FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Ultrasound-Based Instrument Tracking: Application to Fetoscopic Endoluminal Tracheal Occlusion

Daniel Corona Oliveira Costa

WORKING VERSION

Mestrado em Bioengenharia - Especialização em Engenharia Biomédica

Supervisor: Prof. Dr. António Pedro Aguiar

Co-Supervisor: Dr. Gianni Borghesan

Co-Supervisor: Dr. Mouloud Ourak

June 16, 2023

Ultrasound-Based Instrument Tracking: Application to Fetoscopic Endoluminal Tracheal Occlusion

Daniel Corona Oliveira Costa

Mestrado em Bioengenharia - Especialização em Engenharia Biomédica

June 16, 2023

Abstract

Fetoscopic Endoluminal Tracheal Occlusion (FETO) is a minimally invasive surgery for treating severe cases of Congenital Diaphragmatic Hernia (CDH). CDH is a condition characterized by a defect in the diaphragm that affects about 0.8 to 5 per 10 000 fetuses. This defect causes the protrusion of the abdominal contents into the thoracic cavity, leading to pulmonary hypoplasia. During FETO, a fetoscope is positioned inside the fetus trachea under ultrasound guidance, then a latex balloon goes through the fetoscope working channel and is inflated to occlude the fetus trachea. The occlusion leads to the accumulation of lung fluid, causing lung stretch and reversing the pulmonary hypoplasia.

The ultrasound guidance is provided by a sonographer who manually operates an ultrasound probe. The sonographer is required to align the probe with the fetoscope while applying significant forces to obtain images of good quality. The cognitive and physical burden on the sonographer can be reduced by means of an ultrasound-based medical instrument tracking system that can automatically control the probe position based on the fetoscope position. This thesis presents the first research towards the development of this system for FETO.

Firstly, several instrument localization in 2D ultrasound images methods are developed based on the state-of-the-art. These methods provide the instrument tip location in the ultrasound images acquired by an ultrasound probe. Subsequently, their tracking performance is evaluated based on the ground-truth instrument tip position acquired by an optical tracking system. From the mentioned study we could conclude that the best algorithm for real-time application is a deep learning-based algorithm that achieved a maximum root mean square error between the ground-truth instrument tip and the algorithm estimation for the tip position of 6.97 mm. This instrument localization algorithm is then used to provide feedback information to control a robotic arm with 6 degrees of freedom that operates the probe position.

The result of this master's thesis is a real-time ultrasound-based instrument tracking framework for fetoscope tracking during FETO. The framework is based on a finite state machine which was implemented by using different software components. The system achieved a root mean square error of 8.61 mm between the instrument tip position and the probe position that was tracking the fetoscope tip. The system still needs some robustness analysis and experimentation on a more realistic phantom, nevertheless it provides a building block for the automation of the Fetoscopic Endoluminal Tracheal Occlusion surgery.

Keywords: minimally invasive surgery, ultrasound, tracking, deep learning

Resumo

A Oclusão Traqueal por Fetoscopia (FETO) é uma cirurgia minimamente invasiva para tratar casos graves de Hérnia Diafragmática Congénita (HDC). HDC é uma condição caracterizada por um defeito no diafragma que afeta cerca de 0.8 a 5 a cada 10 000 fetos. Esse defeito causa a protrusão dos conteúdos abdominais na cavidade torácica, resultando em hipoplasia pulmonar. Durante o procedimento FETO, um fetoscópio é posicionado no interior da traqueia sendo guiado por ultrassom e, em seguida, um balão de látex passa pelo canal de trabalho do fetoscópio e é inflado com o objetivo de obstruir a traqueia. A obstrução leva ao acúmulo de líquido nos pulmões, causando a extensão do pulmão e revertendo a hipoplasia pulmonar.

A orientação por ultrassom é fornecida por um sonografista que opera manualmente uma sonda de ultrassom. O sonografista deve alinhar a sonda com o fetoscópio, aplicando forças consideravelmente elevadas de forma a obter imagens de boa qualidade. O fardo cognitivo e físico sobre o sonografista pode ser reduzido por meio de um sistema de rastreamento de instrumento médico baseado em ultrassom, esse sistema pode controlar automaticamente a posição da sonda com base na posição do fetoscópio. A primeira investigação para o desenvolvimento desse sistema para FETO é analisada nesta tese.

Primeiramente, diversos métodos de localização de instrumento em imagens 2D de ultrassom são desenvolvidos com base no estado da arte. Esses métodos fornecem a localização da ponta do instrumento nas imagens de ultrassom adquiridas por uma sonda. Subsequentemente, o desempenho de rastreamento dos algoritmos é avaliado com base na posição real da ponta do instrumento adquirida por um sistema de rastreamento óptico. Do estudo mencionado podemos concluir que o melhor algoritmo para aplicação em tempo real é um algoritmo baseado em aprendizagem profunda, apresentando um erro quadrático médio de 6.97 mm entre a posição real e a estimativa da posição produzida pelo algoritmo. Em seguida, esse algoritmo de localização é utilizado para fornecer feedback a um braço robótico com 6 graus de liberdade que opera a posição da sonda.

O resultado desta tese de mestrado é um framework para rastreamento em tempo real de instrumento baseado em ultrassom para o rastreamento de um fetoscópio durante FETO. O framework tem por base uma máquina de estados finita, que foi implementada utilizando diversos componentes de software. O sistema apresentou um erro quadrático médio de 8.61 mm entre a posição da ponta do instrumento e a posição da sonda que estava rastreando a ponta do fetoscópio. O sistema ainda precisa passar por testes de robustez e experimentação em um modelo mais realista, no entanto, o mesmo fornece uma base para a automação da cirurgia de Oclusão Traqueal por Fetoscopia.

Keywords: cirurgia minimamente invasiva, ultrassom, rastreamento, aprendizagem profunda

Acknowledgements

The development of this work would not be possible without the help and insights of several people.

Firstly, I would like to express my gratitude to my supervisors Prof. António Pedro Aguiar, Dr. Gianni Borghesan, and Dr. Mouloud Ourak, who assisted me with the various tasks I had to accomplish during the development of this thesis.

I would also like to express my thanks to Prof. Emmanuel Vander Poorten, who provided me with the opportunity to work within the Robot Assisted Surgery lab at KU Leuven. Additionally, I want to express my appreciation to the PhD candidates Yuyu Cai and Li Ruixuan, who took their time to assist me with the use of the lab equipment.

A thank you for the friends and professors that I have met during my academic journey at the Federal University of Pernambuco, University of Coimbra, and University of Porto. I cherish the friendships and knowledge that these universities have provided me.

I extend my thanks to my friends from Brazil, particularly my dear friends from Campina Grande, PB. I am grateful for their support and I know that I can always count on them.

And last but most important, a heartfelt thank you goes to my family, dad, mom, brother, and sister. Without them I would not be here, nor would I be the person I am today.

Daniel Costa

Contents

List of Figures

List of Tables

Abbreviations

US	Ultrasound
MIFS	Minimally Invasive Fetoscopic Surgery
CDH	Congenital Diaphragmatic Hernia
FETO	Fetoscopic Endoluminal Tracheal Occlusion
PPHN	Persistent Pulmonary Hypertension of Newborn
LV	Left Ventricle
RV	Right Ventricle
MIS	Minimally Invasive Surgery
LHR	Lung-to-head Ratio
O/E LHR	Observed over Expected Lung-to-head Ratio
OTS	Optical Tracking Systems
EMTS	Electromagnetic Tracking Systems
RANSAC	Random Sample Consensus
PCA	Principal Component Analysis
ML	Machine Learning
SVM	Support Vector Machine
AI	Artificial Intelligence
MRI	Magnetic Resonance Imaging
CT	Computed Tomography
DL	Deep Learning
CNN	Convolutional Neural Network
ROS	Robot Operating System
GT	Ground Truth
ROI	Region of Interest
RANSAC	Random Sample Consensus
TE	Tip Error
DoF	Degree of Freedm
EU-Net	Enhanced U-Net
OEU-Net	Orientation Estimation U-Net
RMSE	Root Mean Square Error
PC	Principal Component
URDF	Unified Robot Description Format
eTaSL	expressiongraph-based Task Specification Language
Orocos	Open robot control software
rFSM	real-time Finite State Machine

Symbols

ϕ	Altitude angle of the fetoscope relative to the ultrasound image plane
θ	Azimtuh angle of the fetoscope relative to the ultrasound image plane
λ	Wavelength of sinusoidal factor in Gabor filter
Θ	Orientation of the Gabor filter kernel
ψ	Phase offset of sinusoidal factor in Gabor filter
γ	Spatial aspect ratio of the Gabor filter
σ	Standard deviation of the Gaussian envelope in Gabor filter
μ	Mean of the Gaussian distribution used to initialize the particles of a Particle filter
δ	Standard deviation of the Gaussian distribution used to initialize the particles of a Particle filter
η	State velocity measurement noise of Particle filter
ν	State measurement noise of Particle filter
ξ	Standard deviation of the probability density used to update the particle weights of a Particle filter

Chapter 1

Introduction

The medical technology industry is one of the largest growing industries in the world, and it has been greatly impacted by the digital technologies that are being used in Industry 4.0 (Silva et al., 2022). These technologies are advancing healthcare to unprecedented levels of comfort (Popov et al., 2022). Ultrasound (US) imaging has seen significant improvements throughout this fourth industrial revolution period, leading to an increased interest in improving ultrasound-based navigation systems and automated localization of medical instruments. Compared to other imaging techniques, US technology is advantageous due to its high temporal resolution, deep accessibility, as well as its safety, and low cost (Chen et al., 2019). Furthermore, ultrasound images are widely used for guidance during minimally invasive therapies (Zhao et al., 2020), such as in minimally invasive fetoscopic surgery (MIFS).

MIFS provides an appropriate scenario for introducing US-based technologies thanks to the amniotic fluid medium, which facilitates signal propagation and has no known harm to the fetus, unlike other imaging modalities (Yang et al., 2013). This master's thesis is a building block in the research and development for US-based tracking during MIFS, in particular, applied to the Fetoscopic Endoluminal Tracheal Occlusion (FETO) intervention for fetuses diagnosed with Congenital Diaphragmatic Hernia (CDH).

To provide a complete understanding of the research conducted in this master's thesis, some background information is required. This introductory chapter provides a general introduction to the Congenital Diaphragmatic Hernia condition, followed by a description of the Fetoscopic Endoluminal Tracheal Occlusion procedure, and a brief description of ultrasound imaging. Next, the problem which is addressed in this thesis is presented, succeeded by the aim of this master's thesis. Finally, an outline of the following chapters is provided.

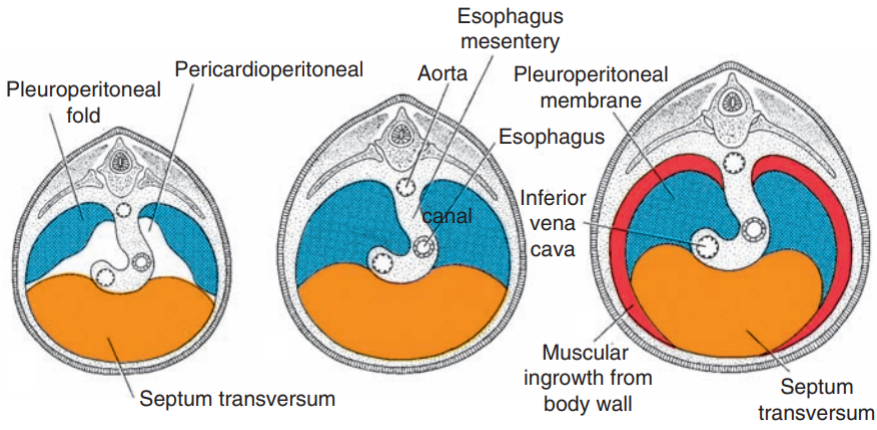


Figure 1.1: Normal development of the diaphragm. The pleuroperitoneal membranes fuses with the septum transversum, isolating the thoracic cavity from the abdominal one (Sadler, 2011).

1.1 Background

1.1.1 Congenital Diaphragmatic Hernia

Congenital Diaphragmatic Hernia (CDH) is a condition characterized by a defect in the diaphragm. It is most frequently caused by failure of one or both of the pleuroperitoneal membranes (see Fig. 1.1) to close the pericardioperitoneal canals (see Fig. 1.2) (Sadler, 2011). In that case, the peritoneal and pleural cavities are continuous, which may lead to a protrusion of abdominal contents into the thoracic cavity (see Fig. 1.3).

The hernias in the diaphragm can be characterized by location or by size. Regarding location, the most common type is the postero-lateral (Bochdalek) hernias with the majority occurring on the left side (85%). The other types are anterior (Morgagni) and central hernias. Whilst the size of the defect may vary between small to diaphragmatic agenesis (Chandrasekharan et al., 2017).

The incidence of CDH ranges from approximately 0.8 to 5 per 10 000 births. The mortality rate of CDH is extremely variable, depending on the institution where CDH is treated, and also on the medical conditions of the fetus, such as incidence of other physiological or anatomical anomalies (Tsao and Lally, 2008).

The detection of CDH can be performed between 4 and 12 weeks after gestation, which is the time period when the diaphragm is being developed. Medical imaging techniques are used to diagnose and to make a prognosis of CDH. Approximately 40% of CDH cases have associated anomalies, and the nature of these anomalies have a huge impact in the prognosis (Deprest et al., 2015). Genetic testing may also be used to augment the information used for the prognosis.

The management of CDH can be performed antenatal or postnatal by means of surgical procedures, fluid/blood management, extra corporeal membrane oxygenation, inhalation of nitric oxide, pharmacologic treatment and/or ventilator support (Gupta and Harting, 2020).

The etiology of CDH remains unclear and it is thought to be multifactorial. CDH can be associated with cardiac, gastrointestinal, genitourinary anomalies and the majority of the cases have

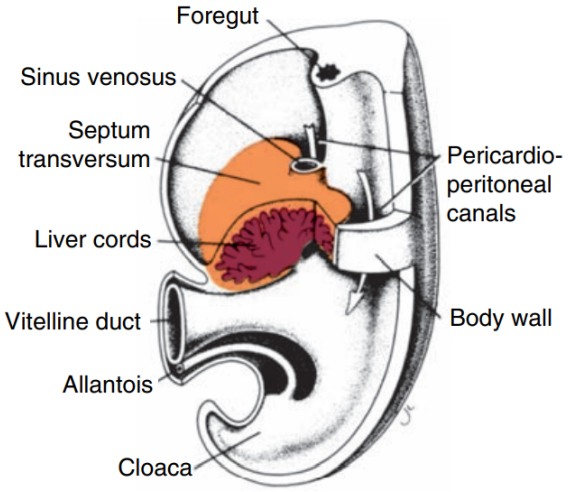


Figure 1.2: Drawing showing the ventral view and the pericardioperitoneal canals of an embryo of 24 days of gestation (Sadler, 2011).

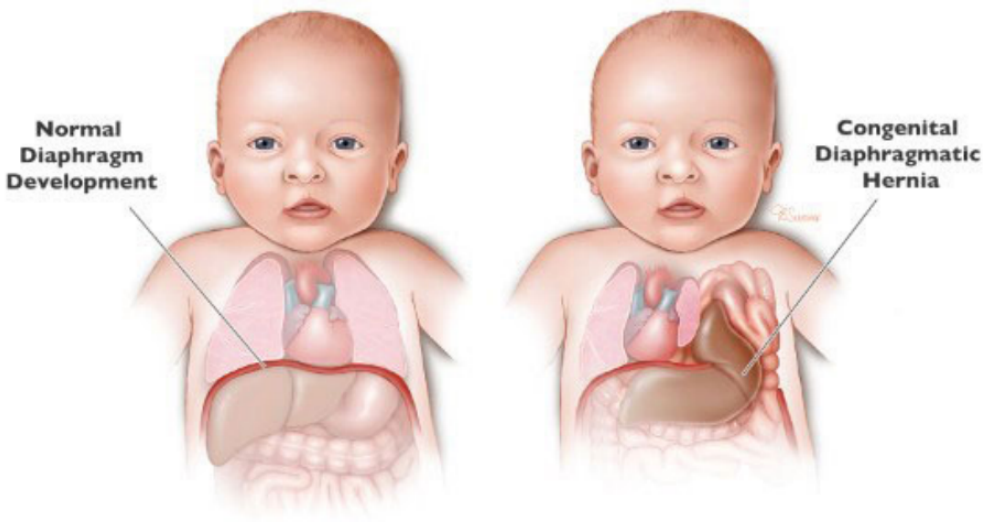


Figure 1.3: Comparison between diaphragm development: (left) normal development, (right) congenital diaphragmatic hernia (Texas Children's Fetal Center, 2023).

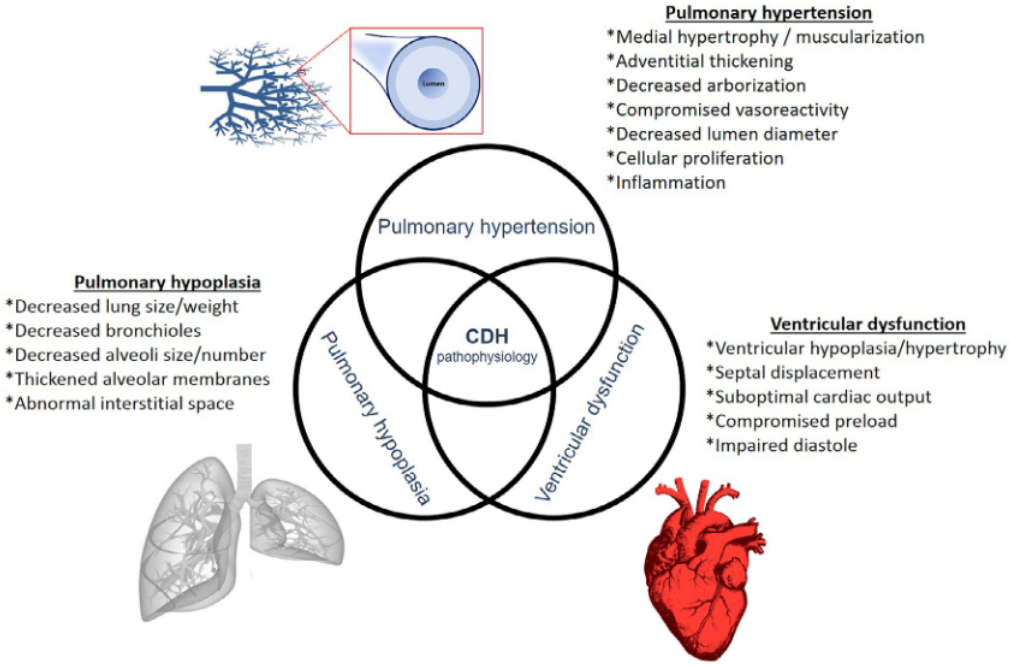


Figure 1.4: The three cornerstones of CDH pathophysiology (Gupta and Harting, 2020)

an isolated diaphragmatic defect presenting with pulmonary hypoplasia and persistent pulmonary hypertension of newborn (PPHN), which are the leading causes of death (Deprest et al., 2015).

