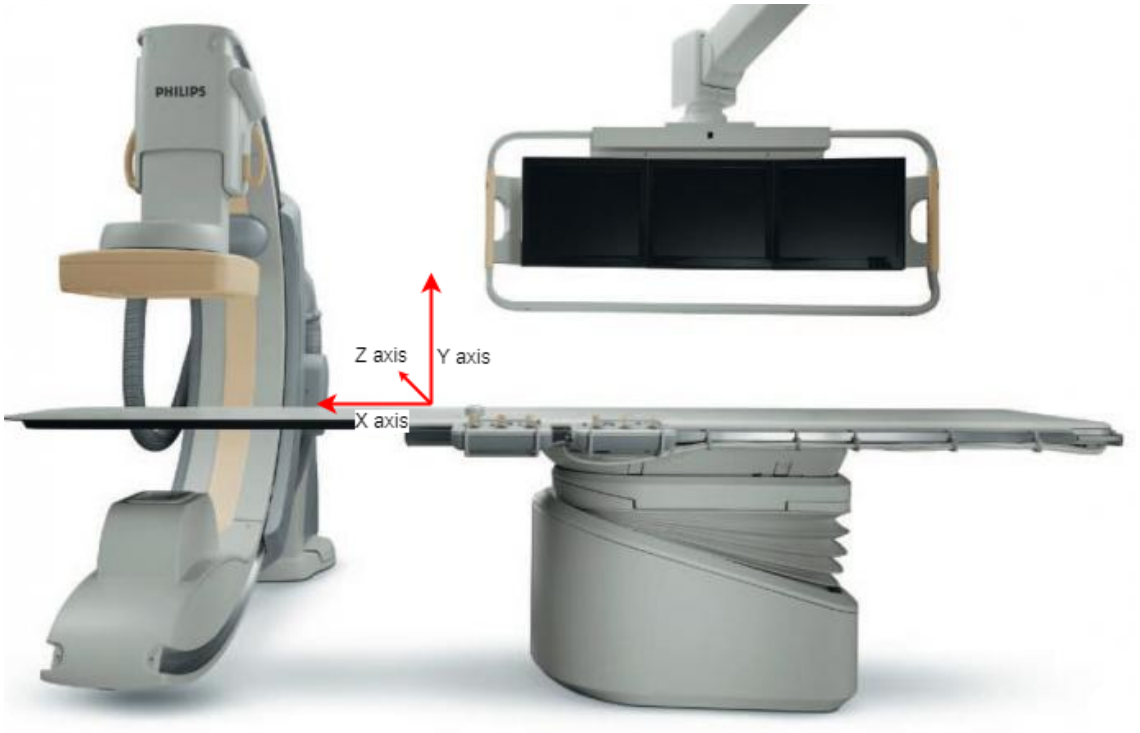


Figure 3.5: Illustration of Philips Allura table frame. The x-axis points towards the head of the patient or the cranial direction , y-axis shows the anterior direction and z-axis points towards the medial direction.

The system starts with the calibration phase, here the position coordinates of the optical markers are transformed to the table frame from the camera frame. The markers' positions are expressed in 3D coordinates in millimeters in the table frame. For calibrating the camera to calculate real-life distances between points, a reference is required. The calibration method for calculating displacement using surrogate signal is discussed later in Section 3.3.5. Precise calibration is required for robot interaction with the real world.

Once the coordinates transformation is done, the pre-processing operation to extract the target of relevance is performed on an initial X-ray image. Manual inspection is done for selecting the target tumor. Based on the shape and size of the tumor, filters could be used to appropriately find the target tumor in later stages.

3.3.3 Processing data

The images and marker data are coupled together as we need the actual tumor location along with the actual marker location to train the motion model. Therefore, a common processing node was created for both images and markers. Having a single data processing node for the images as well as the optical markers helps in reducing the timing overhead created by sending the tumor and marker's location between the nodes.

The ROS code was extended to provide a common and open interface that interconnects the robot with processing nodes. ROS helps to monitor, command and simulate the needle posi-

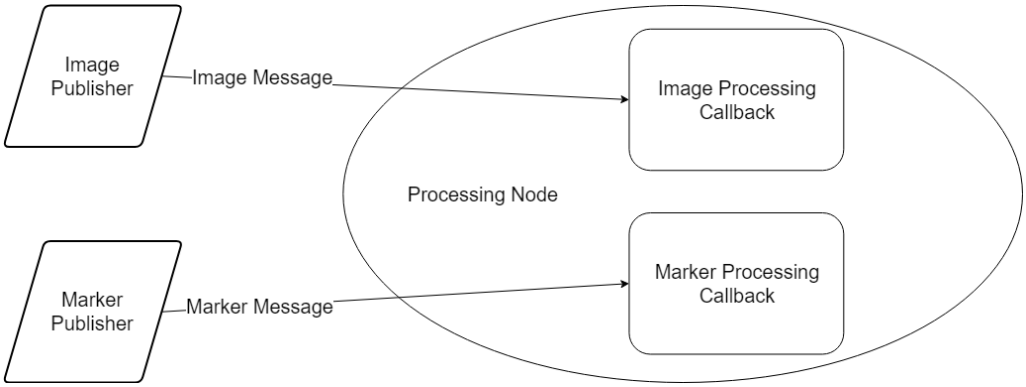


Figure 3.6: Data processing node, The images and the markers (surrogate) both are processed in a single node with separate callback functions.

tioning system. Figure 3.7 shows the flow of data between the active ROS nodes using rqt_graph tool suite. Here, *play_markerdata* and *play_Imagedata* are the publisher nodes for surrogate data (marker coordinates) and CT-scan images respectively. They publish the optical marker data and CT-scan images over the topics *BagMarkerPosition* and *BagImageList*. These topics are subscribes by the *Processing* node that performs image processing and surrogate signal processing as described in the sections below. *Processing* node publishes the *equilibrium_pose* topic, this topic controls the position of Franka Panda robot via *franka_control*.

The next two sections will explain the details of the functionalities of the two functions.

3.3.4 Processing images

Whenever a new image message is received, it is first converted to an OpenCV image matrix. After manually inspecting the decoded image, based on the shape and size of the target tumor, image processing filters could be used. As described in the previous chapter, there are multiple ways to detect the center of the tumor and after the comparison, the blob detection technique was used.

In the available dataset, the tumor is a metal bullet that forms a circular contour. The parameters used for detecting the target were specific to the data. As shown in Figure 3.8, there were multiple tumors present in the liver while recording data. The goal of this assignment was to detect the smaller one placed close to the edge. As the target is circular in nature and smaller than the other tumor present, a filter for size and inertia of blob was used to eliminate the second tumor and the elliptical markers placed over the liver present. The images are assumed to be aligned with the table frame i.e. the table's x-y plane is the sagittal cross-section plane of the patient.

For accurate detection of the tumor, one should carefully choose the filtering parameters for selecting the target. If there are multiple tumors present, there is a high chance that different tumors are detected for different input frames. Also, there is a possibility that the artifacts generated by the tumor, the trail of small circles next to the detected tumor in Figure 3.9 get detected as a tumor.

3.3.5 Processing surrogate data

The markers received needs to be processed before feeding them to the motion model. A series of processing and filtering was done as described by Hoitzing et al. [2019] in his work.

The marker's position received from the CT scan machine is published in the camera coordinates, this needs to be converted in the frame of the table. Also, the marker's data may consist of

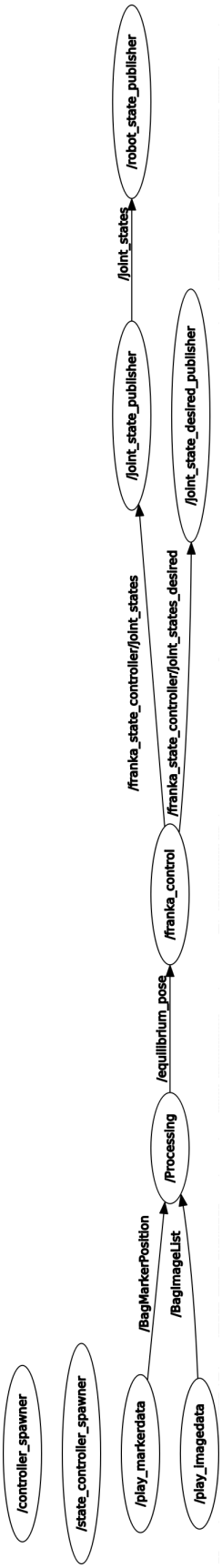


Figure 3.7: Active ROS Nodes [bottom to top] showing the end to end flow of data and processing nodes for the designed system

some false positive marker coordinates generated due to the reflective spots or objects present in the room.