It is thought that CDH causes pulmonary hypoplasia due to an ipsilateral compression of the lung by the abdominal contents. Furthermore, the total pulmonary vascular bed of newborns with CDH has a decreased number of vessels, and there is a pulmonary vascular remodeling and extension of the pulmonary muscle layer into small arterioles. This remodeling and the paucity of pulmonary vasculature contribute to the irreversible consequences of PPHN. The contribution to the reversible consequences mainly comes from the altered vasoreactivity caused by PPHN. CDH also causes cardiac dysfunctions, being related to left ventricular (LV) hypoplasia, reduced LV output, diastolic dysfunctions, RV increased afterload, RV hypertrophy, and septal displacement, which are common conditions in left-sided CDH (Patel and Kipfmueller, 2017). Figure 1.4 presents a summary of the most common characteristic of the CDH pathophysiology.

1.1.2 Fetoscopic Endoluminal Tracheal Occlusion

Fetoscopic endoluminal tracheal occlusion (FETO) is a minimally invasive surgery process where a latex balloon is endoscopically positioned above the carina and inflated to occlude the trachea (see Fig. 1.5). This occlusion leads to the accumulation of lung fluid which causes lung stretch, reversing the pulmonary hypoplasia of a fetus with severe CDH. The lung stretch activates a pathway that stimulates the proliferation and growth of the airways and pulmonary vessels (der Veeken et al., 2018).

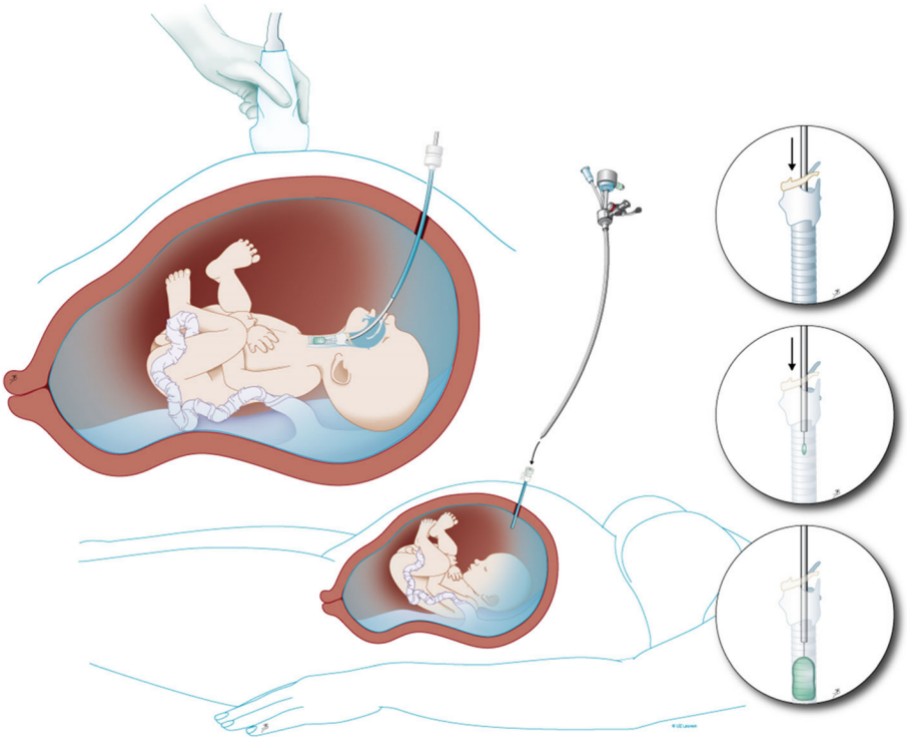


Figure 1.5: Schematic drawing of the fetoscopic endoluminal tracheal occlusion surgery (der Veeken et al., 2018).

The reversal of occlusion is an important component in the fetal treatment strategy. By reversing the occlusion before birth, the balance between type I and type II pneumocytes at birth is more optimal than without the reversing procedure (der Veeken et al., 2018).

The prediction of the FETO outcome is an important measure to determine if the surgical procedure is going to be performed. Nowadays, lung-to-head-ratio (LHR) is the best parameter for prediction. A study to assess the relationship between the observed over expected lung-to-head-ratio (O/E LHR) was performed with 354 fetuses (Deprest et al., 2011), the results can be seen in Figure 1.6. Parameters obtained from magnetic resonance imaging may also contribute to obtaining a prediction of the outcome. These parameters are the volume of both lungs, quantification of the degree of liver herniation, and stomach position (der Veeken et al., 2018).

FETO is typically performed at 27 to 32 weeks of gestation. The patient is positioned in a dorsal supine position such that there is direct access to the fetal mouth. Ultrasound imaging is used to determine the position of the mouth and some external manipulations may be performed in order to obtain an optimal position of the fetus (der Veeken et al., 2018).

Both the mother and the fetus are anaesthetized before the surgery. The mother's womb is sterilized and a skin incision is made for the insertion of a flexible cannula loaded with a pyramidal trocar into the amniotic cavity. The insertion is made in an area devoid of placenta and as perpendicular as possible to the nose tip of the fetus. Then, the endoscope is guided through the cannula by ultrasound imaging. The occlusion of the trachea is done by a detachable inflatable

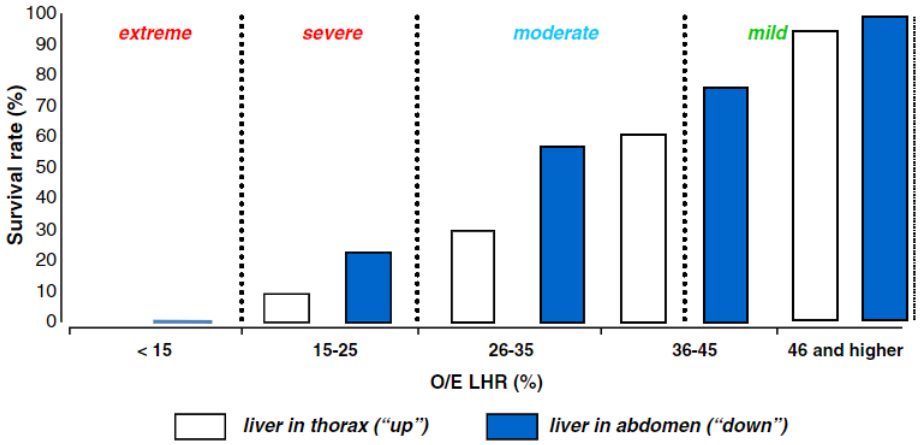


Figure 1.6: Survival rates of fetuses with isolated left-sided CDH, depending on the measurement of the O/E LHR and position of the liver as in the Antenatal CDH Registry (Deprest et al., 2011).

latex balloon (see Fig. 1.7) which is delivered by a catheter. Once the balloon is in the correct position, it is filled via a luer-lock syringe with an integrated one way valve. Landmarks for the balloon insertion are progressively the tip of the nose, philtrum, tongue, the raphe of the palate, uvula, epiglottis, vocal cords and the carina or the tracheal rings (see Fig. 1.8). After the balloon is inflated and detached, the excessive amniotic fluid is drained until a normal volume is achieved. The duration of the FETO procedure ranges from 3 to 93 minutes (der Veeken et al., 2018).

The tracheoscopic sheath (see Fig. 1.7) used to insert the fetoscope has an outer diameter of 3.3 mm, a working length equal to 30.6 cm, it is precurved by 30º, and the tip of the sheat is sandblasted for increased echogenicity (Deprest et al., 2011).

Patients are followed with ultrasound until the reversal of the occlusion. This reversal procedure corresponds to the balloon removal and it is an important step since it triggers lung maturation, reduces morbidity and increases survival chances. The removal can be performed by means of a fetoscopic in utero procedure, puncture of the balloon under ultrasound guidance, or tracheoscopic removal with the baby on placental circulation. In a worst scenario, the removal can be performed with postnatal puncture or tracheoscopy (der Veeken et al., 2018).

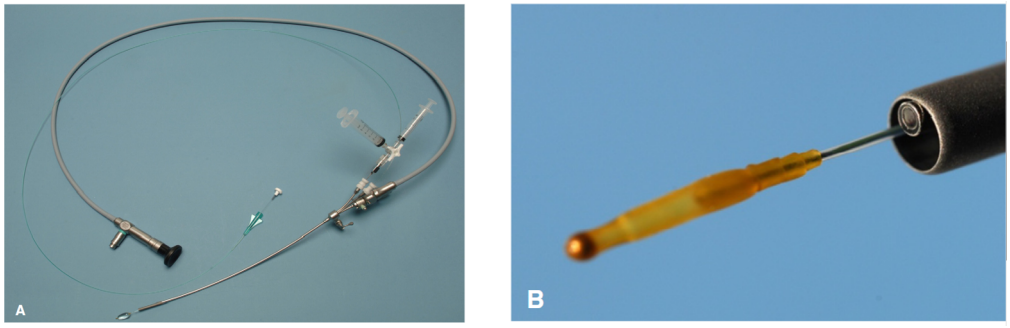


Figure 1.7: (A) Curved sheath loaded with fiber optic fetoscope with deported eye piece. (B) Catheter with deflated latex balloon (Deprest et al., 2011).

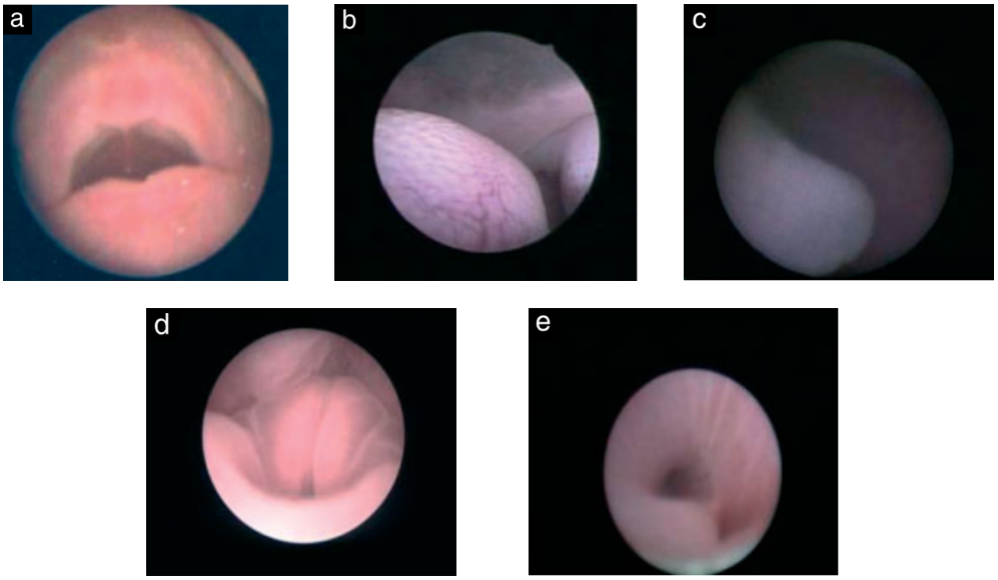


Figure 1.8: Some of the landmarks for the balloon insertion during FETO: raphe of the palate (a), tongue (b), uvula and epiglottis (c), vocal cords (d), and the carine (e) (Deprest et al., 2004).

1.1.3 Ultrasound Imaging

Ultrasound corresponds to acoustic waves with a frequency higher than 20 kHz. Typical medical ultrasound frequencies are between 2 MHz to 40 MHz (Maier et al., 2018). Since US is a wave, it can go under some physical processes such as reflection, refraction, diffraction and scattering. The US probe contains a transducer which converts mechanical energy to electrical energy and vice versa, thus it acts as both the generator and sensor of the acoustic waves. When the probe starts generating ultrasound waves, it also receives reflected acoustic signals which are converted into pixel intensities in an US image. The position which the acoustic signal is presented in the image depends on the time between emission and reflection of the wave, and also on the probe location that received the signal.

1.1.3.1 Spatial resolution

Usually, a distinction is made between two types of spatial resolution in US images, these are the axial and the lateral resolutions. Axial resolution concerns structures lying behind each other relative to the transducer emission direction, while lateral resolution concerns the distinguishability between structures lying next to each other relative to waves emission direction (Maier et al., 2018).

1.1.3.2 Imaging modes

US imaging presents different modes, the most common ones are the A-Mode, B-Mode, and M-Mode. A-Mode, or amplitude mode, is the simplest one in which the height of the reflected ultrasound is displayed over the sonic runtime in the axial direction (Maier et al., 2018). B-Mode,

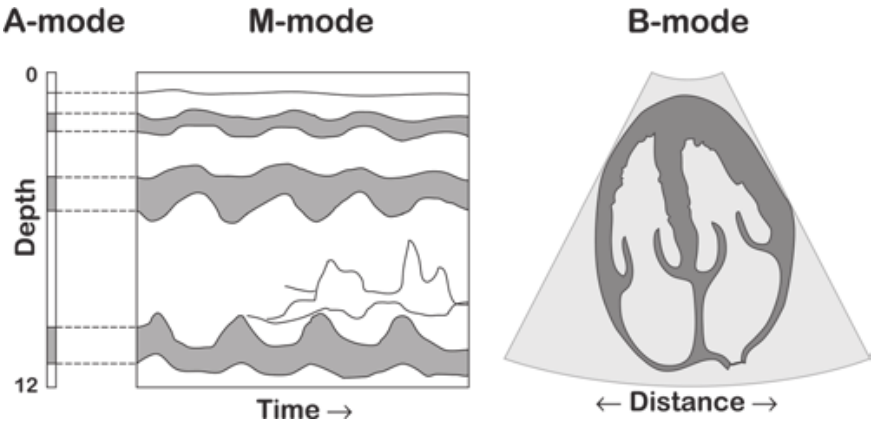


Figure 1.9: Three different US imaging modes: A-Mode, B-Mode, and M-Mode (Conrad, 2010).

or brightness mode, images are generated by combining a multitude of A-Mode scans in different directions into a 2D image (Maier et al., 2018). Moreover, if the A-Mode scans are generated in two dimensions, the resulting signals can be combined to produce a 3D volume. In M-Mode, or motion mode, successive ultrasonic pulses are emitted from the probe, while either an A-Mode or a B-Mode image is acquired each time, allowing a time-dependent measurement of biological structures movement (Maier et al., 2018). The work described in this dissertation focus on the analysis and processing of 2D B-Mode images. A depiction of the different modes is shown in Figure 1.9.

1.1.3.3 Echogenicity

In medical US images, it is important to have knowledge about the concept of echogenicity, which corresponds to the ability of a tissue or organ to reflect the US waves. Hyperechoic structures have higher echogenicity and are presented as lighter objects in the US image, while hypoechoic structures have lower echogenicity and are presented as greyish objects. There are also anechoic structures that do not present any reflection capabilities and thus, are presented as black pixels in the image.

1.2 Problem definition, objectives, and requirements

One of the FETO procedure limitations is the poor quantity of visual information that the surgeon have. The fetoscope visibility is compromised by turbid amniotic fluid, low resolution, and limited field of view of the surgery working space (Gruijthuijsen et al., 2018). Thus, the use of ultrasound scanning augments the real time awareness of the surgeon regarding the different structures that are present in the amniotic sac during the procedure, such as the umbilical cord, the placenta, and the fetoscope. If the fetoscope makes contact with any biological tissue, there is a risk of puncturing the tissue, which could lead to bleeding and termination of the intervention. Hence, the visual information gathered from the 2D US images contributes to a faster and safer FETO. Nevertheless, there are still some difficulties in the positioning of the US probe and the tracking

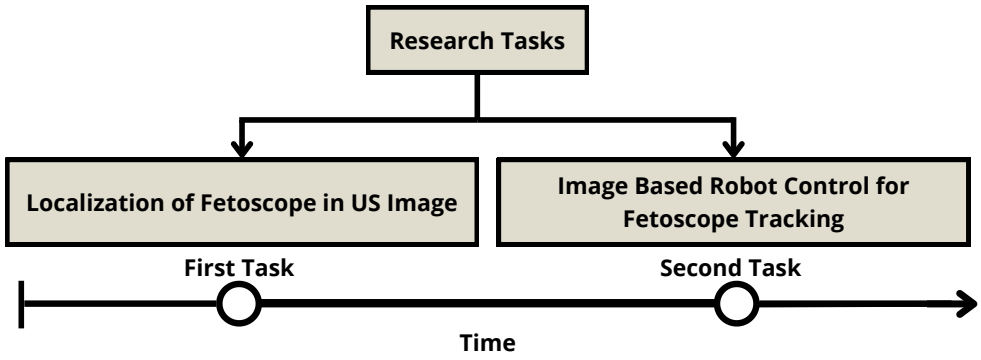


Figure 1.10: Research tasks and development timeline of this master's thesis.

of the fetoscope throughout the procedure. The sonographer that is controlling the probe must correctly align the US image plane with the medical instrument inside the mother's womb. During this procedure, the sonographer has to operate the probe through various angles and may need to apply significant force to obtain images with good quality. Furthermore, even a slight movement between the transducer and the fetoscope may lead to the instrument disappearing in the US image.

The objective of this work is the development of a real-time ultrasound-based instrument tracking framework of the fetoscope during Fetoscopic Endoluminal Tracheal Occlusion. An US probe is positioned by means of an autonomous robotic ultrasound imaging system, which corresponds to a 6 degrees-of-freedom robotic arm with the probe as an end-effector. With improved accuracy and dexterity, modern robot manipulators are capable of performing precise force and position control of the probe (Li et al., 2021a). Hence, contributes to a reduced physical and cognitive burden on the sonographer (Li et al., 2021a), while reducing operation time, which lowers the risk of preterm prelabour rupture of membranes (Jani et al., 2009).

In order to fulfill this objective, two tasks must be accomplished, which are the main contributions of this thesis. The first task focuses on the development of a localization algorithm which is able to localize a medical instrument in US images, while the second task concerns the development of an image-based control algorithm for autonomous robotic ultrasound imaging tracking. The control objective in the second task is to position the ultrasound probe in order to track the medical instrument while using the information given by the algorithm developed in the first task. These tasks are summarized in Figure 1.10.

The requirement for the ultrasound-based instrument tracking is to have the instrument occupying the largest possible area in the US image. Thus, it is necessary to maintain the instrument visible in the US images as long as possible, while the US probe must have a certain orientation that maximizes the instrument area in the image.

1.3 Thesis outline

The present dissertation is organized as follows: Chapter 2 contains an overview of the state of the art on localization in ultrasound images, and on autonomous robotic ultrasound imaging

tracking. Next, Chapter 3 presents a comparative study of the different instrument localization algorithms which were developed to localize and track the fetoscope in US images. In Chapter 4, the ultrasound-based instrument tracking framework is explained and evaluated. Finally, the conclusions and future directions of related research are presented in Chapter 5.