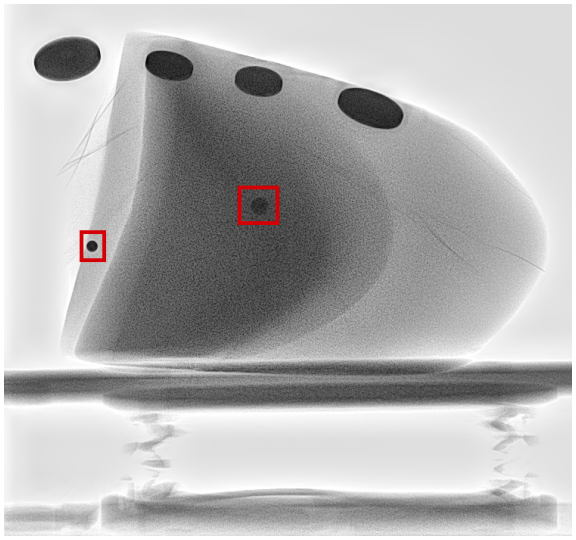


Figure 3.8: CT scan image of the liver phantom. The tumors present in the liver are highlighted with red boxes. The four dark ellipses on top are optical markers attached to the latex sheet that have been used in previous experiments at the University of Twente.

Figure 3.9: Circular artifacts produced due to motion of liver can be seen inside the red box. The four dark ellipses on top are optical markers attached to the latex sheet that have been used in previous experiments at the University of Twente.

The markers are transformed in the table's frame such that the x and y-axis of the table frame are parallel to the patient's longitudinal and sagittal axis. The transform calibration was done using a separate dataset. The calibration dataset consists of the motion of markers in the table frame is expected to follow the motion of the table. For calibration, the patient table was first moved only in the x-direction and then purely in the y-direction. After moving the table in one direction, the unit vector is calculated in that direction by determining the difference between the coordinates of markers in the previous frame and the current frame and normalizing it by the euclidean norm. Once the unit vectors in x and y directions are calculated, the unit vector in the z-direction, $\hat{z}$ is computed by cross multiplication of $\hat{x}, \hat{y}$. The transformation matrix H_{CT} is shown below:

$$\mathrm{H} = \begin{bmatrix} R_{CT} & -p_C^0 \\ 0_{1x3} & 1 \end{bmatrix}$$

Here, R_{CT} is the rotation matrix that converts the points expressed in the camera frame to the table frame. The rotation matrix is a 3x3 orthonormal matrix of unit vectors expressed in the original frame. p_C^0 is the origin of the new frame, i.e. the table frame.

After the calibration is successful, the markers could be expressed in the table frame. The marker filtering is performed based on the movement of markers in the current frame to the previous frame. The filtering is divided into two parts as follows:

- **Distance based filtering:** In this approach, the Euclidean distance is computed between previously detected markers and the newly detected markers. It is assumed that in consecutive frames the markers will not move much therefore the markers with minimum Euclidean distance are selected.
- **Directional filtering:** In this approach, the movement vectors of the base marker is compared with the other markers.

3.3.6 Training and prediction

As per the literature, the liver motion is mainly due to the respiration and the consequent displacement of the diaphragm. As the setup is patient-specific, the breathing cycles are represented as linear, and thus the linear fitting method was used in this project. To train the model for predicting the target motion, supervised learning is used. The linear regression technique is a type of supervised learning that is widely used for continuous response problems. It is a fast and highly interpretable way of learning.

The model used for training is based on the results of Fahmi et al. [2018]. In his work, he reviewed regression models with different parameters, based on his results the best model is selected that is, linear regression using Lasso regularisation. Also, as mentioned in Scikit-learn's official documentation, Lasso regression is best when the dataset is small and not all features are important (Pedregosa et al. [2011]). Lasso with L_1-regularization scales down the weights of big parameters to small and that of small parameters to zero. The cost function of Lasso regression is described as:

$$\frac{1}{m}\sum_{i=1}^{n}(y_i - x_i'\hat{\beta})^2 + \lambda\sum_{j-1}^{m}|\hat{\beta}_i|$$

Where λ is the regularisation parameter. By adding the regularisation parameter, the estimation error for the training set increases but it decreases for the unseen or the new data. Regularisation helps in avoiding the over-fitting problem for highly correlated data.

After the training is finished, the predicted target location could be calculated using the following equation:

$$Y = \beta_0 + \beta_1 x_1 + \beta_1 y_1 + \beta_1 z_1 + ... + ... + \beta_n x_n + \beta_n y_n + \beta_n z_n$$

Here, Y is the predicted tumor location (dependent variable) and $x_1, y_1, z_1 ... x_n, y_n, z_n$ are the coordinates of marker 1 to n table's frame and betas are the coefficients or the weights assigned to each feature. We can see that the target is located only based on the location of the position of the markers in 3D space.

Tumor location prediction

The tumor's location is predicted based on the surrogate signals, i.e. the filtered markers. The tumor's actual location, obtained by the CT-scan images and the corresponding surrogate signals are used to train the regression model for normal breathing cycles.

Once the model is trained, only surrogate signals are used to locate the tumor's actual position in the 3D space. For validation purposes, the CT-scan images are also used in the prediction phase to calculate the error in prediction.

3.3.7 Robot positioning

For this project, the Franka-Emika Panda robotic arm was used. In the proposed system, the robot manipulator will work in conjunction with Philips' interventional x-ray system. The position of the robotic manipulator will be fixed with respect to the CT scanner. The registration between the patient's table frame and the robot frame is critically important. The needle positioning commands and control paths are defined in the robot's frame while the motion of interest is in the patient table frame. For an accurate needle positioning in robot frame the based on the liver motion in the table frame, the transformation from the table frame to the robot frame is required.

In literature, various methods are available to calculate the transformation between the robot frame and the CT frame as mentioned by Kettenbach et al. [2005] and Solomon et al. [2001]. One of the common methods is to place a fiducial marker on the robotic manipulator close to the needle gripper.

The Franka Panda robot has two network ports, one connects to the internet so that the robot can be programmed using the GUI and the other port connects to the user's computer and runs real-time robot control code using UDP messages exchanged every millisecond. The Franka Control Interface (FCI) allows low-level bidirectional connection to the arm. The low-level controller, FCI communicates with *libfranka* interface using Ethernet. *Libfranka* interface allows sending realtime control values to the arm at 1kHz.

The needle positioning system requires the manipulator to stably position the end-effector. There are two main categories of force control schemes position-force control and impedance control. An impedance controller has been used to implement the dynamic relationship between robot motion and the external torques because the position-force control scheme does not take into account the dynamic interaction between the robot's end-effector and the environment (Al-Shuka et al. [2018]). Impedance control is inspired by the human behavior during contact with different environments. Humans have a considerable amount of adaptability to change muscle impedance (e.g., stiffness) when in contact with an unknown environment. If the environment is stiff, the robot should be soft and vice versa. As mentioned by Hogan [1984] in his work, the impedance or the stiffness of the manipulator should be selected appropriately, there is always a trade-off between allowable interface forces and the allowable deviations from desired motion. For this project, the default translational and rotational stiffness of Libfranka's cartesian impedance controller has been used. The translational stiffness of the end-effector was set to 150 N/m and the rotational stiffness was set to 10 N/rad.