Chapter 2

State of the Art

As stated in Section 1.2, this dissertation explores two different tasks in order to achieve its objective. This chapter presents the state of the art on these two tasks, the first section focus on describing the different types of algorithms that are used for localizing and tracking instruments in ultrasound images, while comparing their performance, strengths, and weaknesses in the context of FETO. Next section steers the focus to the state of the art on autonomous robotic ultrasound imaging tracking.

2.1 State of the art on instrument localization in ultrasound images

Robot-assisted surgery can provide different benefits to minimally invasive surgery (MIS), such as a certain level of automation, increased dexterity, reduced tremors while manipulating instruments, and image guidance. Nowadays, MIS is mainly focused on image-guided procedures (Sorriento et al., 2020), where a tracking system delivers the position and orientation of a target relative to a reference point. Tracking systems are able to localize the target by means of three main hardware components: i) one or more sources to generate a signal, ii) one or more receivers to capture the signal, and iii) a data acquisition and signal processing system (Sorriento et al., 2020).

Two different strategies are commonly used for instrument localization: sensor-based and image-based strategies. Sensor-based localization corresponds to the use of external or internal sensor devices, such as optical tracking systems (OTS), and electromagnetic tracking systems (EMTS). These systems are the two main technologies integrated into commercially available surgical navigators (Sorriento et al., 2020). Nevertheless, OTS and EMTS have limitations that impair their use for localizing the fetoscope during FETO. The main drawback of OTS is the requirement of a direct and clear line-of-sight between the source and the receiver of the signal used to track the target. This line-of-sight is difficult to be maintained in a surgical scenario, where multiple instruments and medical doctors are occupying and moving around the patient. This drawback may be diminished by adding more sources or receivers, however, there is an increase

in the system's overall cost (Sorriento et al., 2020). Regarding the EMTS, the tracking accuracy of these systems is reduced by the interference produced by metal and magnetic materials which are commonly present in surgical rooms. Furthermore, it is not possible to integrate electromagnetic sensors into the anatomic structures present in the womb (e.g. umbilical cord), which must be taken into consideration in order to avoid tissue damage due to contact between the fetoscope and these structures.

Image-based approaches focus on localizing the instrument in a medical image, which may come from MRI, CT scans, X-Rays or US. Compared to US imaging, the other imaging techniques have high equipment costs. Furthermore, the involved sensors and imaging equipment may complicate the system setup in the operation room (Yang et al., 2022), which does not happen when using an US probe. Therefore, the development of an US image-based localization strategy is advantageous.

There are two main US formats being used in ultrasound-guided operations, 2D images and 3D volumes (Yang et al., 2022). Although 3D volumes provide a more complete spatial view of the intervention workspace than 2D images, it contains a large number of voxels, which may compromise the real-time localization of instruments due to the time necessary to gather the voxels data from the US probe and to compute an instrument localization algorithm. On the other hand, 2D images have a lower amount of data to be processed, are cheaper, and are more accessible in hospitals than 3D US machines. Zhao *et. al* provides a review on algorithms for biopsy needle localization in both two and three dimensional US images (Zhao et al., 2015), (Zhao et al., 2017). The slowest algorithm in 2D images took 15.7 ms to localize the neddle in an image, while the fastest algorithm for 3D US took an average of 84 ± 8 ms to perform the localization on a single volume. Thus, the 2D format is a better choice when trying to achieve real-time performance for instrument localization in ultrasound images. Moreover, it is the format used in this work.

The existing image-based localization algorithms may be divided into two classes, the non-machine-learning, and the machine-learning (ML) ones. These two classes can be even further segmented: non-machine learning methods include physical space and projection space strategies, while machine-learning methods contain the handcrafted features classifiers and deep learning methodologies (Yang et al., 2022). Figure 2.1 shows a complete view of the different methods.

The following sections provide a description of different image-based methods (see Fig. 2.1), that were developed specifically to detect medical instruments in US images. The majority of these algorithms follow a common pipeline (see Fig. 2.2) which starts with the pre-processing of the raw medical image. Next, a localization algorithm is applied to find the instrument by, e.g., defining a bounding box around the instrument, defining the instrument's axis, or performing a semantic segmentation. Finally, the post-processing is applied to the output of the previous step in order to provide a more refined instrument location.

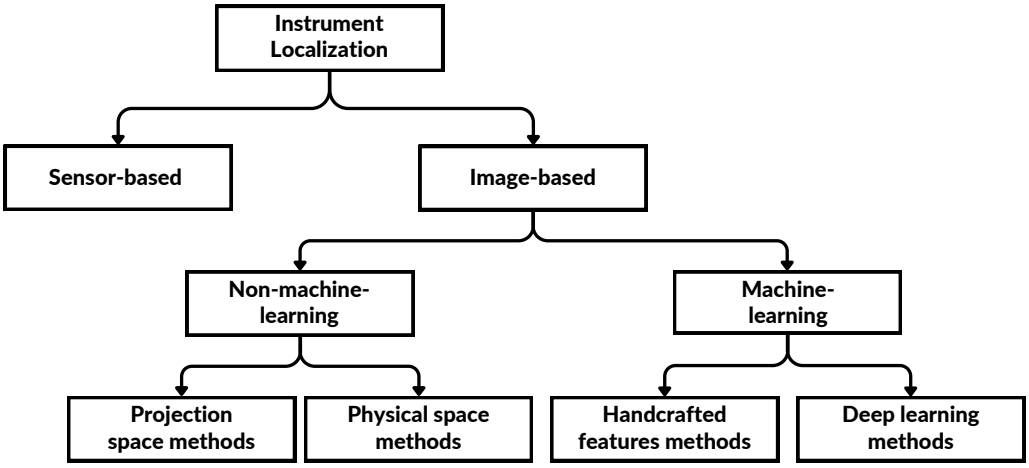


Figure 2.1: Tree diagram with the different types of instrument localization methods (Yang et al., 2022).

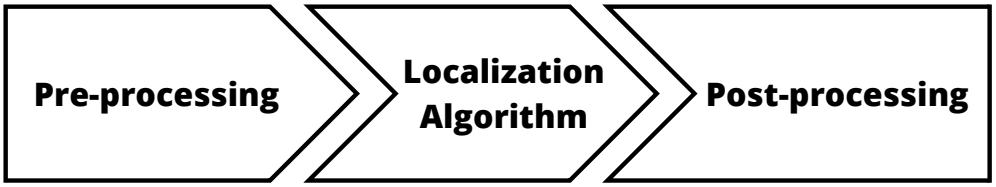


Figure 2.2: Pipeline for image-based instrument localization (Yang et al., 2022).

2.1.1 Non-Machine learning methods

2.1.1.1 Physical space methods

Physical space methods focus on mathematical modeling of the instrument geometry in a straightforward manner, making use of the standard spatial coordinate system (Yang et al., 2022). Generally, these methods start with the application of carefully designed filters or templates to enhance the instrument in the ultrasound image. Then, a segmentation operation takes place, usually, thresholding is performed. Finally, a model fitting is applied to localize the instrument (Yang et al., 2022). One example of this methodology is the work done in Kaya et al. (2015), where an optimized Gabor Filter is used to enhance the shape of a straight biopsy needle considering an estimation of the needle insertion angle, then a random sample consensus (RANSAC) line estimator is used to localize the needle axis, the post-processing consists in applying the same steps performed in the pre-processing, and then use a probability mapping to estimate the instrument tip position. Another example is given by Yang et al. (2013), that uses principal component analysis (PCA) after the pre-processing step to find the orientation of a fetoscope during fetoscopic surgery by modeling the instrument as a straight and connected voxel cluster, the first principal component axis corresponds to the fetoscope shaft direction estimation. Cao et al. (2013) proposes a template matching with a pre-defined catheter filter for coarse segmentation of the instrument.

Physical space methods are limited by prior knowledge of the instrument's shape or orientation. Consequently, some of these methods are semi-automatic, meaning they require some user-input data in order to start the localization and tracking of the medical instrument. Further, they are sensitive to image modality, which impacts the performance of these algorithms when going from an *in-vitro* to an *in-vivo* assessment or when dealing with dynamic backgrounds (Yang et al., 2022). Ultrasound images from FETO operations contain highly dynamic backgrounds, and prior knowledge of the instrument may only be possible if the fetoscope is rigid. Hence, physical space algorithms may have low accuracy for the fetoscope tracking task.

2.1.1.2 Projection space methods

Projection space methods are characterized by the application of a transformation from the physical space at the image coordinate system to a projection space. The transformation is based on prior knowledge of the instrument geometry, so the instrument yields a strong response after the projection. This strong response is translated into high intensity pixels (Yang et al., 2022). Daoud et al. (2017) uses a Radon transform to identify the instrument trajectory in an image produced by features from ultrasound images. In Alsbeih et al. (2020) work, a Radon transform is also performed in order to localize the instrument axis, then a template matching algorithm based on Normalize Cross-Correlation is responsible for tracking its tip. Ding and Fenster (2003) transforms the ultrasound image to a parametric space using the Hough transform in order to localize a biopsy needle, which is modeled as a 2D line.

These non-machine learning methods are usually validated based on *in-vitro* experiments, which underestimate the noise present in *in-vivo* ultrasound images. Since these methods focus on

enhancing the instrument based on the physical structure of the instrument, they lack contextual information, hampering their localization performance (Yang et al., 2022).

2.1.2 Machine learning methods

Machine learning is a branch of artificial intelligence (AI) and consists in statistic-based methods that can learn from data and take decisions or provide predictions. In the computer science field, machine learning presented huge developments in computer vision, robot control, product recognition, and speech recognition (Chalabi et al., 2021). An application example of machine learning is the trending ChatGPT application developed by *OpenAI* (OpenAI, 2023), which has extraordinary semantic, contextual, and speech recognition performance, being able to maintain coherence in long chat conversations.

In recent years, the idea of using machine learning for medical image analysis has gained traction (Kora et al., 2022). ML is able to cope with bigger datasets while reducing intra, as well as inter-operator variability (Kora et al., 2022), and even takes advantage of learning with more data, as it may learn more generalized features. On the other hand, traditional analysis is a time consuming task, especially with higher data volume, and usually presents high intra and inter-operator variability. ML is being used in a variety of medical image analysis tasks such as segmentation, disease categorization, severity grading, and object localization (Kora et al., 2022).

2.1.2.1 Handcrafted features methods

The handcrafted features methods employ feature vector extraction and task classification using machine-learning models. These two steps are usually applied at the pixel level to perform the instrument segmentation (Yang et al., 2022). The pipeline for training and testing, or applying the handcrafted feature ML models is presented in Figure 2.3, the first step consists in dividing the US images dataset into a training set and a testing set, an ML model should not be tested using the data used in the training phase to prevent model overfitting. The training phase consists in the learning of the ML classification model parameters, models examples are the support vector machine (SVM), logistic regression classifier, Fisher's linear discriminant, and K-Nearest Neighbors. During the testing, the already trained model receives the feature vectors extracted from the testing set and performs the pixel-level classification, producing an image with the segmented instrument.

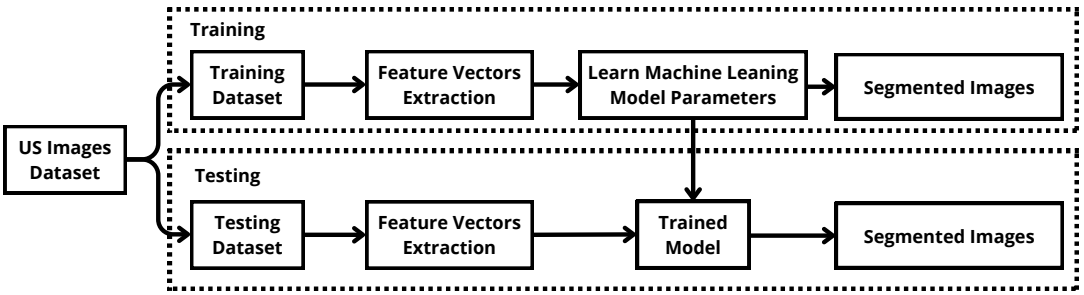


Figure 2.3: Pipeline for machine learning models training and testing using handcrafted features.

Several features have been used to build the feature vector, Beigi et al. (2017) work uses temporal based features extracted from optical flow computation and classifies these features using a SVM model, the output was a segmented instrument location prediction. Mwikirize et al. (2017) uses log-Gabor features that are processed in a histogram of oriented gradients descriptor. These descriptors are then classified by a SVM model which produces the instrument segmentation in the ultrasound images.

The design of handcrafted features is a time consuming task and requires task-related knowledge and experience. In addition, due to the difficulty in designing optimal features, post-processing requires complex techniques to filter out outliers and false positives. Hence, there is an increase in the development of deep learning methods, that are able to automatically learn task-related information and may learn features that convey local and contextual information.

2.1.2.2 Deep learning methods

Deep learning (DL) is a subfield of machine learning which comprises neural network architectures with multiple layers. Convolutional neural networks (CNNs) are a class of architectures which perform convolutions in the input data in order to extract features, allowing better learning of spatial information when compared to traditional artificial neural networks, such as the multilayer perceptron. Nowadays, CNNs are one of the most commonly used DL architectures for medical image analysis. CNNs have the ability to learn deep features from medical images, and, with further processing of these features, different image analysis tasks may be performed including segmentation, detection, or classification (Wang et al., 2021).

CNNs have several advantages, namely, wide application range, fast processing speed, high accuracy, and do not require feature engineering, like the previously described handcrafted feature methods. The CNNs learn the most important features from the given training set (Kora et al., 2022), thus there is no need for feature selection. However, this increases the model's computational complexity and creates the requirement for large training datasets (Kora et al., 2022). The training step in DL is typically slow and relies on labeled data. The gathering of labeled medical images, e.g. US images, can only be done by professional doctors, are generally difficult to collect, as well as expensive and rare, imposing a limitation for the application of DL techniques for medical image processing (Wang et al., 2021). Another drawback in machine learning techniques is the model interpretability, which is difficult to assess since the features are less understandable (Yang et al., 2022).

To overcome the obstacles posed by the scarcity of labeled medical images, several researches have been introducing the transfer learning technique. The basic idea of transfer learning is to take advantage of a pre-trained network, which was trained with the objective of realizing a certain task, and then fine-tune this network by exposing it to the scarce labeled medical data in order to accomplish a similar task. After fine-tuning, the network not only has stronger classification performance, but also, better feature extraction capabilities (Wang et al., 2021).

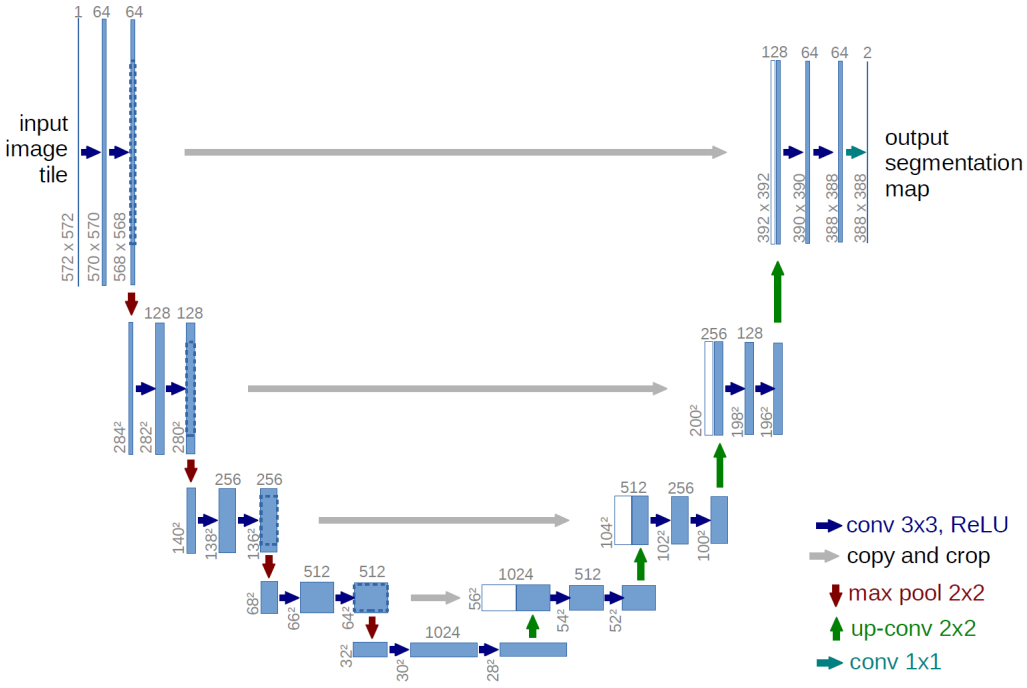


Figure 2.4: U-Net architecture (Ronneberger et al., 2015).

One example of a CNN architecture is the U-Net (see Fig. 2.4), which is normally used as the benchmark for medical image segmentation. U-Net follows an encoder-decoder approach. The encoding is performed by convolutions followed by max pooling operations, while the decoding corresponds to convolutions followed by transposed convolutions. There are also skip connections operations, which directly copy encoder features into the decoder path, thus, enabling the maintenance of low-level features. The semantic segmentation output corresponds to a pixel-wise classification where each pixel is assigned to a specific class, such as background or foreground.

An example of a deep learning method used for instrument localization is given by the work of Chen et al. (2022), where a novel deep CNN, named W-Net, is proposed (see Fig 2.5). The objective of the W-Net is to perform moving needle segmentation by extracting features of two adjacent US frames. When there are reduced needle visibility conditions or hyperechoic tissue around the needle, some DL methods have poor segmentation performance, thus the use of two adjacent frames allows W-Net to learn motion related features which may improve the segmentation output. Mwikirize et al. (2021) uses another approach, where a time-aware network consisting of convolutional layers followed by LSTM (long short term memory) modules is used to localize the medical instrument tip in a 2D ultrasound video. The input to this network is a consecutive sequence of 5 fused images, each fused image contains an enhanced tip image and an US frame. The enhanced tip image is simply the frame subtraction between two consecutive frames.