4 Validation

This chapter discusses the experiments that were specifically designed to validate the tumor location prediction results and to evaluate the time required to process the incoming data and sending a control signal to the robot. The validation for motion estimation was done by calculating the error in the actual tumor's location and the predicted tumor location. For the timing analysis, the maximum execution time was calculated for the system.

4.1 Overview of the experimental setup

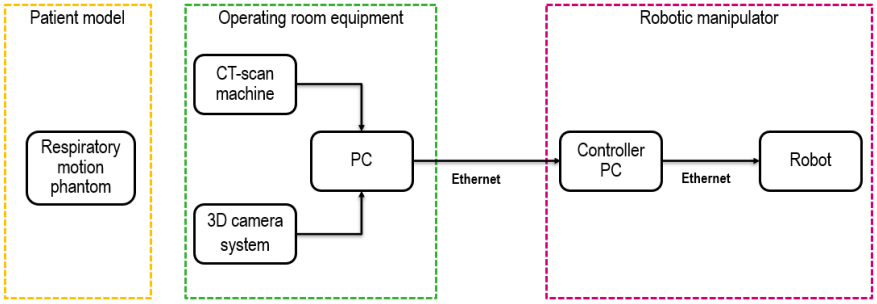


Figure 4.1: Block diagram for the hardware setup for the needle positioning system. The PC of the imaging system connects with the controller PC of the robotic arm via Ethernet. Similarly, the controller PC connects with the robotic arm via Ethernet that helps in establishing real-time communication between systems. The respiratory motion phantom is kept on the patient table of the CT-scan machine.

As mentioned in Section 3.2, there are several hardware components required for the system. The hardware setup for the needle positioning system is divided into three parts as shown in Figure 4.1.

- The CT-scanner, Phillips Allura system for imaging the liver tumor and providing surrogate data.
- The respiratory motion phantom. The robotic phantom and its peripherals to mimic the motion induced in liver due to patient's breathing cycles as described in Section 2.3.
- The robotic manipulator, Franka Emika-Panda robot as described in Section 2.5.

Figure 4.2 shows the motion phantom placed on the patient table. The motion phantom is covered with a green colour latex skin with the optical markers placed over it. The C-arm is rotated to the sagittal plane with respect to the patient table. The movable lamp with the inbuilt 3D camera system is placed over the phantom in such a way that all the markers are visible during the whole breathing cycle. After positioning the motion phantom correctly and placing the optical markers over the skin, visual inspection of the motion is performed to make sure the motion of skin and phantom is satisfactory.

Figure 4.3 shows the robot used for the demonstration of the needle positioning arm. The robot has its own dedicated hardware controller that is connected to a PC from which the robot state could be controlled by a user. The PC contains an Intel Core I7-7700 processor with a 16GB DDR 4 RAM, running a Linux with PREEMPT_RT patched kernel with Ubuntu 16.04 LTS Bionic Beaver as the operating system and the ROS Kinetic as middleware.

Due to COVID-19 regulations, the complete system was not validated by interfacing all the components of the hardware. The data recorded using Philips' Allura system for a previous

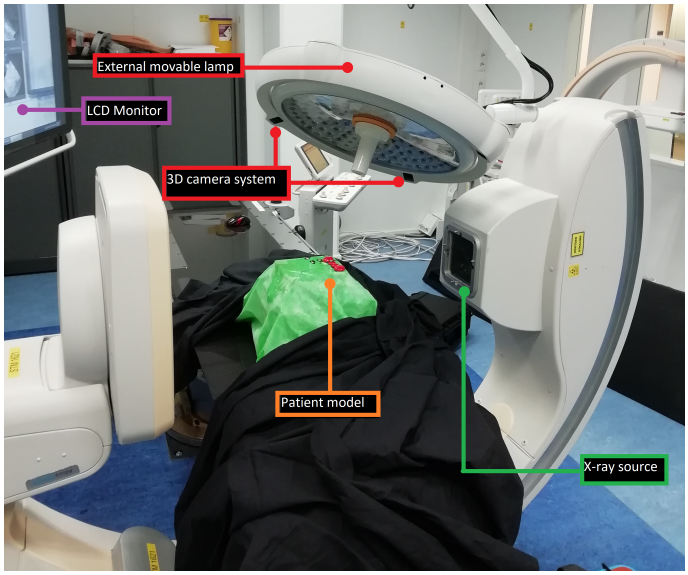


Figure 4.2: Philips Allura CT scanner positioned to capture the motion in sagittal plane. The liver phantom was placed under the green coloured latex skin. The 3D camera system can be seen embedded on the external movable lamp positioned over the liver Hoitzing et al. [2019].

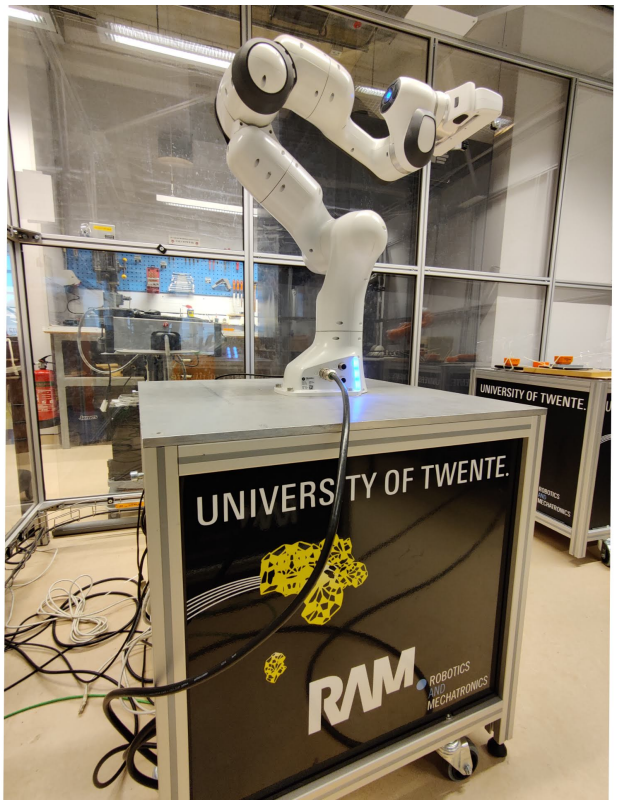


Figure 4.3: Franka Panda robotic manipulator mounted on the aluminium frame containing the dedicated controllers.

study was used. The same respiratory data was used for both the experiments discussed below. The data was sufficient to demonstrate the proof of concept and answer the research questions defined earlier.

As shown in Figure 4.4, the software setup has been employed such that the CT scan images and marker positions could be published locally from the PC that was used to control the robot. The

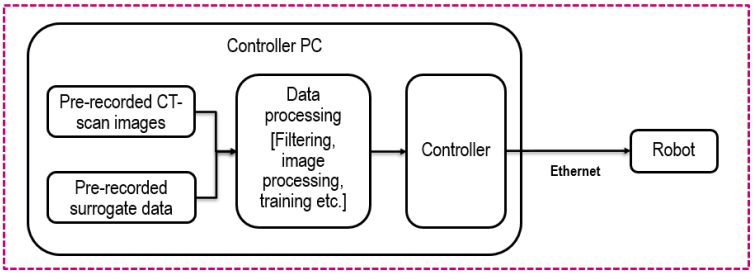


Figure 4.4: The final adapted experimental setup with software nodes to replace the CT-scan machine and the 3D camera system

flow of data in the software is shown in Figure 3.2. Assuming that when tested with an actual CT scan machine, the images and the surrogate data will be published at the same rate. Figure 3.6 shows the rqt graph for the active ROS nodes when the application is running.

This validation setup only reports about the part of the needle placement procedure. That means this study only focuses on the time taken for the data collection, training respiratory motion model for estimation of target tumor motion, and positioning the robotic arm.

4.2 Dataset

The available pre-recorded dataset consists of a total of 4 breathing cycles with a total of 280 data points. The first three breathing cycles were used for the training purpose and the 4th breathing cycle was used for the validation purpose.