Recent developments in deep neural networks resulted in advanced approaches for 2D instrument detection, that not only localize the instrument but also provide an estimation for the instrument pose. Du et al. (2018) developed a neural network architecture that is divided into two branches, one of them is a detection network that has a similar structure to the U-Net, and

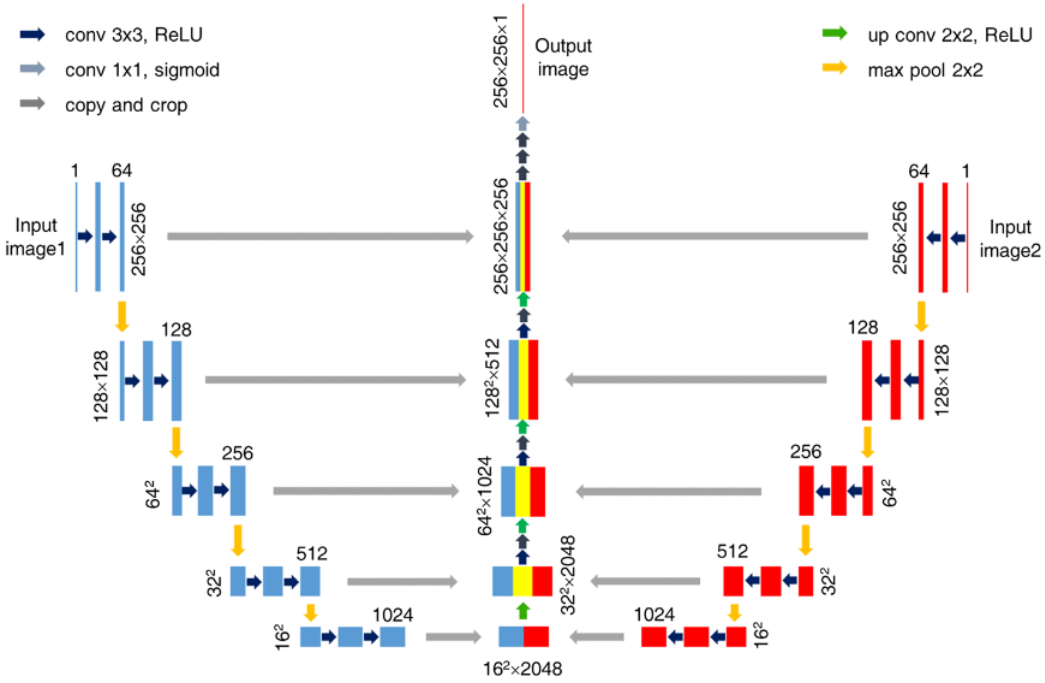


Figure 2.5: W-Net architecture (Chen et al., 2022).

the other branch is a regression network that gives a pose estimation for articulated instruments in minimally invasive surgeries. The pipeline for the pose estimation framework proposed in Du et al. (2018) is presented in Figure 2.6. Other works, such as the one developed by Hasan et al. (2021), follow a similar strategy where the neural network has more than one branch, each branch is responsible for one task, being one of the tasks the instrument segmentation or detection, and the other task the pose estimation. These type of neural networks are denominated multi-task since they are designed in such a way that they can accomplish more than one task at the same time.

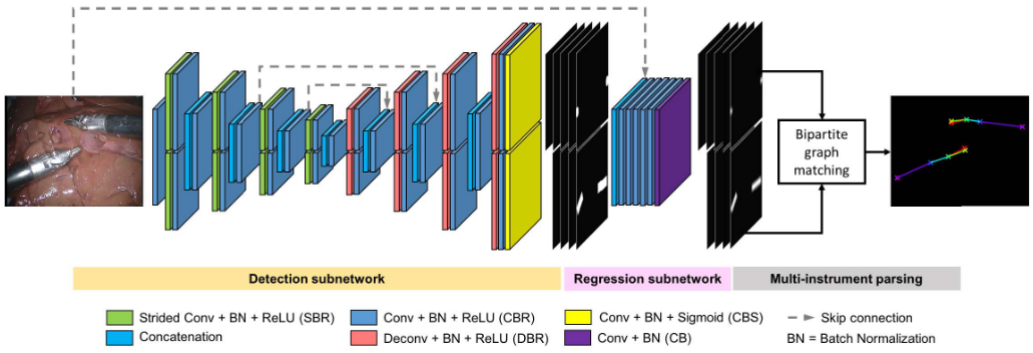


Figure 2.6: Pipeline for pose estimation framework developed in (Du et al., 2018).

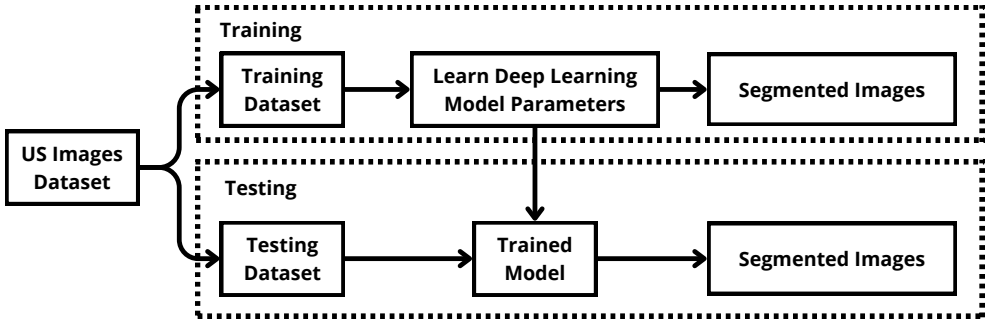


Figure 2.7: Pipeline for deep learning models training and testing.

The common framework used to train and apply DL methods for instrument detection is presented in Figure 2.7.

These machine learning methodologies present several advantages for the medical instrument localization task. Opposite to non-machine learning techniques, there is no necessary prior knowledge regarding instrument geometry. Moreover, deep learning networks can automatically learn powerful and general features that are used to segment and localize the instrument in the US images, despite the presence of noise.

2.1.3 Summary and discussion

The ultrasound-based instrument localization algorithms have been divided into four different methodologies. These algorithms have different advantages and disadvantages and some of them are more suitable for localizing a fetoscope during FETO, where the instrument has several degrees of freedom. Table 2.1 shows an overview of the properties of the different methodologies.

As of the present date and based on the state-of-the-art presented in this chapter, there is no algorithm developed for fetoscope localization in 2D ultrasound images during FETO. Thus, this thesis explores the use of some state-of-the-art methodologies for localizing the fetoscope.

The physical and projection space methodologies have the advantage of not requiring the training of a model or labeled data. Although it is necessary to have prior knowledge about the instrument geometry, these methods can be developed with simple image processing techniques while presenting fast image processing, which is advantageous for real-time instrument tracking. Since the physical and projection space methodologies have similar properties, a single algorithm following one of these methods should be developed for fetoscope localization. This algorithm should provide a sufficient assessment of whether these types of methods are suitable for tracking the medical instrument during FETO.

Regarding the machine learning methods, the algorithms based on handcrafted features require the design of features which is a time consuming task and it is not guaranteed that it will yield good localization performance. Thus, algorithms focused on deep learning models are advantageous for fetoscope tracking due to their capability to automatically learn features and should be developed

Table 2.1: Overview of US-based instrument localization methods

Property	Physical Space	Projection Space	Handcrafted Features	Deep Learning
Require instrument's geometry prior knowledge	Yes	Yes	No	No
Benefits from availability of more data	No	No	Yes	Yes
Simple image processing operations	Yes	Yes	No	No
Learn contextual information	No	No	No	Yes
Require labeled data	No	No	Yes	Yes
Require model training	No	No	Yes	Yes
Easy interpretability	Yes	Yes	Yes	No
Automatic learning of features	No	No	No	Yes
Semi-automatic	Yes	Yes	No	No
Fast image processing	Yes	Yes	Yes	Yes

instead of the handcrafted features methods. Although deep learning models are difficult to interpret, they can benefit from more data and are able to learn contextual information which is an advantage for the processing of ultrasound images.

It is important to note that the state of the art on instrument localization in 2D US images is focused on clinical interventions where the instrument has several constraints in its movement, such as in US-guided biopsy or US-guided regional anesthesia interventions. There is an absence of complex decisions and ease in reaching the target site. For example, once the entry and target points are defined for a biopsy needle, the needle will follow a straight path and exit along the same path (Antico et al., 2019). Hence, the development of an US-based instrument tracking algorithm in a working space where the instrument has several degrees of freedom is a challenging task, but also, a novel technique that is addressed in this master's thesis.

2.2 State of the art on autonomous robotic ultrasound imaging tracking

Minimally invasive surgeries are usually associated with a long learning curve due to the limited view and lack of depth perception provided by the common optical instruments used during the procedure, such as endoscopes (Antico et al., 2019). Ultrasound imaging is able to provide an extra field of view while giving real time awareness of the interventional site to the surgeon. In standard US guided procedures, a sonographer is responsible for manipulating the ultrasound probe based on interpretation of the US image and mental construction of the anatomy being scanned (Li et al.,

2021a). The quality of the images is highly dependent on the operator, who is subject to heavy physical and cognitive burden.

The introduction of robot manipulators to US guided procedures brings precise force and position control of the probe. Hence, being able to take some of the physical and cognitive burden from the sonographer. There are different levels of autonomy for robotic ultrasound imaging, and the next section deals with the description of the different levels.

2.2.1 Level of autonomy for robotic ultrasound imaging

The definition of the different autonomy levels for robotic ultrasonography here presented is based on the work of Li et al. (2021a), where the increasing level of autonomy is based on "which extent the robotic system may improve ease of use, relieve operator burden, and reduce the user-dependency in US acquisitions" (Li et al., 2021a). The work from Monfaredi et al. (2015) that describes three different types of robotic ultrasound imaging based on autonomy is also taken into consideration.

The first level, or level 0, there is no autonomy, the US probe is manually positioned by the hands of a sonographer. Next level, or level 1, the user directly instructs the probe motion and the robotic system follows those instructions. This level includes teleoperated systems, where the control of the robot is performed in real time by a remote sonographer using a leader/follower approach. Level 1 also includes systems where a sonographer manually performs the desired trajectory with the US probe, while it is being recorded, and then the robotic system automatically executes the recorded trajectory.

Level 2 deals with human-robot cooperation systems. These systems make use of a shared control strategy to position the probe, the robot controls some of the degrees of freedom, and others are controlled by the sonographer. Hence, these systems are able to alleviate some of the burden from the sonographer.

Next, level 3 includes the robotic systems where the scanning path is manually defined, but the robotic US acquisition is completely automatic. Level 4 is used to describe the autonomous robotic US imaging systems. These systems are constituted of three components: a robot arm for probe manipulation, a robot controller, and a tracking system to obtain the pose of the probe. Three types of devices are commonly used to track the probe's pose, optical tracking systems, electromagnetic tracking systems, and encoded mechanical systems. Table 2.2 summarizes the different levels of autonomy in robot ultrasound imaging.

2.2.2 Autonomous robotic ultrasound imaging tracking

In autonomous robotic ultrasound imaging systems the 6 degrees of freedom (DoF) of the US probe are automatically controlled by a robot. The major challenges in the development of a robust and accurate robot control in these types of systems consist in the processing of the images and in the control of the out-of-plane degrees of freedom.

Table 2.2: Levels of Autonomy in Robotic Ultrasound Imaging (Li et al., 2021a) (Monfaredi et al., 2015)

Level of Autonomy	Description
0	Probe is manually controlled
1	Probe motion controlled by a robotic system that follows direct instructions from an operator
2	Human-robot cooperation systems, some degrees of freedom of the probe are controlled by a human, other by a robot
3	Probe motion automatically controlled by a robot where the scanning path was manually pre-defined
4	Autonomous robotic ultrasound imaging systems, probe motion is automatically controlled by a robot

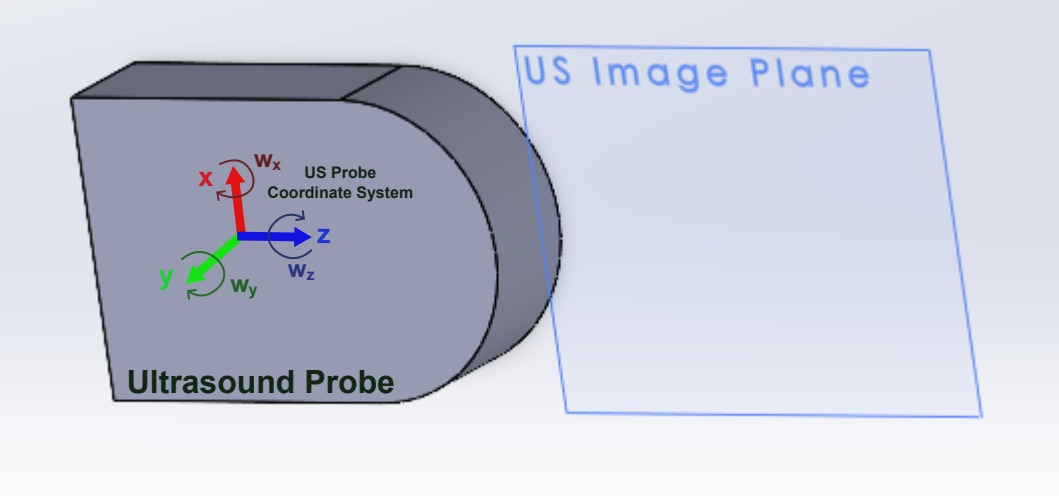
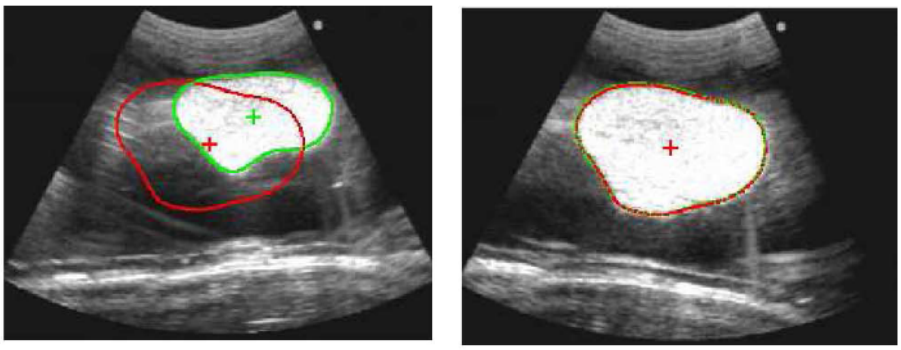


Figure 2.8: Schematic representation of the coordinate system of an ultrasound probe.



(a) (b)

Figure 2.9: Initial (a) and final (b) object cross section during visual servoing, the red line represents the desired cross-section (Mebarki et al., 2010).

The six DoF of a rigid object is described by the position and the rotation of the system. In this work, the US probe is considered a rigid object, and its degrees of freedom can be separated into two types, the in-plane, and the out-of-plane DoF. The in-plane corresponds to the degrees of freedom that when changing with a certain velocity (in-plane motion), maintain the US image plane on the same 2D geometric plane. The out-of-plane degrees of freedom when modified (out-of-plane motion), cause the US image plane to change to a different 2D plane. Using Figure 2.8 as example for a US probe rigid object, the linear velocities v_x and v_z in the x and z axis, respectively, as well as the angular velocity w_y correspond to the three in-plane motions. The linear velocity v_y in the y axis, and the angular velocities w_x and w_z correspond to the out-of-plane motions.

Mebarki et al. (2010) proposes a method that allows both in-plane and out-of-plane probe motion control. The feedback control strategy uses visual features extracted from the US images acquired in real time by the probe. The probe pose is obtained from the mechanical encoders in the 6-DoF medical robot used to actuate the probe. The objective of this method is to position the probe in order to view a desired cross section of a given soft tissue object, this is accomplished by visual servoing, where the desired cross section is described by image moments. These image moments are the visual features used in the feedback control.

The first step in Mebarki et al. (2010) work was the development of the analytical form of an interaction matrix, which maps the US probe motion to the changes in the visual features. This development depends on an online estimation of the object surface. Using this method, the control error converged in less than 90 seconds when performing the visual servoing in a soft tissue object. The object cross section before and after applying the visual servoing control can be observed in Figure 2.9. This method was devoted to motionless objects and its convergence time is slow for real-time tracking applications, thus its application to tracking a fetoscope during FETO, which is a highly dynamic scenario, is not feasible.

Nadeau and Krupa (2011) made use of the image intensity as visual features to create an online estimation of the 3D US image gradient, allowing the real-time computation of the interaction matrix. This approach also applies visual servoing to automatically control the US probe motion.

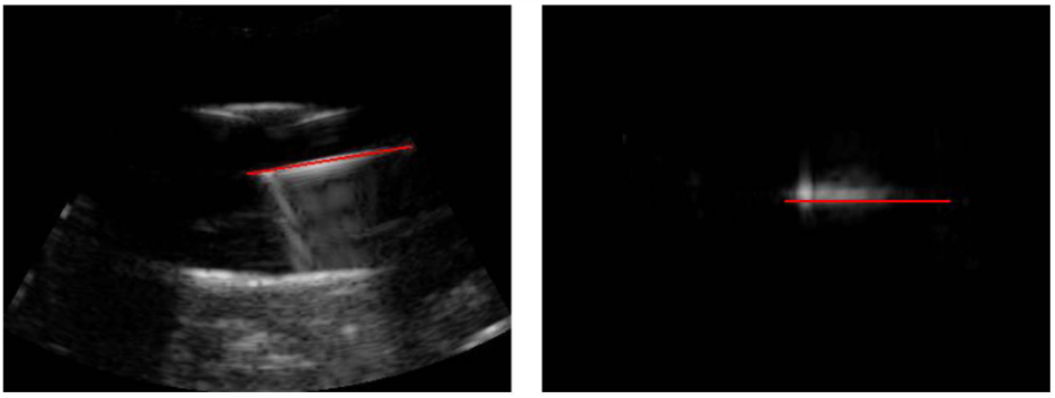


Figure 2.10: Two orthogonal plane views from a US volume with the estimated needle axis in red (Chatelain et al., 2013).

The objective of the control is to view a desired cross section of an organ. However, the visual servoing convergence duration is longer than 60 s, while the organ is not as dynamic as a surgical instrument being manipulated during FETO, thus, this method is also not feasible for fetoscope tracking.

Duflot et al. (2016) uses a similar strategy, but instead of extracting intensity-based features, it extracts the shearlet coefficients of a region of interest in the ultrasound image, in order to obtain noiseless and redundant visual features for a more robust and accurate visual servoing. Not only it is also proposed to achieve a cross section of an organ, but it also requires an user-defined region of interest in the ultrasound image, thus, it is not recommended to real-time tracking of medical instruments.

Chatelain et al. (2013) proposes a control scheme to automatically guide a robot that is holding a 3D US probe in order to keep a needle within the field of view, that is, the US volume. The needle tip position in the US volume is estimated by means of a physical space detection method that makes use of RANSAC to estimate the needle axis with Kalman filtering in closed-loop to achieve the real-time tracking of the needle (see Fig. 2.10). The visual features used to build the interaction matrix applied in the control scheme are the needle tip x and z coordinates in the probe coordinate system (see Fig. 2.11), these coordinates are extracted from processing the 3D ultrasound volume acquired by the probe. This strategy achieved a mean tracking error of 1.10 mm for the x-axis, and 0.23 mm for the z-axis. This method was able to stabilize itself in less than a few seconds.

The work from Chatelain et al. (2013) can be used as a reference to design the in-plane probe motions for an autonomous robotic US imaging system with the objective of tracking a fetoscope during FETO, considering the instrument is within the field of view. Nevertheless, it is necessary to design a novel control scheme for the out-of-plane motions.

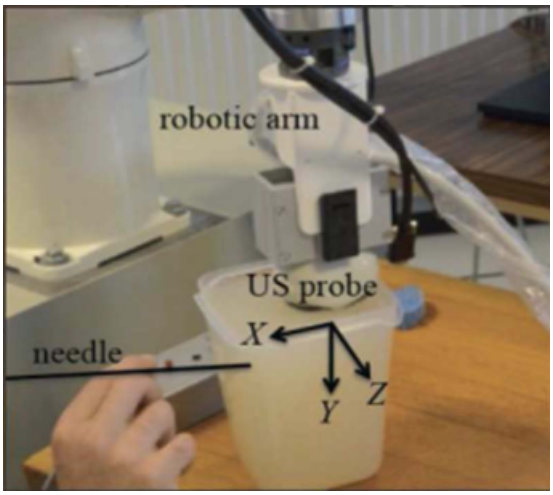


Figure 2.11: Experimental setup used in Chatelain et al. (2013) work showing the 6-DoF robotic arm, the needle, the US probe, and its coordinate system.

2.2.3 Summary and discussion

The majority of the state-of-the-art strategies for autonomous robotic ultrasound imaging tracking are based on visual servoing. The control of the US probe position is done by extracting features from the US image, the features velocities are computed and converted into velocities for the end-effector, which is the probe. This conversion is done through an interaction matrix.

These visual servoing strategies require that the object that is being tracked does not move or have a small velocity. Furthermore, it is necessary to define a desired cross section of the object to be observed in the ultrasound image. These requirements cannot be imposed for the fetoscope tracking during FETO. Not only the instrument to be tracked is constantly moving with variable velocity, but also it is not possible to define a desired cross section since the instrument can have several orientations.

The work of Chatelain et al. (2013) was the only one presented in the state of the art that is not based on visual servoing. Although it uses 3D ultrasound volumes, it presents a good strategy for controlling the in-plane motions of the US probe by keeping track of the instrument tip position. However, it is still necessary to develop a strategy for the out-of-plane motions.

This master's thesis presents the development of a novel framework for tracking a fetoscope using an autonomous robotic US imaging tracking system. This framework must be based on 2D ultrasound images. It must also be able to deal with the in-plane and out-of-plane motions of the US probe with the objective of tracking an instrument that can move in different directions with variable velocity.

Chapter 3

A Comparative Study of Instrument Localization Algorithms in Ultrasound Images

The current chapter aims to describe and compare the five developed methods for fetoscope localization in ultrasound images. One of them is a physical space method that filters the US image with a Gabor filter in order to enhance the instrument. The other four are deep learning methodologies that make use of different CNN architectures for segmenting the instrument in the images.

The chapter starts with a presentation of the experimental setup used to obtain the US images and how the ground-truth for instrument localization is computed. Next, the obtained US image datasets are presented. Then, a complete description is provided for the different localization methods. Finally, the results of the methods are compared and discussed.

3.1 Experimental setup

The B-Mode 2D ultrasound images used to evaluate the developed algorithms and train the machine learning models were obtained by a *Sonosite M-Turbo* (FUJIFILM Sonosite, 2023) ultrasound machine (see Fig. 3.1) with a sampling rate of 30 fps. The images size is 480x640 pixels.

Since the amniotic fluid is 98% water (Tong et al., 2009), a representative and simplistic phantom that simulates the amniotic cavity medium is a box filled with water (see Fig. 3.3). A stainless steel rod shaft (see Fig. 3.2) was used to mimic the fetoscope that must be localized. Table 3.1 shows the rod dimensions in comparison to a fetoscope that is commonly used in FETO.

A *fusionTrack 250* (fTk250) (Atracsys, 2017) optical tracking system (see Fig. 3.1) tracks the pose of the instrument tip and the pose of the US probe, both poses are relative to the OTS base frame. These poses consist in both position and orientation. An optical marker is attached to the rigid instrument and another to the US probe 3.3. Passive fiducials are fixed to the markers. These

(a) *Sonosite M-Turbo* (*FUJIFILM Sonosite, 2023*) (b) *fusionTrack 250* (*Atracsys, 2017*)

Figure 3.1: Ultrasound machine and optical tracking system used in the experimental setup

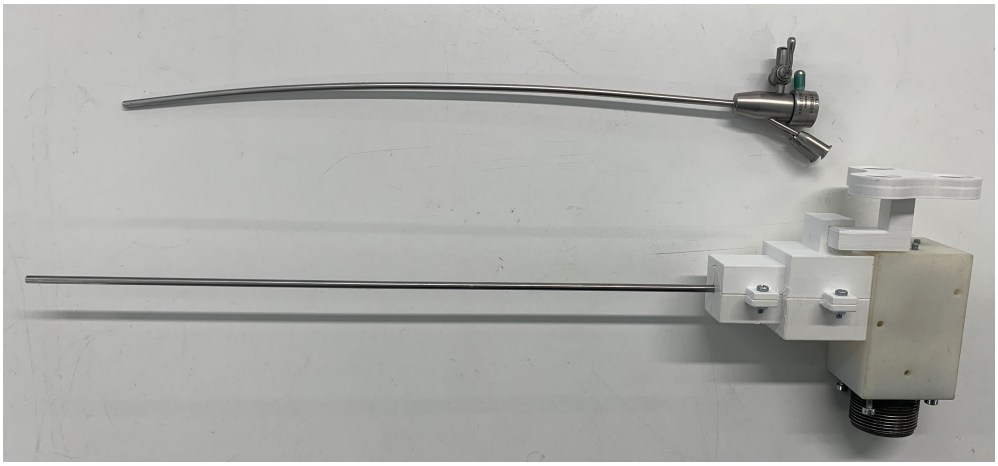


Figure 3.2: Tracheal fetoscope (top) next to the stainless steel rod shaft (bottom) used as the fetoscope to be tracked.

Table 3.1: Stainless steel shaft and fetoscope instrument specifications

Instrument	**Outer Diameter**	**Length**
Stainless Steel Shaft	3.4 mm	25cm
Fetoscope (Deprest et al., 2011)	1.3 mm	30.6 cm

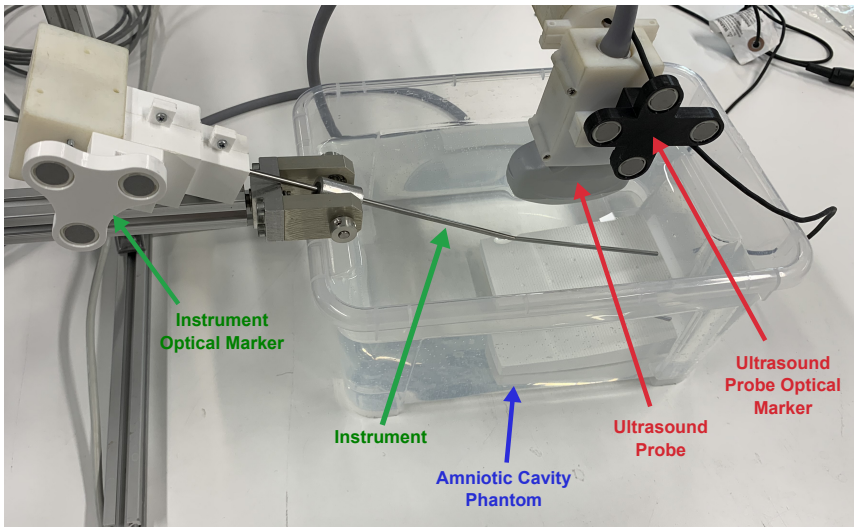


Figure 3.3: Setup for data acquisition.

fiducials reflect infrared light supplied by an external illumination source to the OTS cameras (Atracsys, 2017).

The ultrasound machine sends image data to a frame grabber that is connected to a computer via an USB port, while the data produced by the *fTk250* system is sent through a Gigabit Ethernet connection (Atracsys, 2017) to the same computer.

The computer has an Ubuntu 20.04 operative system, containing an *Intel Core i7* processor, and a *NVIDIA GeForce RTX 2060* graphical processing unit. The Robot Operating System (ROS) middleware is used to save the OTS and US image data. *Python* programming language is used to write scripts to interface with ROS by making use of the *rospy* client library. Figure 3.4 shows the flow of data between the different hardware components in the experimental setup.

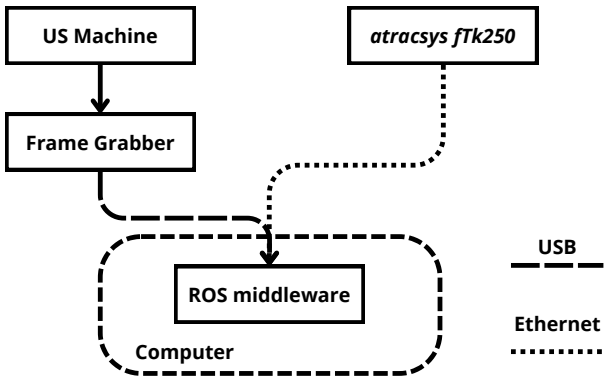


Figure 3.4: Data flow pipeline

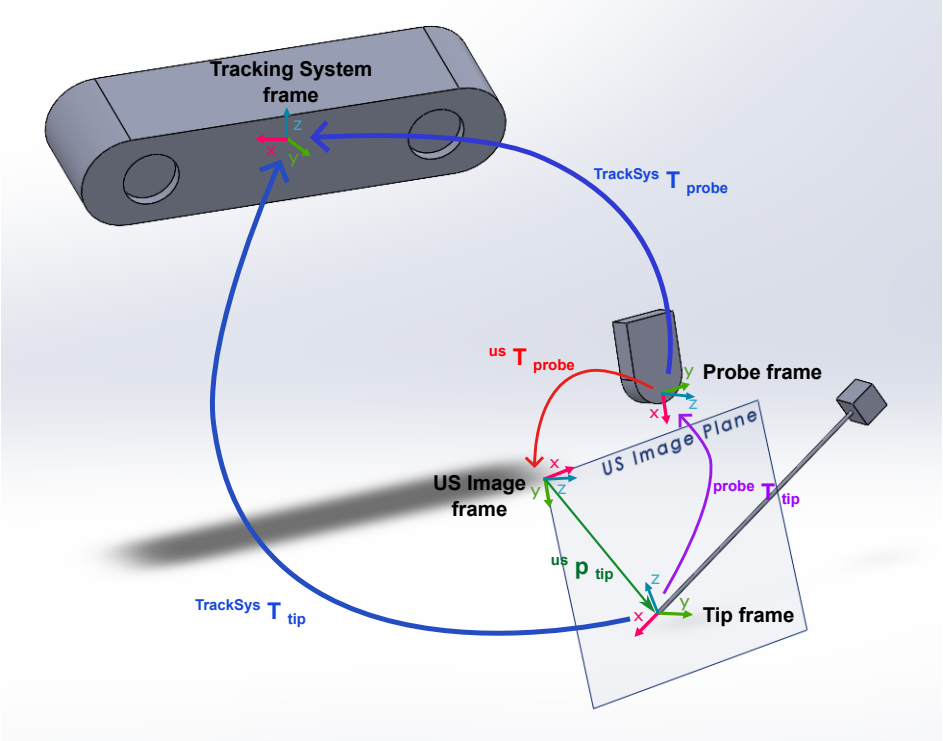


Figure 3.5: Representation of the experimental setup, frames, and transformation matrices between frames. The arrows go from the origin frame to the target frame. Blue arrows represent transformations to the OTS frame, red arrows are transformations to the US image frame, and purple arrows to the probe frame.

Figure 3.5 shows a representation of the experimental setup with the homogeneous transformation matrices between frames. The terminology for transformation matrices has the following rule: a homogeneous transform from frame A to frame B is represented by the matrix ${}^{B}T_{A}$. The origin position vector of frame A relative to frame B is ${}^{B}p_{A}$, and the rotation matrix from frame A to frame B is ${}^{B}R_{A}$.

3.2 Ultrasound image dataset

Two types of ground truths (GT) need to be produced for each ultrasound image, one is the instrument tip position in the US image frame, and another is the segmented instrument mask corresponding to the instrument location in the US image. The tip position GT is the target for the localization, or tracking task, while the mask GT is the target for the segmentation task. All localization methods developed have the same objective which is tracking the instrument tip in consecutive US image frames, in order to achieve this objective, an instrument segmentation task needs to be performed. The DL models use the instrument mask GT in order to be trained to perform this segmentation task.

3.2.1 Tip position ground-truth

The tip position GT relative to the US image (${}^{us}p_{tip}$) is simply obtained by representing the tip frame origin in the US image frame (see Fig. 3.5). This transformation is done by applying the following equations:

$$ {}^{TrackSys}p_{tip} = {}^{TrackSys}T_{tip} \cdot [0,0,0,1]^T \tag{3.1} $$

$$ {}^{probe}p_{tip} = ({}^{TrackSys}T_{probe})^{-1} \cdot {}^{TrackSys}p_{tip} \tag{3.2} $$

$$ {}^{us}p_{tip} = {}^{us}T_{probe} \cdot {}^{probe}p_{tip} \tag{3.3} $$

where the transformation matrices ${}^{TrackSys}T_{tip}$ and ${}^{TrackSys}T_{probe}$, representing the transformation from the instrument tip and the US probe to the optical tracking system, respectively, are directly obtained from the *fTk250* system. ${}^{TrackSys}p_{tip}$ is the tip position in the optical tracking system frame and ${}^{probe}p_{tip}$ is the position of the instrument tip in the US probe frame. ${}^{us}T_{probe}$ is the transform matrix from the probe frame to the US image frame that is computed by a robotic ultrasound image calibration method. The calibration method utilized a Z phantom consisting of three layers of Z-shaped crossing nylon wires, and by performing specific US probe scanning trajectories to record specific positions in the ultrasound image, a least squares method is used to obtain the ${}^{us}T_{probe}$ matrix. From ${}^{us}T_{probe}$, not only the rotation and translation between frames can be obtained, but also the scaling from image pixel distance to physical distance, which is equal to 0.34 mm/pixel. This calibration step is detailed in the work of Li *et. al* (Li et al., 2021b).

3.2.2 Instrument segmentation ground-truth

Regarding the instrument segmentation mask computation, it is necessary to obtain the orientation of the instrument relative to the image. Firstly, a parametrization of this orientation is defined by means of two angles, an azimuth angle θ and an altitude angle ϕ (see Fig. 3.6). ϕ is computed using

$$ \phi = arccos\left(\frac{z_{us} \cdot x_{tip}}{||z_{us}||.||x_{tip}||}\right) \tag{3.4} $$

where z_{us} is the basis vector from the ultrasound frame that is normal to the ultrasound image plane, x_{tip} the x-axis basis vector of the instrument tip frame (see Fig. 3.6), and $||.||$ the norm of a vector. θ is computed by the following equation:

$$ \theta = atan2\left(y_{us} \cdot x_{tip},\ x_{us} \cdot x_{tip}\right) \tag{3.5} $$

where x_{us} and y_{us} are, respectively, the x and y-axes basis vectors from the ultrasound image frame.

In case the instrument is parallel to the image plane ($\phi \approx 0$), the segmentation mask is produced by using the tip position ${}^{us}p_{tip}$ and the angle θ. The instrument rod shaft has a cylindrical shape, thus the instrument segmentation mask corresponds to a rectangle, which is the result of

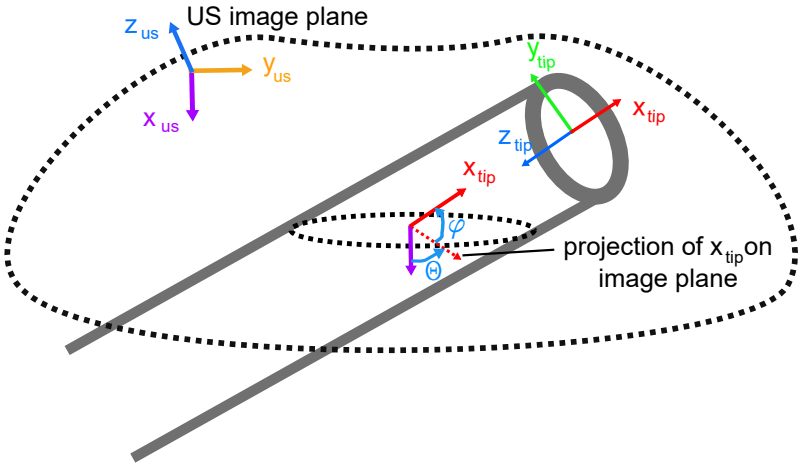
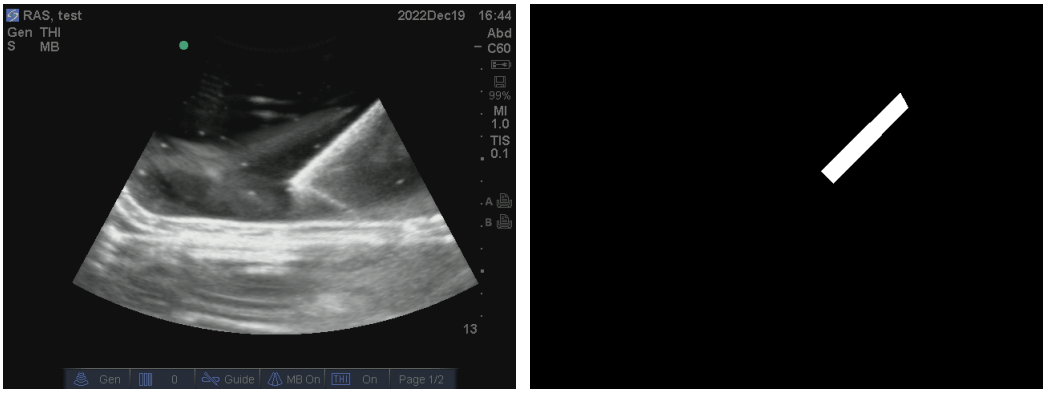


Figure 3.6: Parametrization of the instrument orientation relative to the image frame with the azimuth angle θ and the altitude angle ϕ.

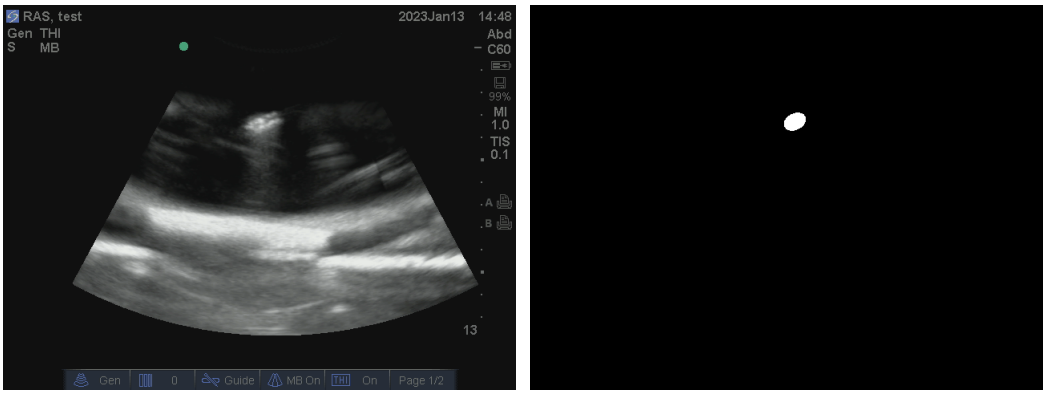
intersecting a cylinder with a parallel plane. Since the instrument has a diameter of 3.4 mm (Table 3.1), and the scaling from pixel distance to physical distance is 0.34 mm/pixel, the width of the segmentation mask must be 10 pixels, while the length is variable, depending on the tip position. A segmentation example is presented in Figure 3.7.