The available CT scan images are stored in a binary file format. Each binary file contains a header that gives the details about the timestamp at which the image was captured and the size of the image followed by the image data. The marker's data was stored in a separate file. The marker data also consists of timestamps along with the number of markers detected and then X, Y, Z coordinates of the markers. These coordinates are expressed in the camera frame and thus requires processing before it could be used. Since the experiments were performed on ROS as the interface to connect multiple systems, the data was required to be converted in a suitable format with some pre-processing as discussed below.

If the image processing part is performed locally at the CT scan machine, the additional step of converting CT images to binary files and back to readable OpenCV image matrix could be avoided. If only the actual tumor location is published from the CT scan machine, it will reduce the subscribers in the data processing node, as a result, the processing time could also reduce as there will be only a single processing thread running in the ROS node.

4.2.1 Data pre-processing

As ROS cannot transport images in binary format, all the available files were first converted into an OpenCV matrix. As per the timestamp available in the header of each binary file, the list of images was arranged sequentially. Similarly, all the markers are read from the marker's file and were arranged sequentially based on their timestamp. Once the data is arranged in chronological order, they are stored in a rosbag file. A rosbag file is a set of tools provided by ROS to record or play the recorded data. The data recorded in the rosbag file is published exactly in the order it was recorded. Rosbag is a high-performance tool that avoids deserialization and reserialization of the messages when recording or playing the data.

The OpenCV images cannot be stored in the rosbag file directly, a ROS built-in tool, CVBridge is used to convert the images to ROS message. These images are then recorded in a rosbag file

with a loop running at a controlled rate. The available data was published at a rate of 15 images per second, therefore the rosbag file must store only 15 images per second.

This conversion was one of the major drawbacks of the previous work. This process is time-consuming and can be avoided. This conversion will not be required when the system is tested by integrating the CT scan machine directly with the controller PC as shown in 4.1.

4.3 Experiment 1: Real-time tumor motion estimation and needle positioning

The system is required to perform real-time only in the prediction phase where robotic assistance is required to the surgeon for performing the surgery. Therefore, for live tracking of the target tumor, the timing analysis was done for the prediction phase.

As shown in Figure 3.2 when the system is in prediction phase, only the surrogate data is processed for target motion estimation and the controller translates the predicted location to a real-world coordinate for the robot. The control loops of the Franka Panda robot runs at 1 kHz frequency. The controllers for the Franka Panda robot run on RTOS and are strictly real-time.

To evaluate if the setup could run with real-time processing capability in the prediction phase, the execution time of the designed system was logged. The wall time API of ROS was used to log the timestamp for every new set of markers or images received. For the setup to behave like a real-time system, it should be able to process the surrogate signals and estimate the location of the tumor for all the received input data points. In other words, before a new message is received the previous message should be processed and the predicted location of the target should be sent to the robotic arm.

By analysing the difference between the processing time of the consecutive set of markers allows us to calculate the maximum jitter (J) over the whole sample:

$$J = \max_{k}(|T_k - T_{k-1}|)$$

Here, T_k is the time taken for processing the k^{th} set of markers, and T_{k-1} is the time taken for processing its previous markers. For evaluation of the system in terms of processing speed, the total execution time is calculated by considering the maximum time taken by any of the data points in the whole set.

Procedure To start with the experiment, first, the Franka Panda robot is unlocked and the brakes are released. Next, all the desktop applications that are running on the computer are closed (except the background application) and the needle positioning system is started using the Linux terminal. To start the system, a roslaunch file is used that starts the master node and the other required nodes as shown in Figure 3.7. To measure the response time of the system while needle positioning, an internal flag is used in the software for distinguishing the learning phase and the prediction phase. As soon as the training is completed the flag is automatically set and the incoming CT-scan images are not processed after that. The response time of the system is logged using the time at which the surrogate data is received and the time at which the control signal is sent to the robot.

Analysis During the training phase, if the total processing time of the system is less than the time at which new data is received then the CT scan machine and the 3D camera system can be utilized at their full capacity. The CT scan machine and the 3D camera system publishes the images and the surrogate data at a rate of 15 messages per second. Therefore, a new message is received every 67 ms and if the processing time is more than 67 ms then the input messages should be stored to avoid loss of information.

Furthermore, during the prediction phase as the robot is used to position the needle in real-time, the system should be able to estimate the target position well in time. The robot's control loops update the robot's state every 1 ms, that is the control loops are running at 1kHz.

4.4 Experiment 2: Tumor motion estimation

The second experiment that was performed was to evaluate the target motion estimation. The target location was predicted for each set of markers positions received.

As explained in the methodology in the previous chapter, the motion estimation is divided into two parts, i.e. training phase and prediction phase. Therefore, the complete data set of four breathing cycles were divided into two, three cycles for training the motion model and the last breathing cycle for the prediction and the validation.

The experiment uses five optical markers each with three dimensions, resulting in a total of 15 surrogate signals. The correlation between the surrogate signals and the target tumor could be modelled and the output of the model will be an estimation of the motion of interest.

The motion phantom used for this project considers the target motion in only two directions with normal breathing cycles. Due to the simplicity of the motion estimation problem, the linear regression algorithms are best suitable for estimation. Like an ordinary linear model, Lasso regression fits a model using least-square regression but also applies a regularization technique that prevents overfitting by penalizing large model-coefficient weights.

The model evaluation is done by calculating the prediction accuracy using Root Mean Square Error (RMSE).

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Where, n is the number of samples in the validation set, $\hat{y}_i$ is the predicted value of i^{th} sample, and y_i is the corresponding actual value. The error is calculated between the predicted tumor location and the actual tumor location as detected in CT-scan images.

Procedure For this experiment, similar to the previous experiment the application is started using the roslaunch file but the robot is not unlocked because the robot is not required in this experiment. As the aim of this experiment is to evaluate the accuracy of the tumor motion estimation, the system is required to process the CT-scan images in the prediction phase as ground truth for the tumor position. Therefore, we need to process the CT scan images in the prediction phase also. All detected target tumor positions and the estimated tumor positions during the respiratory cycle are stored for error calculations as well as the plotting graph.

Analysis The estimation error could be compared with the literature. As per the research of Lobbes et al. [2018], in most diagnostic procedures the only accuracy requirement is that the needle tip is positioned within the target. Therefore, the needle placement error should be lower than at least half of the lesion size. As reported by Welch et al. [1989], a prospective analysis of 1000 procedures found that most of the liver tumors are between 20 – 50 mm in size, with a total range of 1 – 21 cm. Therefore, a needle placement error of 5 mm would therefore be sufficient in the vast majority of liver biopsy cases.

5 Results and Discussions

This section presents the results obtained by experiments described in Chapter 4. The experiments were performed on a pre-recorded dataset from a previous study performed by Hoitzing et al. [2019] due to limited access to hardware because of COVID-19 regulations.

The results consist of a performance evaluation of the designed needle positioning system for its target motion prediction accuracy and for its response time for the input signal. Furthermore, the interpretation of results discusses the observations and outcomes of the experiments.