(a) Original ultrasound image (b) Ground truth segmented ultrasound image

Figure 3.7: Example of instrument segmentation mask when the instrument is parallel to the image plane

In case the instrument is not parallel to the image plane ($\phi > 0$), the segmented mask is computed by modelling the instrument as a cylinder and intersecting it with the image plane. An example of a segmented image produced by this intersection is shown in Figure 3.8.



(a) Original ultrasound image (b) Ground truth segmented ultrasound image

Figure 3.8: Example of instrument segmentation mask when the instrument is not parallel to the image plane.

3.3 Localization algorithms

3.3.1 Gabor filter localization algorithm

The Gabor filter algorithm is a semi-automatic approach, hence it depends on a first estimate of instrument location given by the user. Using this estimate, the algorithm will obtain an estimate of the instrument location by enhancing the instrument structure in the image with a Gabor filter. Afterward, a particle filter is used to predict the instrument tip position based on the previous tip location and on the optical flow information of feature points around the instrument.

This algorithm was based on the work of Kaya and Bebek (2014), where the Gabor filter is used to localize a biopsy needle in US images. The Gabor Filter methodology developed in this work has the following steps: pre-processing, binarization, morphological operations, axis localization, tip localization, and statistical filtering.

3.3.1.1 Pre-processing

The first step is to crop the image, eliminating unnecessary metadata that is presented in the image. The cropped image size is 390x540 pixels. If the US image is the first frame of an US video, the user introduces an estimate for the instrument axis position by selecting two points in the image. This axis estimate is a finite line which starts in one of the user-defined points and finishes in the other. Then, a region of interest (ROI) is defined around the instrument axis. The ROI is a rectangle with 40 pixels width, a length equal to 110% of the axis length, and defines an estimate for the instrument location. In the following frames, the ROI is obtained from the previous frame to the one that is being processed.

Next, the image is filtered with a Gabor filter. The Gabor filter is a linear filter that analyzes whether there is a specific frequency content in the image in a specific direction. The 2D function of this filter in the spatial domain is given by

Table 3.2: Gabor filter parameters

λ	ψ	γ	σ
15	0	0.05	10

$$g(x,y) = exp\left(-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right) exp\left(j\left(2\pi\frac{x'}{\lambda}+\psi\right)\right) \tag{3.6}$$

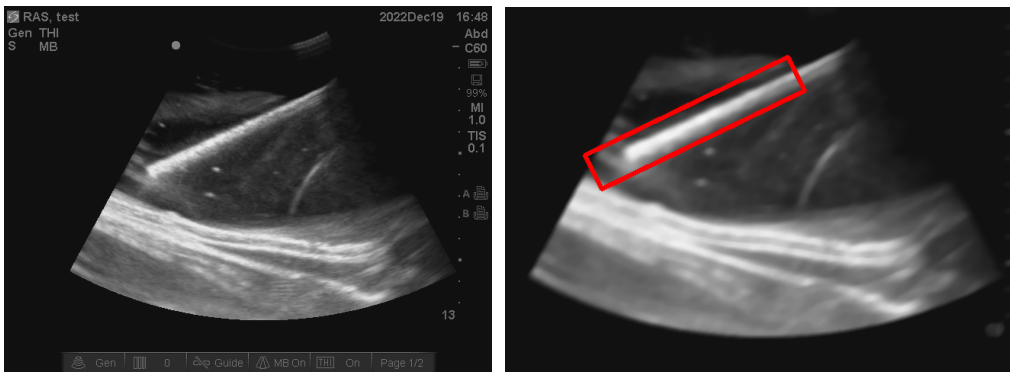
where

$$x' = x\cos\Theta + y\sin\Theta \tag{3.7}$$

$$y' = -x\sin\Theta + y\cos\Theta \tag{3.8}$$

with λ being the wavelength of the sinusoidal factor, Θ the orientation of the Gabor kernel, ψ the phase offset, γ the spatial aspect ratio, and σ the standard deviation of the Gaussian envelope. The value of the parameter Θ is equal to the ROI orientation in the image. Since the ROI defines an estimate for the axis location, its orientation is also an estimate for the instrument orientation. Thus, the Gabor filter kernel has the same orientation as the instrument, which will enable an enhancement of the instrument structure while filtering out the orthogonal structures. The size of the kernel is 9x9 pixels. The values of the other parameters are shown in Table 3.2. Those values were manually selected by evaluating a finite number of different values and choosing the ones which gave the best instrument tip position estimate, that is, the closest to the ground truth.

Then, a median filter with a kernel size of 7x7 pixels is used to reduce the speckle noise that is characteristic of ultrasound images (Zhao et al., 2020). Figure 3.9 shows an example of the pre-processing output with a defined ROI.



(a) Ultrasound image before pre-processing

(b) Ultrasound image after pre-processing with overlayed ROI (red rectangle)

Figure 3.9: Gabor filter algorithm pre-processing output.

Table 3.3: Adaptive threshold parameters

b	c
51	-7

3.3.1.2 Binarization

After pre-processing the US image, a binarization step is performed. Because this binarization process is used to segment the instrument in the image, it can also be called a segmentation step.

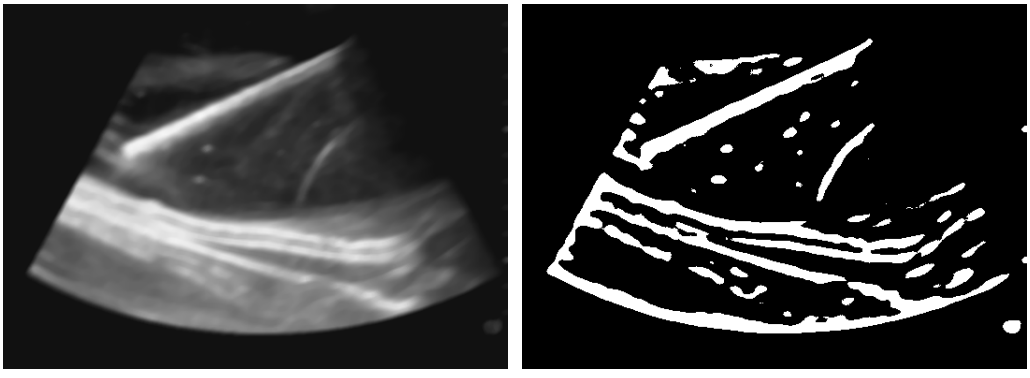
An adaptive threshold is utilized to binarize the image. The threshold value is set on a pixel-by-pixel basis by computing a weighted average of the bxb region around each pixel minus a constant c (Kaehler and Bradski, 2016). The adaptive thresholding technique is useful when there are strong illumination regions or reflectance gradients that are needed to be thresholded relative to the general intensity gradient (Kaehler and Bradski, 2016), which is the case of the US images being processed in this work. The parameters to calculate the threshold value are defined in Table 3.3. Those values were manually selected in the same manner the Gabor filter parameters were selected in the pre-processing.

An example of a binarized, or segmented, US image can be observed in Figure 3.10.

3.3.1.3 Morphological operations

The thresholded image (see Fig. 3.10b) contains a lot of blobs and other structures that need to be filtered out of the image. Hence, two morphological operations are used to clean the image. Firstly, an erosion operation removes the blobs that are smaller than the structuring element, while removing protrusions of the other blobs. The second operation is a dilation that will fill the concavities of the remaining structures and expand its regions. The structuring element of both operations is a square with size 9x9 pixels, the anchor point is the center pixel.

The result of the morphological operations is intersected with the interior region of the ROI defined in the pre-processing. Figure 3.11 contains an example of this intersection.



(a) Ultrasound image before binarization (b) Ultrasound image after binarization

Figure 3.10: Gabor filter algorithm binarization output.

(a) Ultrasound image before morphological operations and intersection

(b) Ultrasound image after morphological operations and intersection

Figure 3.11: Gabor filter algorithm morphological operations output.

3.3.1.4 Axis localization

After the morphological operations, there are some blobs, or connected components, in the image. One of these components corresponds to the instrument. A random sample consensus (RANSAC) line estimator is fitted to the blobs that contain at least 50 pixels, while the blobs smaller than 50 pixels are disregarded. The RANSAC estimates the instrument axis location and orientation. The component with the highest RANSAC fitting score is considered as being the instrument. An example of the estimated instrument axis is presented in Figure 3.12.

The estimated axis is used to build a new ROI in the same way the user-defined axis was used to define a ROI in the first frame. This new ROI is used in the next frame instead of the ROI from the first frame, meaning the ROI location and orientation is updated according to the instrument motion.

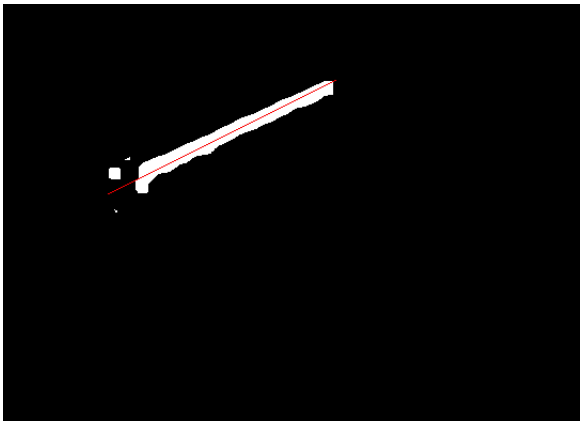


Figure 3.12: Instrument axis estimate (red line).

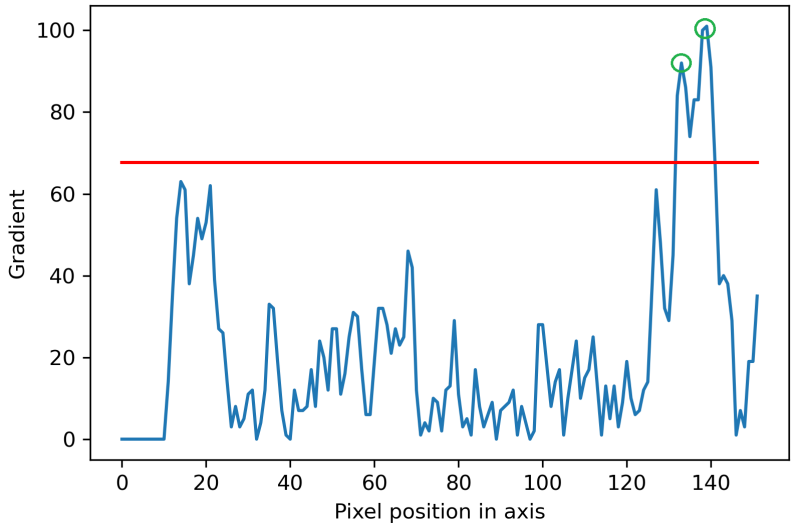


Figure 3.13: Gradient along instrument axis with gradient threshold (red line) and possible tip positions (green circles).

3.3.1.5 Tip localization

After localizing the axis, the instrument tip position is estimated. Considering the instrument is hyperechoic, while the background is hypoechoic, by analyzing the intensity of the pixels along the instrument axis as proposed in Zhao et al. (2015). A drop in the pixel's intensity along the axis will happen at the end of the instrument, this drop corresponds to a high gradient value. The intensity gradient $g(i)$ is calculated along the instrument axis as expressed in the following equation:

$$g(i) = I(i) - I(i - step), \tag{3.9}$$

where i is the ith pixel along the instrument axis, $I(i)$ the intensity of the ith pixel, and $step$ a constant defined as 10 in this work. The $step$ constant is used to avoid detecting intensity gradients that are a consequence of the noise in the image.

More than one region may have a high gradient along the axis, thus, a threshold is used to determine the possible tip positions. The tip location is selected as being the pixel along the axis that is a local gradient maxima with a gradient value higher than a threshold. The threshold used was manually selected and is equal to the mean of the gradient values along the axis plus two times the standard deviation of the gradient. If there is still more than one possible tip position, the one that is farthest away from the US probe is selected. That is because the closest possible tip position may be the beginning of the instrument and not the tip.

Figure 3.13 shows the gradient along the instrument axis in an US image. Figure 3.14 contains an example of the tip position estimate obtained from the gradient computation.

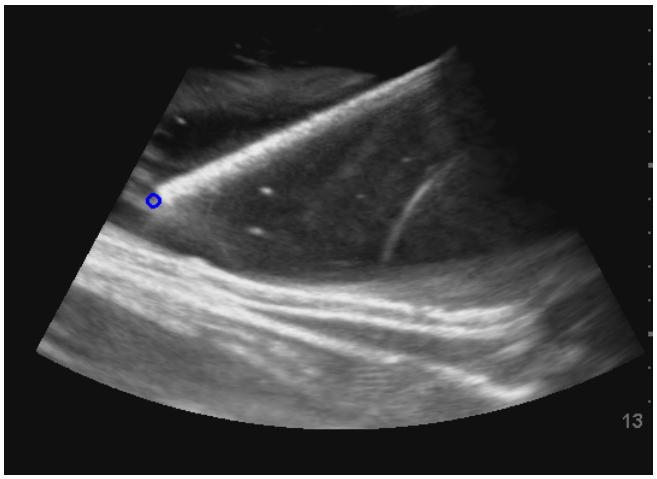


Figure 3.14: Instrument tip position estimate (blue circle).

3.3.1.6 Statistical filtering

A statistical filter allows the prediction of the current state of a system based on the previous state, a measurement of the state, and some other apriori information about the system. The statistical filtering process can be divided into two phases (Kaehler and Bradski, 2016). In the first phase, or prediction phase, information learned in the past states is used to predict the current state. In the second phase or correction phase, a measurement of the state is performed and then reconciled with the state prediction from the previous phase.

Statistical filters have been extensively used in different works to track and predict the position of an instrument in a sequence of ultrasound images. Alsbeih et al. (2020) compares two of the most used statistical filters, the Kalman filter, and the particle filter with the objective of tracking the tip of a biopsy needle in US images. In Alsbeih et al. (2020), it is shown that the best tracking performance can be achieved by using a particle filter, which is also used in this work.

In the particle filter algorithm, the probability distribution of a state is approximated by a sample set $\{s(t), w(t)\}$, being $s(t)$ the hypothetical state, and $w(t)$ the weight of the state in frame t. This sample set is also called a particle, and usually, many particles are used to estimate the state.

In this work, the state is the tip position estimate $[u, v]^T$ in the image coordinate system. This estimate is given based on the previous system state $s(t-1)$, the state measurement $s^m(t)$ produced in the tip localization step, and the instrument velocity measurement $v^m(t)$ that is given by a sparse optical flow between consecutive frames.

The optical flow is calculated based on Lucas and Kanade sparse optical flow algorithm (Open Source Computer Vision, 2023), this algorithm only uses a limited quantity of points in the image to compute the optical flow, that is the pattern of apparent motion of the objects in an image. Thus, the optical flow can be used as an estimate of the instrument velocity in the image. The maximum number of points used to compute the optical flow is 30.

The points used in the Lucas and Kanade algorithm must be inside the ROI defined in the axis

localization step, and are selected based on Shi and Tomasi work (Shi and Tomasi, 1994), which detects points in the image that contain good features to track. The average velocity of the points that are being tracked is calculated and used as the velocity measurement $v^m(t)$. The velocity of each point is given by the optical flow algorithm, which requires two images as input, these two images are the current image being processed and the previous frame.

The steps of the particle filter algorithm are detailed below.

Particle Filter Algorithm

(a) Initialize n particles equally weighted and with state distribution according to a gaussian distribution $N(\mu, \delta)$. The mean μ of the gaussian distribution is the tip localization obtained for the first US image.

(b) Measure the average motion velocity of the instrument $v^m(t)$, using the optical flow between the previous and current frame.

(c) Update the states with the measured velocity: $s_i(t) = s_i(t) + v^m(t) + \eta$, being η the velocity measurement noise taken from a gaussian distribution.

(d) Get a measurement for the state $s^m(t)$, which corresponds to the output of the tip localization step added with a gaussian white noise ν that is the measurement noise.

(e) Update the particle weights using a probability density function $w_i(t) = \frac{exp(-x_i^2/2)}{\sqrt{2\pi\xi}}$. x_i is the distance between the ith state $s_i(t)$ and the measurement $s^m(t)$, and ξ is a fixed value corresponding to the standard deviation of the probability density.

(f) The current state estimation is produced by the weighted average $s^* = \sum_{i=0}^{i=n} s_i(t) \times w_i(t)$. The final estimate s^* corresponds to the tip position given by the Gabor filter localization algorithm.

The different parameters of the particle filter are presented in Table 3.4. The values of these parameters were manually selected in the same manner the Gabor filter parameters were chosen.

Table 3.4: Particle Filter Parameters

n	δ	η	ν	ξ
2000	5	$N(0, 1)$	$N(0, 0.2)$	0.2

A diagram representing all the steps of the Gabor filter localization algorithm is shown in Appendix A.

3.3.2 Deep learning localization algorithms

The deep learning localization algorithms are composed of three parts, pre-processing, image segmentation, and post-processing. The pre-processing includes some image transformations, such as cropping, or normalization. Then, the image goes through the deep learning model, a neural

network, which produces a segmentation mask for the instrument location. The post-processing consists in the localization of the instrument axis and tip. These algorithm parts are detailed below.

Pre-processing

(a) The image is cropped to a size of 340x450 pixels, in order to eliminate the unnecessary metadata from the US image

(b) The image is resized to 256x256 pixels

(c) The image pixels values are normalized by removing the mean and dividing by the standard deviation

Segmentation

(a) The image is fed into a deep learning neural network that was trained to perform the instrument segmentation, thus, the output of the network is a segmentation mask of the instrument location

Post-processing

(a) The instrument axis position is estimated in the segmented image produced by the neural network model, this estimation corresponds to the same procedure described in section 3.3.1.4 for the Gabor filter algorithm

(b) Next, the tip position is estimated as being the pixel along the axis that is farthest away from the US probe

Different neural network architectures were used to perform the segmentation task. The training of these architectures and their description are discussed in the following sections.

3.3.2.1 Deep learning training

A dataset with 5292 US images was obtained using the experimental setup, all these images have a ground-truth mask for segmentation and a ground-truth for the instrument tip position, which were computed according to the process explained in section 3.2. The dataset was then split into two sets, one with 4619 images for training the neural networks, and the other with 673 images to test it.

The training was performed with a cross-validation with random subsampling approach. In each training epoch, the training set is randomly divided with an 80:20 ratio, 20% of the images are used to validate the neural network, while the other 80% are used to optimize, or update, the neural network weights in order to reduce a loss function. The validation allows the assessment of the model performance in images the model has not seen during training. The Adam optimizer is used to update the neural network weights. The learning rate of the optimizer started at 0.001, and after every 5 epochs, it was multiplied by a factor equal to 0.1. The number of epochs was set to 20.