5.1 Real-time performance evaluation

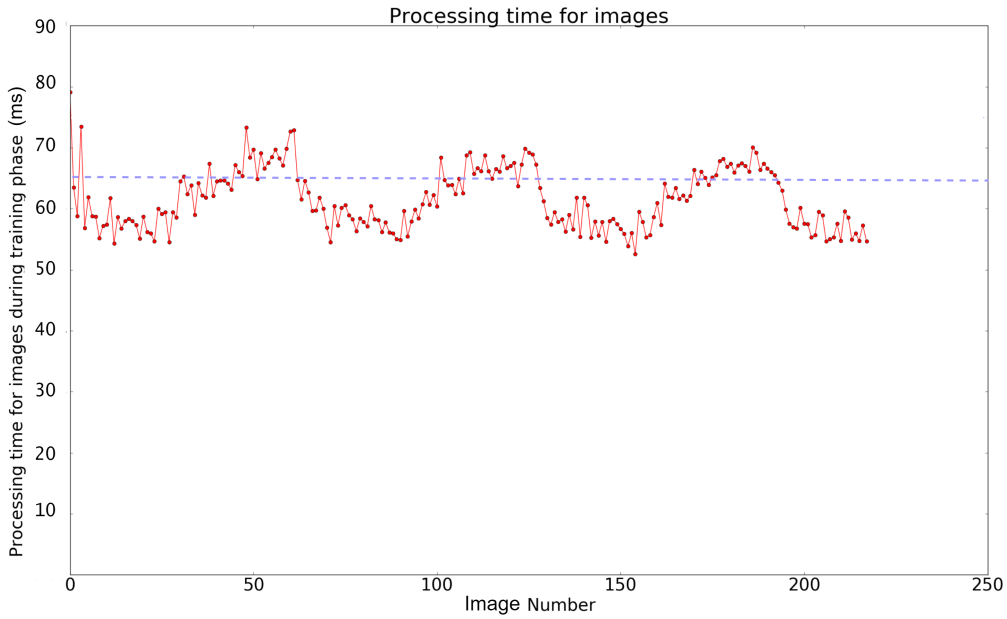


Figure 5.1: Processing time for the CT scan images received during the training phase. The red line shows the variation in processing time for the input images. The blue dotted line shows the average time at which a new image message is received for processing.

The response time was measured for the time consumed in the processing of the surrogate data, the CT scan images, and in sending the target coordinates to the robot. Time taken for processing each set of surrogate data and image was calculated as described in Chapter 4. Figure 5.2 shows the processing time of the individual set of markers in the marker filtering block for training purpose, which includes detection and threshold-based filtering. Figure 5.3 shows the processing time of markers during the validation or the prediction phase which follows the same filtering methods. It can be observed from the graphs that the time consumed for processing during the collection of data for training is less than the time taken while the prediction phase. This extra time is observed because the system calls the ROS handle to publish the tumor's location to the robot. Figure 5.1 shows the plot of the processing time for each image during the learning phase. Each image is processed to locate the target tumor's coordinate for training the respiratory motion model.

We can observe a periodic trend in processing time for images, this is observed mostly due to the periodic motion of the liver. Due to the respiratory induced motion, the position liver in the CT scan images changes periodically in AP and CC directions. This motion affects the processing time because when the tumor moves towards the top of the left direction in the

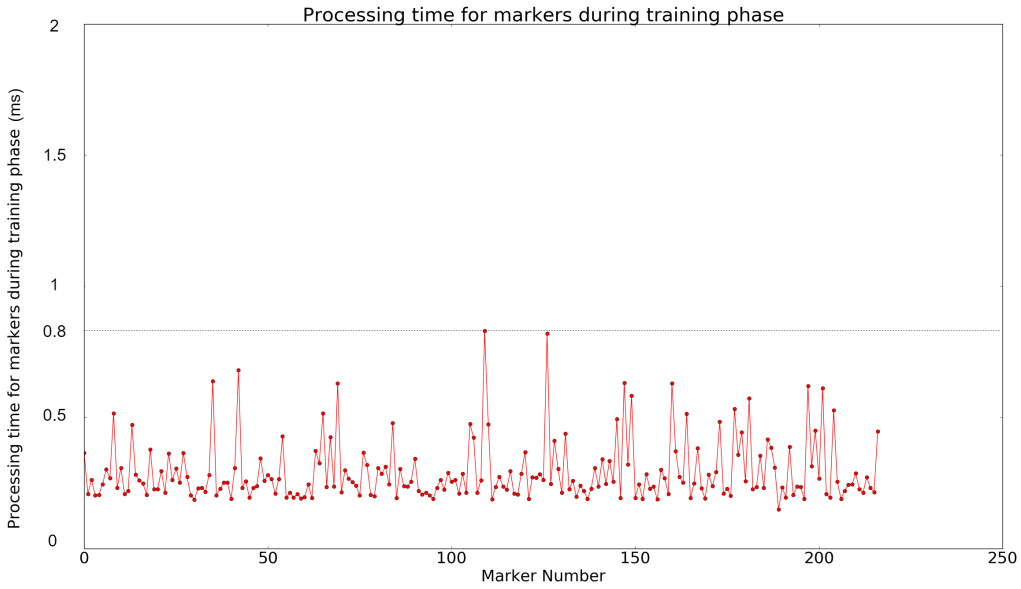


Figure 5.2: Processing time for the received surrogate data during the training phase. The red line shows the variation in processing time for the received set of marker locations. The black dotted line shows the maximum time taken to process a set of markers.

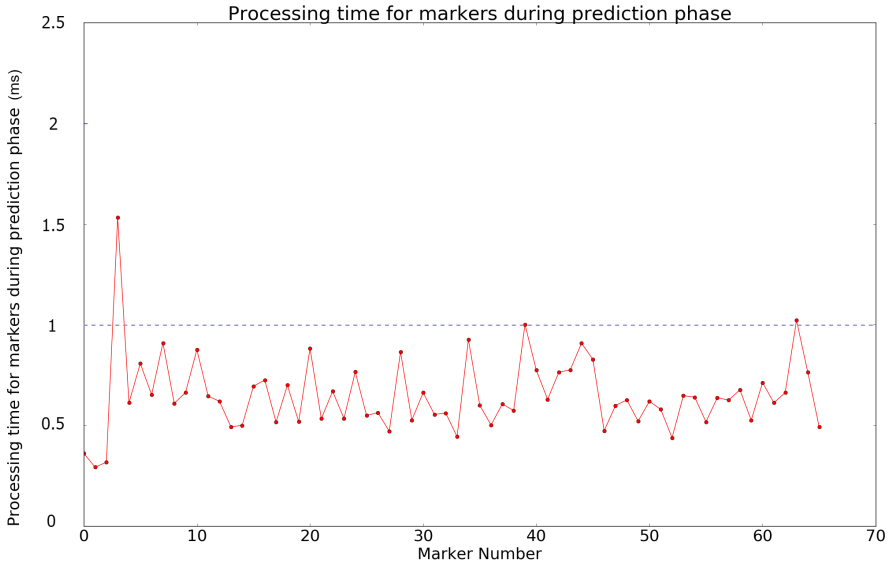


Figure 5.3: Processing time for the received surrogate data during the prediction phase. The red line shows the variation in processing time for the received set of marker locations. The blue dotted line shows the max allowable response time.

image, the time is taken to locate the tumor reduces. The blob detection algorithm used to locate the tumor starts finding the contours from left to right and top to bottom.

As shown in Figure 5.1 and 5.2, the processing time for the images is more than the processing time for markers. As for the training data set it was observed over a total of 218 data points that processing an image and detecting the tumor takes a maximum of 75 ms whereas receiving a set of the marker and filtering it takes maximum 0.8 ms. The markers and CT scan images are published at a rate of 15 data points per second which means a new message is received every

67 ms (highlighted by blue line in the Figure 5.1). Therefore, during the training phase when the images are processed, the incoming messages are stored in a queue as the processing time is more than the time between two consecutive messages received.

Once enough data is collected, it is used to train the regression model. As the dataset is small and the model is also not complex, the learning is finished quickly. It takes around 1 to 1.5 second to train the model. As soon as the model is trained, the prediction phase starts.

During the prediction phase, the markers are processed and the target location is published to the robot with an average processing time of 0.7 ms. The maximum jitter observed was 1.2 ms. Therefore, by the time a new image or a new set of markers is received the previous messages are already processed and the control signal to the robot is sent. The control loops for the motion generator and the controller for the Franka Panda robot run in real-time, that is the control loop reads the input and updates the robot's state every 1 ms.

5.2 Tumour location prediction accuracy

The target tumor's location is predicted based on the movement of the optical markers or the surrogate data. For this experiment, a total of 5 optical markers were placed on the latex skin over the liver. The actual location of the lesion has been detected using CT scan images. Figure 5.4 shows the motion of the target subjected to breathing that is from expiration toward inspiration in the cranio-caudal and posterior-anterior directions.

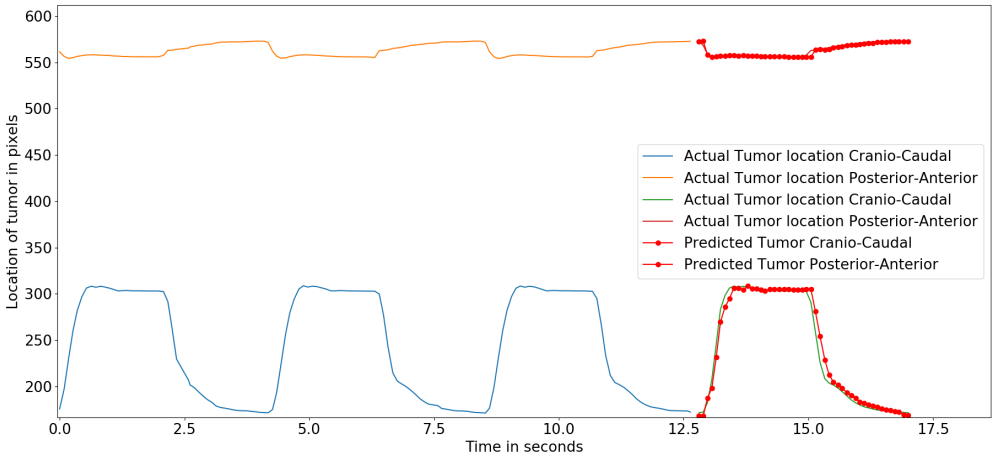


Figure 5.4: Actual and predicted tumor location in cranio-caudal and posterior-anterior direction. The first three breathing cycles were used for training and last breathing cycle was used for validation.

As the breathing cycles are constantly repeating over time, the complete data set was again fed to the modelx for visualisation purpose as shown in Figure 5.5.

It could be observed from the graph of Figure 5.4 that the tumor location prediction error is more in the cranio-caudal direction compared to the posterior-anterior. The evaluation of the regression model is shown in the table.

Direction	RMSE (in mm)	Std. deviation (in mm)
Cranio-Caudal	1.58	1.54
Posterior-Anterior	0.3	0.2802

Table 5.1: Calculated prediction error of the target tumor in CC and AP directions

For each of the received surrogate signal during the prediction phase, Figure 5.6 and Figure 5.7 shows the plot of the error calculated between actual tumor location and predicted tumor

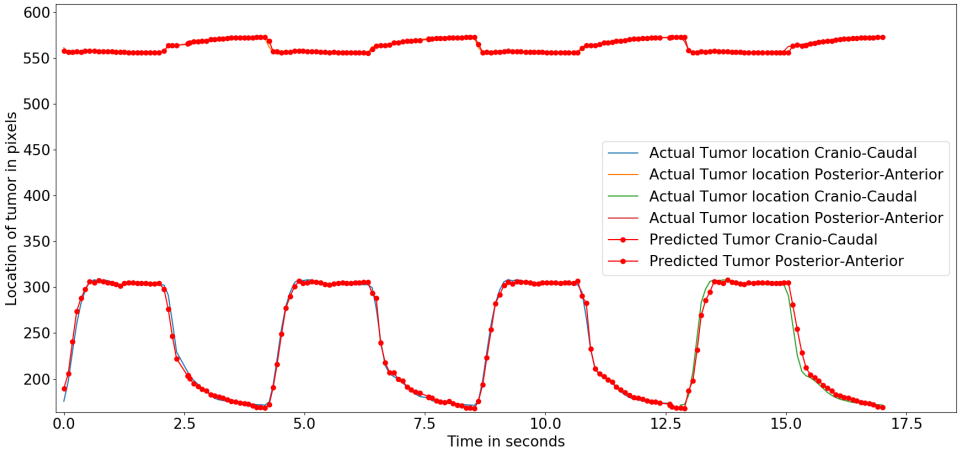


Figure 5.5: Prediction results for complete dataset, only for visualisation purpose (as the breathing cycles are constant over time). The first three cycles are for training set and last cycle is for validation set

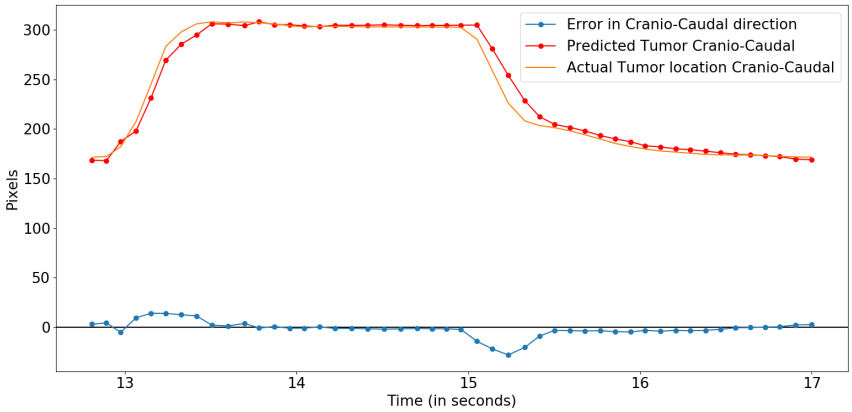


Figure 5.6: Target tumor's position prediction error in CC direction

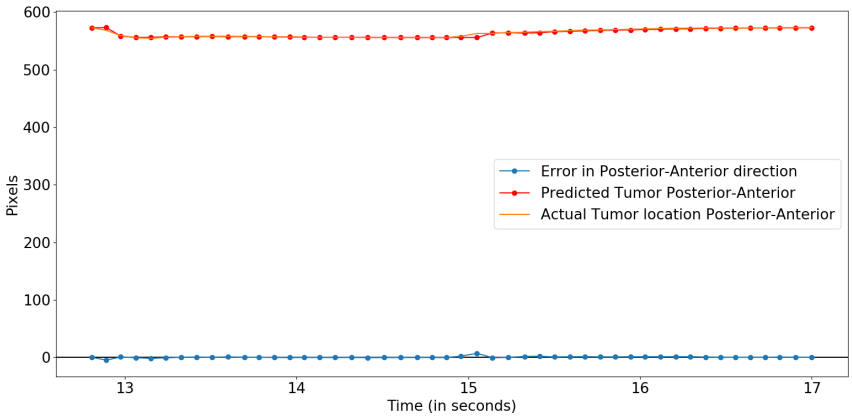


Figure 5.7: Target tumor's position prediction error in AP direction

location. The calculated standard deviation for the validation set is shown in Table 5.1. The peaks seen in the graphs for prediction error are mainly due to the change in motion from

inspiration to expiration and vice versa. We can see that the motion estimation is more accurate when the person holds breath after complete inspiration and complete expiration.

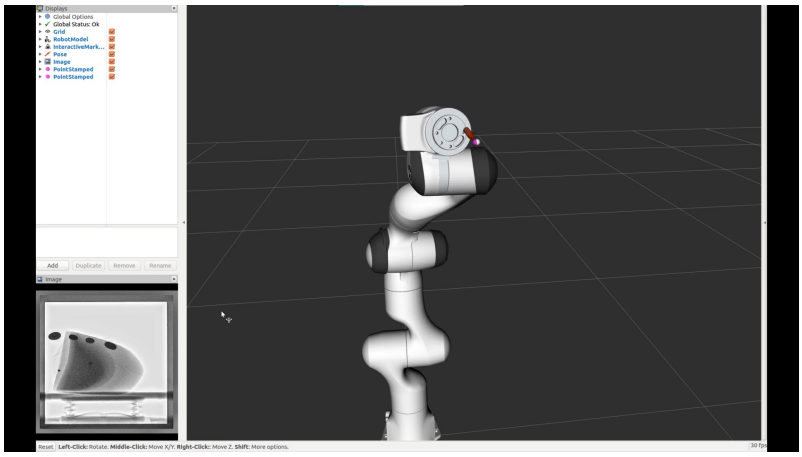


Figure 5.8: Visualization screen showing the robot pointing towards the predicted tumor location (pink ball), the actual tumor location (white ball) can be seen overlapping the predicted tumor location

Figure 5.8 shows the visualization of the actual system on RViz. The CT scan image can be seen on the bottom left corner of the image with the spherical target tumors embedded inside the liver. The robotic arm can be seen pointing towards the predicted tumor location in the pink ball and the actual tumor location in the white ball. The predicted tumor location and the actual tumor location can be seen overlapping each other.