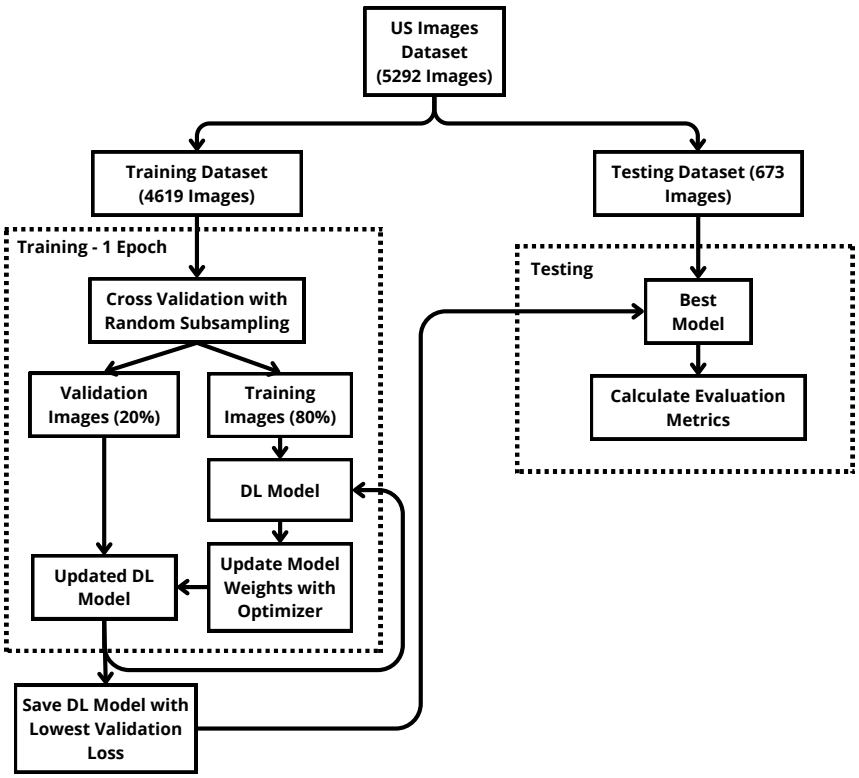


Figure 3.15: Schematic representation of the training and testing of the Deep Learning Models

After the model weights are updated, the validation set is applied to the model. The performance of the model on the validation set is registered in terms of the loss function values. At the end of the training epochs, the model with the lowest loss on the validation set is saved and used on the test dataset. Figure 3.15 shows the training and testing process for the deep learning models.

The loss used to train the models for the segmentation task was based on Zhao et al. (2022) work. The loss function is a hybrid function consisting of a sum of two commonly used loss functions (Equation 3.12), one is the binary cross entropy loss (BCE), and the other is the Dice score loss.

$$BCE = -\sum_{i=1}^{i=N} y_i log(p_i) + (1-y_i) log(1-p_i) \tag{3.10}$$

$$Dice = 1 - \frac{2|A \cap B|}{|A \cup B|} \tag{3.11}$$

$$Loss = BCE + Dice \tag{3.12}$$

where N is the number of pixels in the image, y_i the ground truth, that is 1 for instrument pixels and 0 for background pixel, p_i the probability of a pixel being an instrument pixel, $|A \cap B|$ the area of overlap between the output segmentation mask (A) from the neural network and the ground-truth segmented mask (B), and $|A \cup B|$ the total number of foreground pixels in both A and B.

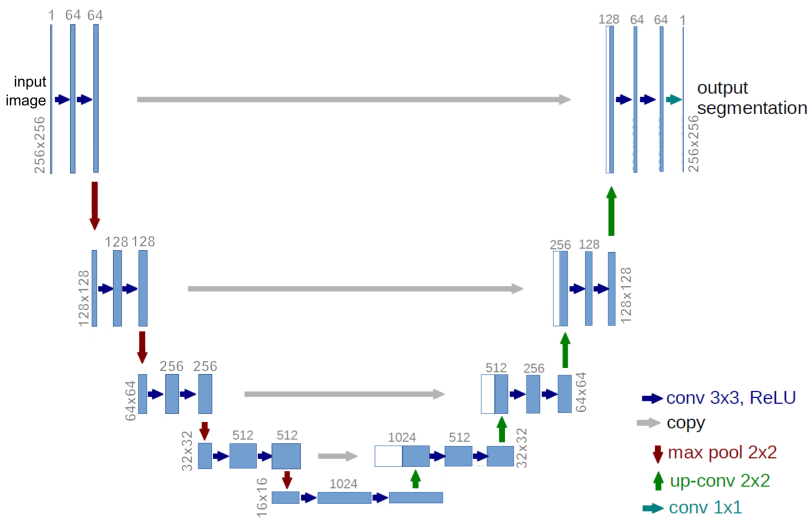


Figure 3.16: U-Net architecture with modified input and output sizes.

All the neural network models were trained on a Windows 10 OS making use of a *NVIDIA GeForce RTX 3060* GPU to accelerate the training.

3.3.2.2 U-Net method

The U-Net neural network was already presented in section 2.1.2. In the original architecture, the input and output have different sizes, while in the architecture used in this work, both the input and output have the same size of 256x256 pixels (see Fig. 3.16). All the operations used in the U-Net remain the same.

3.3.2.3 EU-Net method

The Enhanced U-Net (EU-Net) has only one difference from the U-Net architecture presented in the previous section, which is a second channel in the input (see Fig. 3.17). One of the channels corresponds to one frame of an US video, while the second channel is the same frame, but after a thresholding operation. Since the instrument is hyperechoic and the water is hypoechoic in the US images, the instrument is most likely surrounded by pixels with low intensity. An adaptive thresholding operation will separate the high intensity pixels in a certain region of the image from the low intensity pixels, thus, indicating a possible instrument location as the foreground. The adaptive threshold is used in the second channel, which works as one hot encoding for each pixel, being the foreground pixels the ones with a higher probability of being from the instrument. This approach may enable the U-Net model to more accurately locate the instrument.

3.3.2.4 W-Net method

The W-Net was already presented in section 2.1.2, and the architecture used for the fetoscope localization is the same one presented in Chen et al. (2022). One of the inputs of the W-Net

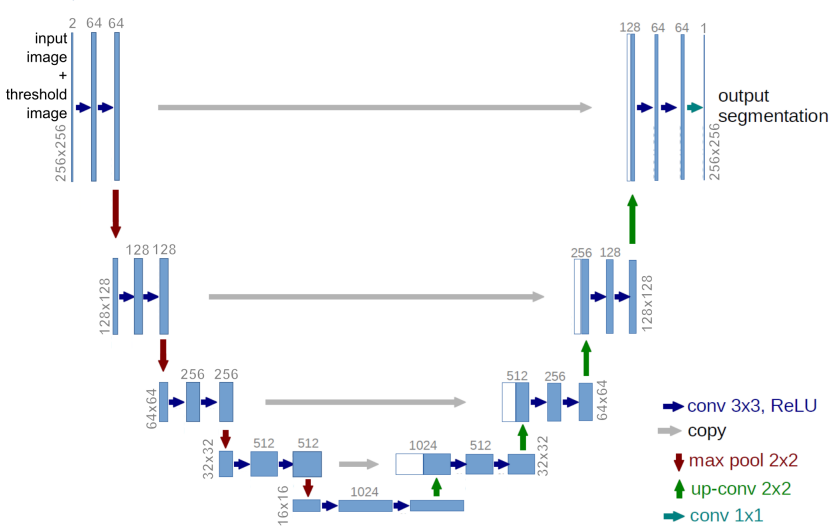


Figure 3.17: Enhanced U-Net architecture.

network is a certain US image from the US video being recorded by the US probe, while the second input is an image delayed by 10 frames from the other input. Since the W-Net requires two inputs, it also requires more memory storage in order to save the last 10 frames received from the US machine.

3.3.2.5 OEU-Net method

The Orientation Estimation U-Net (OEU-Net) is a multi-task neural network that provides two outputs. The OEU-Net architecture has a U-Net like structure to perform the instrument segmentation task, and a fully connected network based on Ahmad et al. (2020) that is used to provide a pose estimation for the instrument. The pose is represented by two angles, the altitude and azimuth angles described in Figure 3.6. The OEU-Net architecture is presented in Figure 3.18.

Since the OEU-Net is designed to achieve two objectives at the same time, it requires two loss functions in order to train this network. The loss function for the segmentation task is the hybrid loss (Eq. 3.12) which is also used for training the other models. The pose estimation loss function is the mean squared error loss (MSE):

$$MSE = \sum_{i}^{N} (y_i - \hat{y}_i)^2 \tag{3.13}$$

where N is the number of values being estimated, in this case, there are two angles being estimated, the altitude and the azimuth angles. $\hat{y}_i$ is the angle value estimated by the OEU-Net and y_i the ground truth angle value.

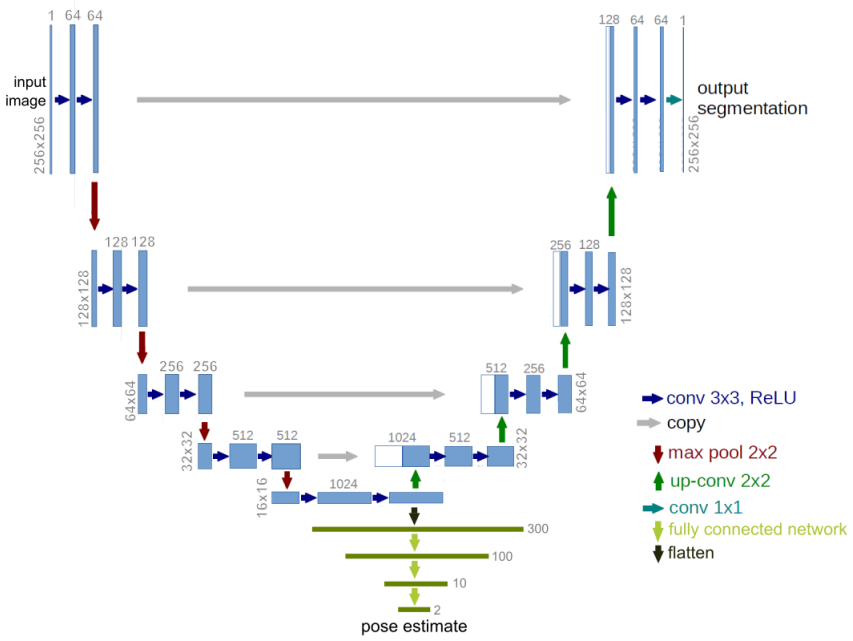


Figure 3.18: OEU-Net architecture.

3.4 Results and Discussion

3.4.1 Deep learning training

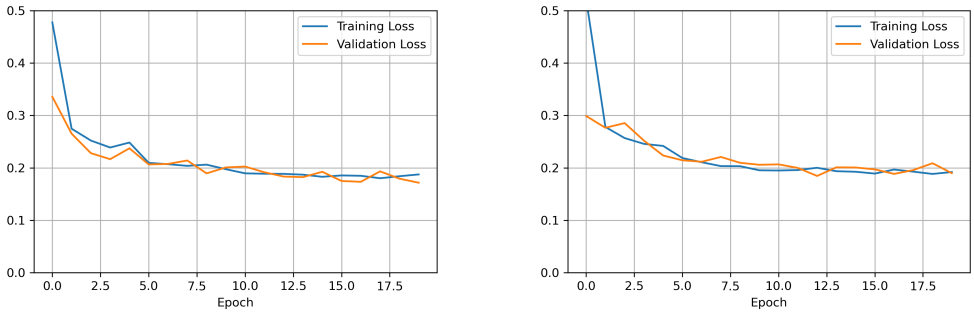
The training and validation loss are calculated throughout the DL models' training epochs (see Fig. 3.19). The model with the lowest validation loss is the one that is saved and tested. It can be observed in Figure 3.19 that after around 10 epochs the training and validation losses become almost static, meaning the model does not have a significant improvement after that point.

The U-Net, EU-Net, and OEU-Net networks converged to similar validation loss values, around 0.2, while the W-Net model converged to a validation loss close to 0.15, meaning that it was able to achieve a better overall segmentation performance in the training and validation datasets. The loss of the OEU-Net is on a different scale and has a different behaviour due to the addition of the MSE loss that is used to learn the instrument orientation.

The time required to train each neural network is presented in Table 3.5.

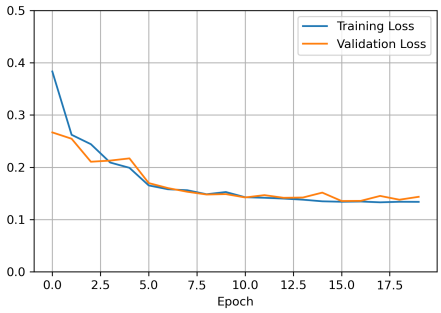
Table 3.5: Total Training Time - Deep Learning Models

Model	U-Net	EU-Net	W-Net	OEU-Net
Training Time (min)	32	44	80	50

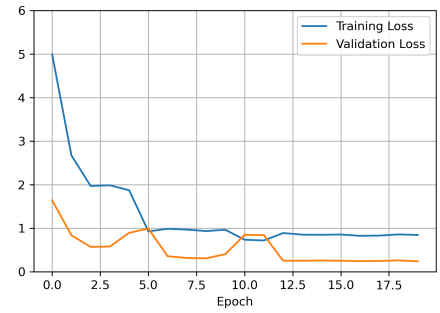


(a) U-Net

(b) EU-Net



(c) W-Net



(d) OEU-Net

Figure 3.19: Training and validation loss history.

Table 3.6: Segmentation Results on Test Dataset - DL models

Model	IoU (%)	Dice (%)	Precision (%)	Recall (%)	Computation Time (ms)
U-Net	50.7	60.0	64.9	59.9	25.54 ± 6.51
EU-Net	52.2	63.9	**70.5**	60.6	24.87 ± 9.98
W-Net	**57.2**	**67.8**	69.0	68.3	76.25 ± 9.01
OEU-Net	54.2	65.7	63.7	**73.0**	26.76 ± 7.88

3.4.2 Deep learning segmentation and prediction performance

The neural networks were all trained with the objective of segmenting a medical instrument in ultrasound images. The OEU-Net had an extra objective which was the instrument orientation prediction. The segmentation performance is evaluated by means of five metrics: Intersection over Union (IoU), Dice score, precision, recall, and the computation time required to segment one image. These evaluation metrics are computed using the following equations:

$$IoU = \frac{|A \cap B|}{|A \cup B|} \tag{3.14}$$

$$Dice = \frac{2|A \cap B|}{|A \cup B|} \tag{3.15}$$

$$Precision = \frac{TP}{TP + FP} \tag{3.16}$$

$$Recall = \frac{TP}{TP + FN} \tag{3.17}$$

where A denotes the ground-truth segmentation mask, B the neural network segmentation output, TP the true positive predictions, FP false positive predictions, and FN false negative predictions.

IoU represents the area of overlap between the predicted and ground-truth segmentation masks. The Dice score is positively correlated to the IoU, but the IoU penalizes single instances of bad classification more than the Dice score. Hence, the Dice score is a metric closer to the average performance. Precision shows how well the model can identify the foreground and the instrument while highlighting the limiting of false positives. Recall indicates the model's ability to capture the positive instances. Table 3.6 presents the average of each evaluation metric regarding the segmentation performance on the test dataset.

The W-Net had the overall best performance, which corroborates with the loss values during training and validation. However, the W-Net is the slowest model, it takes about three times more time to segment an image than the other models.

Since the evaluation metrics do not follow a Gaussian distribution, the violin plot for each metric is used to grasp a better understanding of the model's performance (see Figs. 3.20, 3.21, 3.22, and 3.23).

A common behaviour is observed in the violin plots for the U-Net model, where there is a higher density of values around 0.9 and around 0. While in the violin plots of the EU-Net, it is possible to observe that there is still a high density of values around 0.9, but the density of values

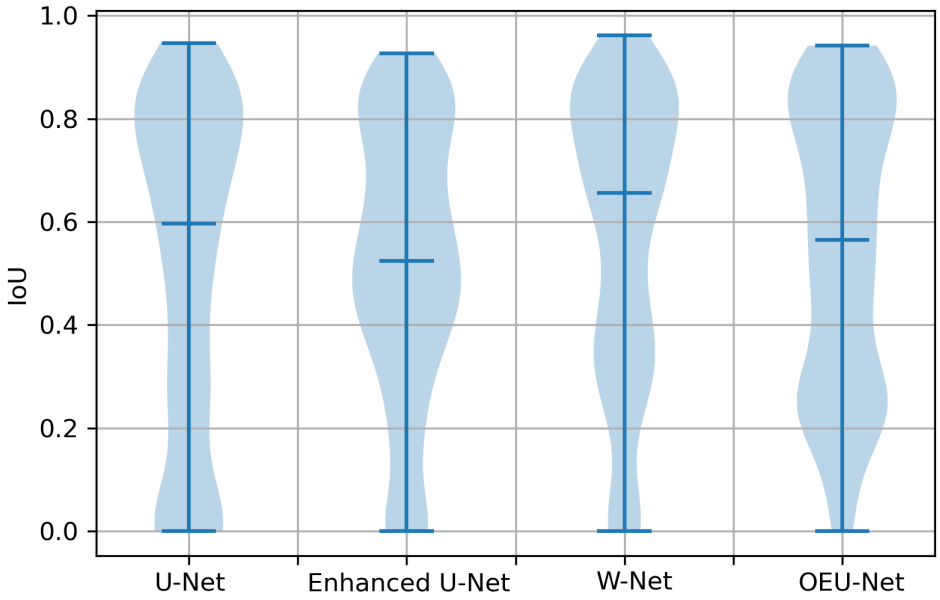


Figure 3.20: Violin plot of IoU score from deep learning models segmentation performance.

around 0 was reduced, while the density around 0.5 is increased. Thus, the implementation of the second channel with the thresholded image in the EU-Net was able to improve the segmentation performance. Since there was a reduction of results around the value 0, it means that in the images where the U-Net struggled to find a correct location for the fetoscope, the second channel from the EU-Net model provided possible locations for the fetoscope that were taken into consideration when producing the segmentation mask. It is important to note that the EU-Net had the best precision results.

W-Net had the best overall results for the evaluation metrics. It presented an even lower density of values around 0 in the violin plots. Hence, the introduction of two frames into the model and the possibility to learn motion related features contribute to a better segmentation performance.

The OEU-Net was the model with the highest recall and lowest precision, which means that it overestimates the foreground mask, leading to a higher quantity of positive pixels on the segmentation output. This overestimation of the fetoscope area may contribute to the reduced density of results around 0 that is shown in the violin plots. When compared to the other models, the density of values for the OEU-Net is more homogeneously distributed between 0.1 and 1 for the IoU, Dice, and precision scores. On the other hand, the recall scores distribution is concentrated around 0.8.