5.3 Discussions

The observations made while doing the experiments are discussed in this section. The results are interpreted keeping in mind the system described in Chapter 3.

5.3.1 Interpretation of results

Response time:
For the performed experiments, the maximum time taken for processing the data received was nearly 76 ms (nearly 75 ms for image processing, and 1 ms for marker processing). This means the data could be processed at a rate of 12 to 13 messages per second. But, the CT scan machine publishes data at a maximum speed of 15 messages per second. Therefore, there is a need to store the incoming messages in a buffer so that none of the data points is missed. The system is required to perform in real-time during the prediction phase which is possible because, for prediction, only processing of the surrogate data (i.e. optical markers positions) is required. As we can see from Figure 5.3, the mean processing time for a set of markers is 0.7 ms with and standard deviation of 0.1 ms. The minimum time taken for processing a set of markers is 0.3 ms and the maximum time taken is 1.5 ms. The average processing time of markers was 0.7 ms which is less than the time at which the robot updates its state. Therefore, the system is capable of receiving and processing the surrogate signal without keeping any message in a buffer.

Position accuracy:
Figure 5.4 shows that the regression model with Lasso regularisation is suitable for the given scenario. The RMSE values presented in Table 5.1 shows that the error is minimal in posterior-anterior direction while there is comparatively more error in cranio-caudal direction. The target location prediction accuracy is within the boundaries of the requirement mentioned Chapter 3.

As the recorded data was limited (only four breathing cycles) for training first three breathing cycles were used and for validation of the model, only the fourth breathing cycle was used. But,

for visualisation purposes, the complete data set was once again fed to the regression model. It can be observed from Figure 5.5 that for the first three breathing cycles, the prediction is more accurate than the prediction for the fourth breathing cycle. This is because the first three breathing cycles were used for training the motion model. For visualisation purpose, the complete data set was used in the loop to mimic a continuous breathing pattern. It was suitable to use the dataset in a continuous loop due to the nature of breathing patterns used, i.e. normal breathing cycles that have a constant rate of inhalation and exhalation. The liver motion phantom used to mimic the breathing cycles of humans was programmed to reproduce the motion of the liver in normal respiration and avoiding any unexpected inhalation and exhalation motion.

It can be further observed from the graphs in Figure 5.7 and in Figure 5.6, that the prediction error is mostly observed during the inhalation and exhalation period. For a real application of the needle positioning system, this should not be of much concern as the needle will be inserted while the breath is held by the patient. The needle placement is not performed when an ongoing motion observed in the target due to respiration.

6 Conclusion and future work

6.1 Conclusion

By analysing the results obtained and the their interpretations given in Section 5.3.1, this section concludes the findings and answers the research questions described in Section 1.3.
A needle positioning system for compensating respiratory induced liver motion was designed and analysed for processing time for data and prediction accuracy of tumor motion. The research aimed at extending the previous studies at the University of Twente on on respiratory motion modelling.

The designed system adapts to the patient's respiratory patterns and positions a robotic manipulator for performing diagnosis or treatments for liver tumor. The experiments were carefully designed for validating the motion prediction accuracy and analysing the response time of the system during the learning phase and the robot positioning or prediction phase. Once the learning phase is over and the motion model is ready, in the prediction phase the target tumor location could be predicted in real-time with the help of the optical markers.

The following is research question which has been studied:

How to develop a needle positioning system for compensating the respiratory induced liver motion in real time?

For the given system the research question is supported by two sub-questions for validation:
The first sub-question that concerns the response time:
How fast the designed system can process and respond to the incoming surrogate signals and position the robotic arm for needle insertion? Is it possible to compensate for respiratory induced motion in the liver and position the robot in real-time?

The system can run the motion estimation algorithm to locate the current location of the tumor and to position the robotic manipulator in real-time. The model has been trained on a small dataset of just 3 breathing cycles and a linear regression algorithm. The total training time was less than the time for half the breathing cycle that is, around 1.5 s. The average response time of the system (in the prediction phase) to the input surrogate signal over 48 samples is 0.7 ms. A new surrogate signal is received every 67 ms, which gives the system enough time to process the previous message. Therefore the system can respond in real-time as the maximum execution time is much less than the time between two incoming messages.

The second sub-question that concerns the predication accuracy:
How accurate is the motion estimation for liver tumor with the selected surrogate signal?

The motion estimation algorithm can predict the tumor's position with an RMS error of nearly 1.58 mm with a standard deviation of 1.54 mm in cranio-caudal direction and with an RMS error of 0.3 mm and with a standard deviation of 0.28 mm in the anterior-posterior direction. The error was mainly observed during the inhalation or expiration phase of the breathing cycle. The location predictions work best when there is no abrupt change in the breathing cycle. For practical application, the prediction and positioning system could be used to assist the surgeons in needle insertion when the patient holds breath.

6.2 Future work

Several recommendations could be made on the basis of findings of this study. There are still many limitations in the present real-time tumor-tracking and positioning system that limit the confidence in robot positioning. The largest potential uncertainty of the system is in the latency between communication channel of the OR PC and the robot PC. Other uncertainties includes

the detection of deformation of the tumor, change in breathing rate, etc. Some of the general recommendations are mentioned below, these should be studied and improved further:

- *System integration:* For continuing the research on the needle positioning system, several parts could be improved in the future. To start with, the experimental setup can be tested in the presence of an actual CT scanner with a real-time recording of data and processing it. There might be errors and delays associated with the system when integrated together that needs to be handled.
- *Respiratory motion phantom:* The motion phantom could be upgraded to generate breathing cycles with varying breathing rates and patterns. Breathing rate is the number of breathing cycles per minute, abnormalities might occur in rate, rhythm, and in the effort of breathing in patients. The current respiratory motion model might not give accurate predicted tumor location if the breathing rate is not constant.
- *User interface:* To improve the interaction with the system, a simple user interface could be created to locate and select the tumor to be operated. This could be helpful in the case of multiple tumors of similar shape and size.

Bibliography

Abayazid, Naghibi, Sirmaçek, and Berijanian. Toward patient-specific estimation of hepatic tumors respiratory motion: A finite element-based machine learning approach. 2019. URL https://www.cars2019.org/. 33rd International Conference on Computer Assisted Radiology and Surgery 2019, CARS 2019 ; Conference date: 17-06-2019 Through 21-06-2019.

Al-Shuka, Leonhardt, Zhu, Song, Ding, and Li. Active impedance control of bioinspired motion robotic manipulators: An overview. *Applied Bionics and Biomechanics*, 2018(iii), 2018. ISSN 17542103. doi: 10.1155/2018/8203054.

Barry. Liver biopsy, 2015. URL https://www.medindia.net/surgicalprocedures/liver-biopsy.htm.

Bell and Shetty. X-ray artifacts, 2020. URL https://radiopaedia.org/articles/x-ray-artifacts.

Beneduce, Castaldi, Marino, Tono, Gatta, Pontisso, and Fassina. Improvement of liver cancer detection with simultaneous assessment of circulating levels of free alpha-fetoprotein (afp) and afp-igm complexes. *The International journal of biological markers*, 19:155–9, 11 2003. doi: 10.1177/172460080401900211.

Berijanian, Abayazid, and Naghibi. Estimation of hepatic tumors respiratory motion using learning algorithms and surrogates M . (Maryam) Berijanian. Master's thesis, 2018.

Bihlmaier, Beyl, Nicolai, Kunze, Mintenbeck, Schreiter, Brennecke, Hutzl, Raczkowsky, and Wörn. Ros-based cognitive surgical robotics. 2016. doi: 10.1007/978-3-319-26054-9_12.

Blausen.com staff. Anatomical terms of location. *WikiJournal of Medicine*, 2014. doi: 10.15347/WJM/2014.010. URL https://en.wikiversity.org/wiki/WikiJournal_of_Medicine/Medical_gallery_of_Blausen_Medical_2014.

Bradski and Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.", 2008.

Clearpath Robotics. Intro to ros, 2020. URL http://www.clearpathrobotics.com/assets/guides/kinetic/ros/Intro%20to%20the%20Robot%20Operating%20System.html#general-concepts.

Clifford, Banovac, Levy, and Cleary. Assessment of hepatic motion secondary to respiration for computer assisted interventions. *Computer Aided Surgery*, 7(5):291–299, 2002. ISSN 10929088. doi: 10.1002/igs.10049.

Fahmi, Simonis, and Abayazid. Respiratory motion estimation of the liver with abdominal motion as a surrogate. 14(6):1–11, 2018. ISSN 1478596X. doi: 10.1002/rcs.1940.

Frank, Krieger, Leonard, Patel, and Tokuda. ROS-IGTL-Bridge: an open network interface for image-guided therapy using the ROS environment. *International Journal of Computer Assisted Radiology and Surgery*, 12(8):1451–1460, 2017. ISSN 18616429. doi: 10.1007/s11548-017-1618-1.

Franka Emika. Franka control interface documentation, 2020a. URL https://frankaemika.github.io/docs/requirements.html.

Franka Emika. Robot and interface specifications, 2020b. URL https://frankaemika.github.io/docs/control_parameters.html.

Hoffman. Picture of liver, 2014. URL https://www.webmd.com/digestive-disorders/picture-of-the-liver#1.

Hogan. Impedance Control: an Approach To Manipulation. *Proceedings of the American Control Conference*, 1(March):304–313, 1984. ISSN 07431619. doi: 10.23919/acc.1984.4788393.

Hoitzing, Abayazid, and Vuurberg. Respiratory Motion Modelling in the Liver: Tumour Motion Prediction for Robot-Assisted Needle Guidance. Master's thesis, 2019.

International Federation of Robotics Frankfurt IFR. Demystifying collaborative industrial robots, December 2018. URL https://ifr.org/downloads/papers/IFR_Demystifying_Collaborative_Robots.

S. Jacob. How serious is a liver biopsy?, 2020. URL https://www.medicinenet.com/how_serious_is_a_liver_biopsy/article.htm.

Kazanzidesf, Chen, Deguet, Fischer, Taylor, and Dimaio. An open-source research kit for the da Vinci® Surgical System. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 6434–6439, 2014. ISSN 10504729. doi: 10.1109/ICRA.2014.6907809.

Keall, Mageras, Balter, Emery, Forster, Jiang, Kapatoes, Low, Murphy, Murray, Ramsey, Van Herk, Vedam, Wong, and Yorke. The management of respiratory motion in radiation oncology report of AAPM Task Group 76. 33(10):3874–3900, 2006. ISSN 00942405. doi: 10.1118/1.2349696.

Kettenbach, Kronreif, Figl, Fürst, Birkfellner, Hanel, Ptacek, and Bergmann. Robot-assisted biopsy using computed tomography-guidance: Initial results from in vitro tests. *Investigative Radiology*, 40(4):219–228, 2005. ISSN 00209996. doi: 10.1097/01.rli.0000155285.05672.cf.

Klinder, Lorenz, and Ostermann. Free-breathing intra- and intersubject respiratory motion capturing, modeling, and prediction. *Medical Imaging 2009: Image Processing*, 7259:72590T, 2009. ISSN 16057422. doi: 10.1117/12.811990.

Leonard, Wu, Kim, Krieger, and Kim. Smart tissue anastomosis robot (STAR): A vision-guided robotics system for laparoscopic suturing. *IEEE Transactions on Biomedical Engineering*, 61 (4):1305–1317, 2014. ISSN 15582531. doi: 10.1109/TBME.2014.2302385.

Lobbes, Abayazid, Broeders, and Katwijk. Needle guidance technology for image-guided percutaneous procedures. Master's thesis, 2018.

Makary and Daniel. Medical error-the third leading cause of death in the US. *BMJ (Online)*, 353(May):1–5, 2016. ISSN 17561833. doi: 10.1136/bmj.i2139. URL http://dx.doi.org/doi:10.1136/bmj.i2139.

McClelland, Hawkes, Schaeffter, and King. Respiratory motion models: A review. 17(1):19–42, 2013. ISSN 13618415. doi: 10.1016/j.media.2012.09.005. URL http://dx.doi.org/10.1016/j.media.2012.09.005.

Miao, Zhao, Hong, Zhu, and Li. Doppler radar-based human breathing patterns classification using support vector machine. In *2017 IEEE Radar Conference (RadarConf)*, pages 0456–0459, 2017.

Naghibi, Costa, and Abayazid. A soft robotic phantom to simulate the dynamic respiratory motion of human liver. In *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob)*, pages 577–582, 2018. doi: 10.1109/BIOROB.2018.8488115.

Pedregosa, Varoquaux, Gramfort, Michel, Thirion, Grisel, Blondel, Prettenhofer, Weiss, Dubourg, Vanderplas, Passos, Cournapeau, Brucher, Perrot, and Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Philips. Augmented Reality Surgical Navigation technology, 2020. URL https://www.philips.com/a-w/research/research-programs/augmented-reality-surgical-navigation.html#_zeroxray.

Philips Healthcare. Making the difference with Philips Live Image Guidance Helping everyone benefit from AlluraClarity. 2014.

Reisner, King, Klein, Auner, and Pandya. A prototype biosensor-integrated image-guided surgery system Raman spectroscopy. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 3(1):82–88, 2007. doi: 10.1002/rcs.

H. Shirato, S. Shimizu, T. Kunieda, K. Kitamura, M. Van Herk, K. Kagei, T. Nishioka, S. Hashimoto, K. Fujita, H. Aoyama, K. Tsuchiya, K. Kudo, and K. Miyasaka. Physical aspects of a real-time tumor-tracking system for gated radiotherapy. *International Journal of Radiation Oncology Biology Physics*, 48(4):1187–1195, 2000. ISSN 03603016. doi: 10.1016/S0360-3016(00)00748-3.

Solomon, Bohlman, Kavoussi, and Stoianovici. Technical developments. *Foundry Management and Technology*, 129(12):18, 2001. ISSN 03608999. doi: 10.1080/10406020109355177.

The American Cancer Society. Ablation for liver cancer, 2020. URL https://www.cancer.org/cancer/liver-cancer/treating/tumor-ablation.html#:~:text=They%20are%20less%20likely%20to,a%20little%20over%20an%20inch).

The MathWorks INC. Ridge regression, 2018. URL http://nl.mathworks.com/help/stats/ridge.html.

Waelkens, V. Oosterom, V. den Berg, Navab, and Leeuwen. *Surgical Navigation: An Overview of the State-of-the-Art Clinical Applications*. 04 2016. ISBN 978-3-319-26049-5. doi: 10.1007/978-3-319-26051-8_4.

Wasza, Fischer, Leutheuser, Oefner, Bert, Maier, and Hornegger. Real-Time Respiratory Motion Analysis Using 4-D Shape Priors. *IEEE Transactions on Biomedical Engineering*, 63(3):485–495, 2016. ISSN 15582531. doi: 10.1109/TBME.2015.2463769.

webMD. Liver biopsy, 2020. URL https://www.webmd.com/digestive-disorders/digestive-diseases-liver-biopsy#1.

Welch, Sheedy, Johnson, Johnson, and Stephens. CT-guided biopsy: Prospective analysis of 1,000 procedures. *Radiology*, 171(2):493–496, 1989. ISSN 00338419. doi: 10.1148/radiology.171.2.2704815.

World Health Organisation. Europe cancer statistics, 2012. URL https://www.euro.who.int/en/health-topics/noncommunicable-diseases/cancer/data-and-statistics.