The joint training of the instrument orientation prediction and instrument segmentation that was performed with the OEU-Net contributed to a higher number of foreground pixels in the output segmentation mask while reducing the number of images that presented results equal to zero.

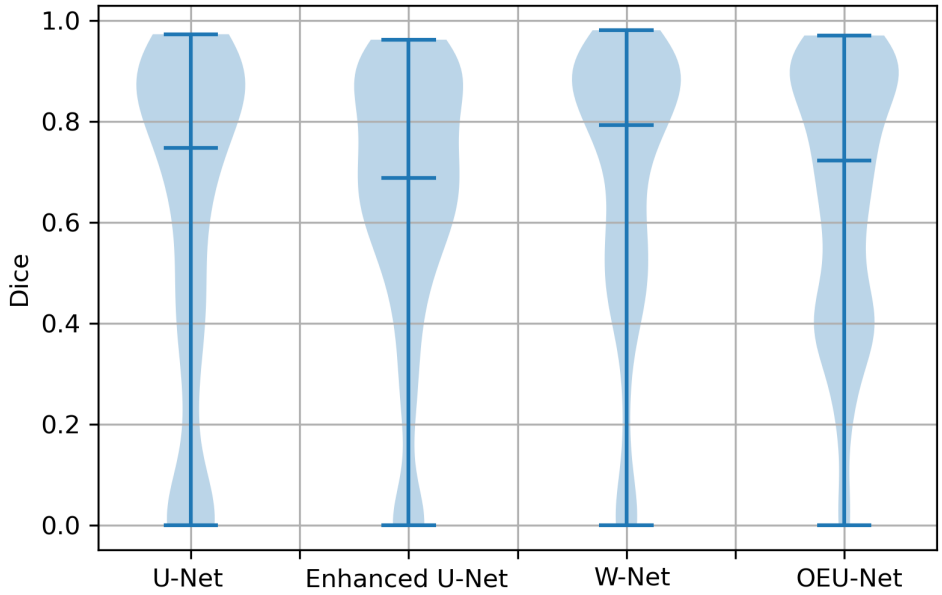


Figure 3.21: Violin plot of Dice score from deep learning models segmentation performance.

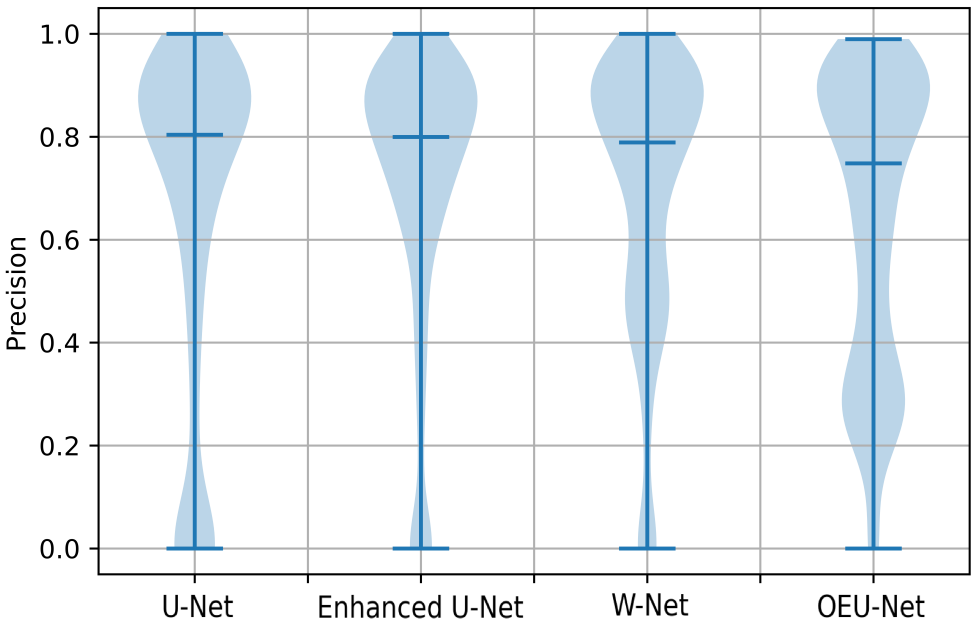


Figure 3.22: Violin plot of precision score from deep learning models segmentation performance.

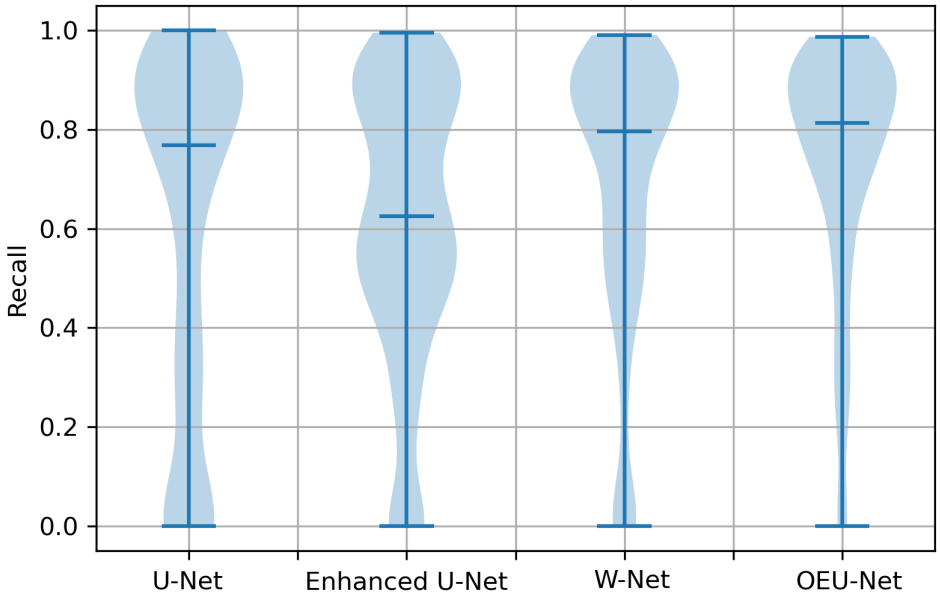


Figure 3.23: Violin plot of recall score from deep learning models segmentation performance.

The instrument orientation prediction performance from the OEU-Net was assessed through the root mean square error (RMSE) between the ground truth orientation and the predicted orientation on the test dataset. Table 3.7 shows the results.

The error was higher for the altitude angle than for the azimuth angle. Since the azimuth angle is the in-plane orientation of the instrument, it is more easily estimated by using the information in the US image plane. Nevertheless, the OEU-Net was able to estimate the instrument orientation relative to the US image plane maintaining an average RMSE below 6 degrees.

Considering the length of the fetoscope shaft that is equal to 25 cm and an orientation error of 5 degrees, the maximum instrument tip deviation obtained is 2.18 cm, which is enough to have the instrument tip out of the US plane. Hence, the OEU-Net does not provide a sufficiently good orientation estimation in order to model the instrument pose in the 3D space.

Table 3.7: Instrument Orientation Prediction Results - OEU-Net Model

Altitude Angle RMSE (degrees)	**Azimuth Angle RMSE (degrees)**
5.92	3.56

Table 3.8: Number of Frames US Videos

US Video 1	US Video 2	US Video 3
623	382	333

3.4.3 Tracking performance

The instrument tracking performance was evaluated using two metrics, the tip error (TE), and the computation time required to localize the instrument tip in a single image. The TE corresponds to the mean error between the ground truth instrument tip position and the instrument tip position estimated by the tracking methodology. The RMSE between these two positions was also calculated. The TE was calculated for both the x-axis and the y-axis directions from the US image frame using the following equations:

$$x-axis\ TE = \frac{\sum_{i=0}^{N}(x_i^* - \hat{x}_i)}{N} \tag{3.18}$$

$$y-axis\ TE = \frac{\sum_{i=0}^{N}(y_i^* - \hat{y}_i)}{N} \tag{3.19}$$

where N is the number of ultrasound frames in the video, $\hat{x}$ and $\hat{y}$ the estimated x and y coordinates, while x^* and y^* are the ground truth x and y coordinates for the tip position.

The developed instrument localization algorithms were applied to three different ultrasound videos (Table 3.8) for the tracking evaluation.

The tracking performance results are presented in Tables 3.9, 3.10, and 3.11. The estimated instrument tip trajectories in comparison with the ground-truth trajectory for the different tracking methods are shown in Figures 3.24, 3.25, and 3.26.

The Gabor method presented the best tracking performance, with the lowest tip errors in all US videos, being also the fastest method. The W-Net model was the slowest one, while the other neural networks had similar computation times. Since the instrument tracking must be performed in real-time with a sampling rate of 30 US frames per second, meaning 33.33 ms between each frame, the W-Net method is a poor choice for real-time tracking.

The deep learning algorithm with the lowest tip error was the OEU-Net. Furthermore, the OEU-Net is a fully automatic method, which is a great advantage for real-time tracking when compared to the Gabor method that is semi-automatic, requiring an user input. Thus, from the analyzed instrument location algorithms in 2D ultrasound images, the OEU-Net would be the preferred one.

Table 3.9: Tracking Performance - US Video 1

Method	x-axis TE (mm)	y-axis TE (mm)	RMSE (mm)	Computation Time (ms)
Gabor	**-0.79 ± 6.02**	**3.52 ± 5.84**	**6.45**	**35.27 ± 7.64**
U-Net	5.92 ± 4.21	-6.92 ± 1.65	7.19	36.35 ± 11.00
EU-Net	6.76 ± 4.06	-6.84 ± 1.64	7.47	37.15 ± 12.99
W-Net	5.95 ± 4.29	-7.10 ± 1.69	7.32	87.32 ± 15.35
OEU-Net	5.15 ± 5.18	-6.20 ± 2.32	6.97	37.70 ± 13.07

Table 3.10: Tracking Performance - US Video 2

Method	x-axis TE (mm)	y-axis TE (mm)	RMSE (mm)	Computation Time (ms)
Gabor	**2.63 ± 2.83**	**-0.63 ± 3.59**	**3.75**	**30.76 ± 5.56**
U-Net	5.77 ± 3.49	-6.25 ± 1.91	6.64	36.55 ± 13.65
EU-Net	6.31 ± 2.92	-6.26 ± 2.86	6.92	35.01 ± 13.87
W-Net	5.93 ± 2.97	-5.92 ± 2.58	6.55	85.26 ± 16.54
OEU-Net	3.85 ± 3.37	-3.68 ± 2.65	4.84	37.77 ± 10.34

Table 3.11: Tracking Performance - US Video 3

Method	x-axis TE (mm)	y-axis TE (mm)	RMSE (mm)	Computation Time (ms)
Gabor	1.71 ± 2.40	**-0.87 ± 2.99**	**3.03**	**31.10 ± 4.94**
U-Net	-1.51 ± 1.95	-4.68 ± 0.54	3.76	36.52 ± 14.13
EU-Net	-0.95 ± 2.30	-4.85 ± 0.47	3.87	36.04 ± 14.03
W-Net	-1.30 ± 1.82	-4.53 ± 0.46	3.59	86.83 ± 20.64
OEU-Net	**0.82 ± 2.36**	-4.97 ± 0.45	3.94	36.85 ± 9.41

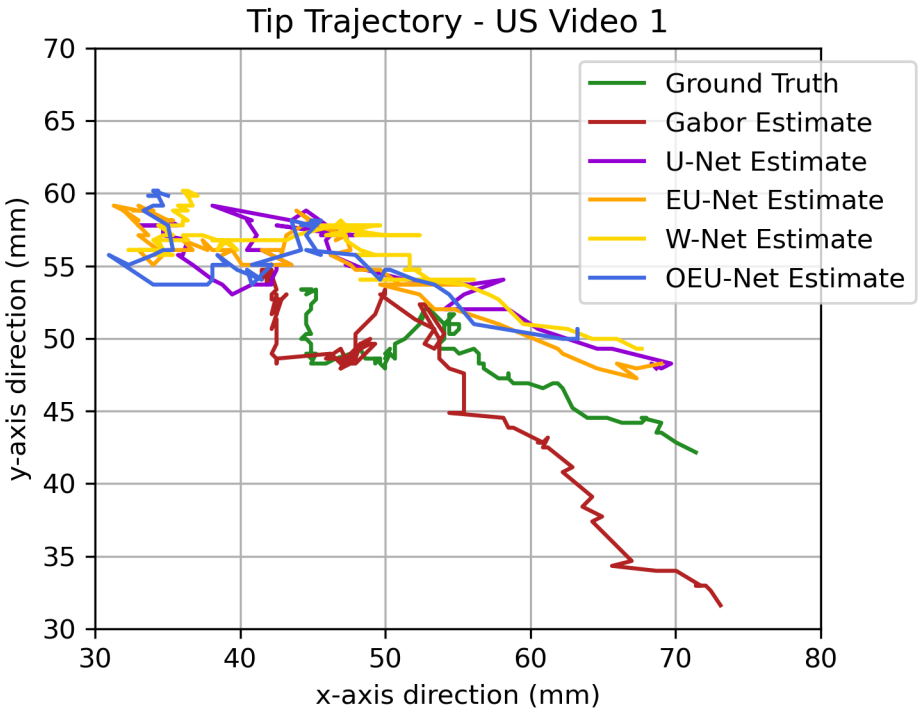


Figure 3.24: Estimated instrument tip trajectories compared to the ground truth trajectory for US video 1.

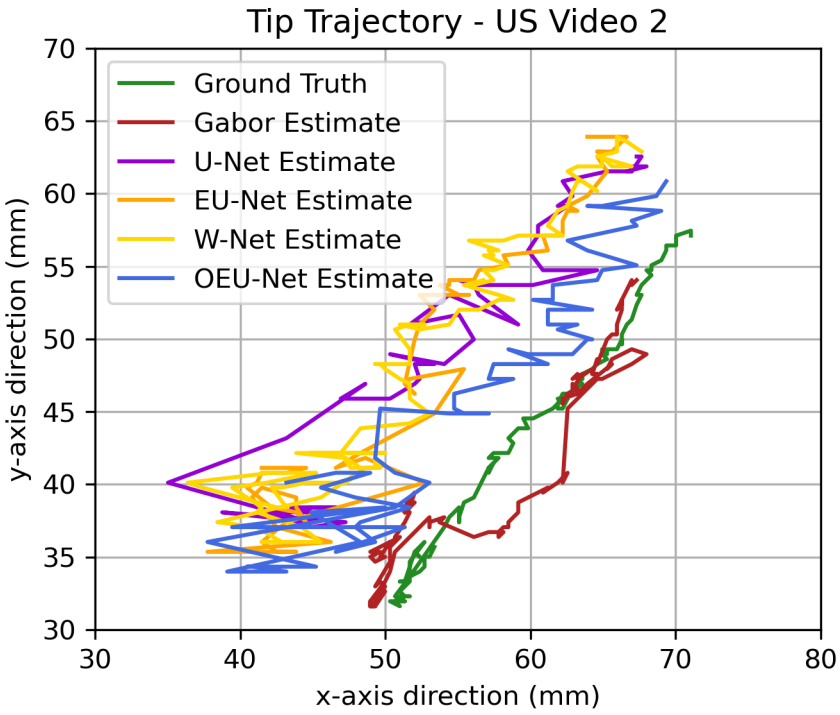


Figure 3.25: Estimated instrument tip trajectories compared to the ground truth trajectory for US video 2.

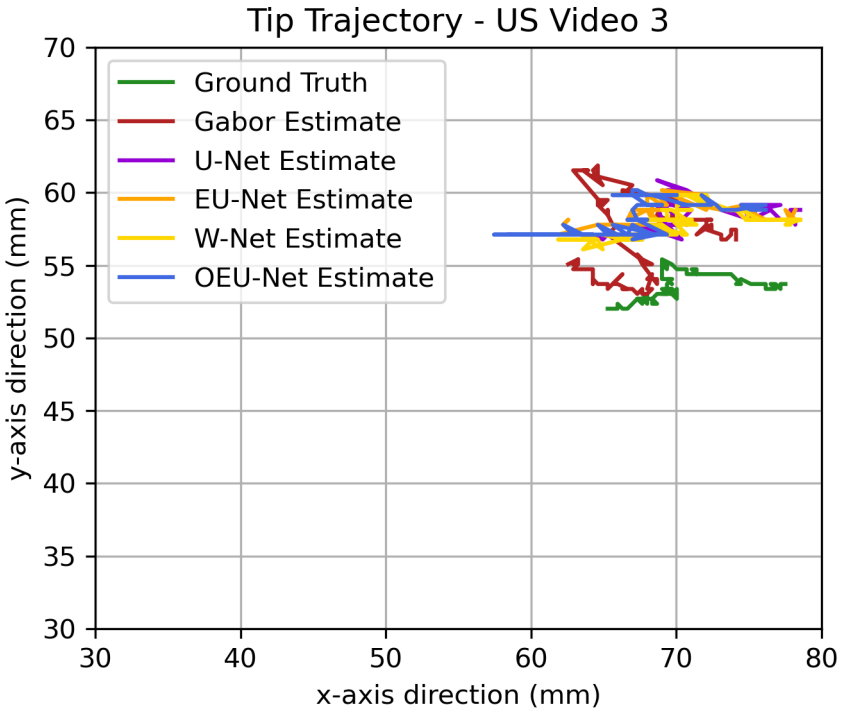


Figure 3.26: Estimated instrument tip trajectories compared to the ground truth trajectory for US video 3.

The tip trajectories estimated by the deep learning methods are close to one another, while being at an almost constant distance from the ground truth, indicating the presence of a systematic error. Moreover, in case the tip error had only a random component, its average would be close to zero as in the Gabor method, but all the deep learning methods have an average tip error that is deviated from 0. Thus, a Principal Component Analysis (PCA) is performed on the tip error distribution in order to further analyze this systematic error.

Since all the deep learning methods have a similar distribution, and the OEU-Net would be the preferred method, the PCA is discussed only for the OEU-Net algorithm.

The PCA started by calculating the error between the ground truth tip position (x^*, y^*) and the estimated tip position $(\hat{x}, \hat{y})$ obtained from the OEU-Net localization algorithm in the three US videos used to test the algorithm. The error is simply the difference between the ground truth and the estimated tip position:

$$x - axis\ error = x^* - \hat{x} \tag{3.20}$$

$$y - axis\ error = y^* - \hat{y} \tag{3.21}$$

Then, the error is standardized by subtracting the mean and dividing it by the standard deviation. Next, the PCA is applied on the standardized error distribution. The loadings of the PCA

are vectors that represent the direction in which the distribution is projected in order to produce the scores. The loading of the first principal component (PC) is also the direction that explains the highest amount of variance in the error distribution. The loading of the second principal component is orthogonal to the loading of the first PC and explains the second highest amount of variance.

The standardized error distributions are plotted with the PC's loading direction, the instrument orientation, and the trajectory orientation (see Figs. 3.27, 3.28, and 3.29). These plots give some insight on whether the systematic error is related to the instrument orientation or to the trajectory orientation in the ultrasound videos used for testing.

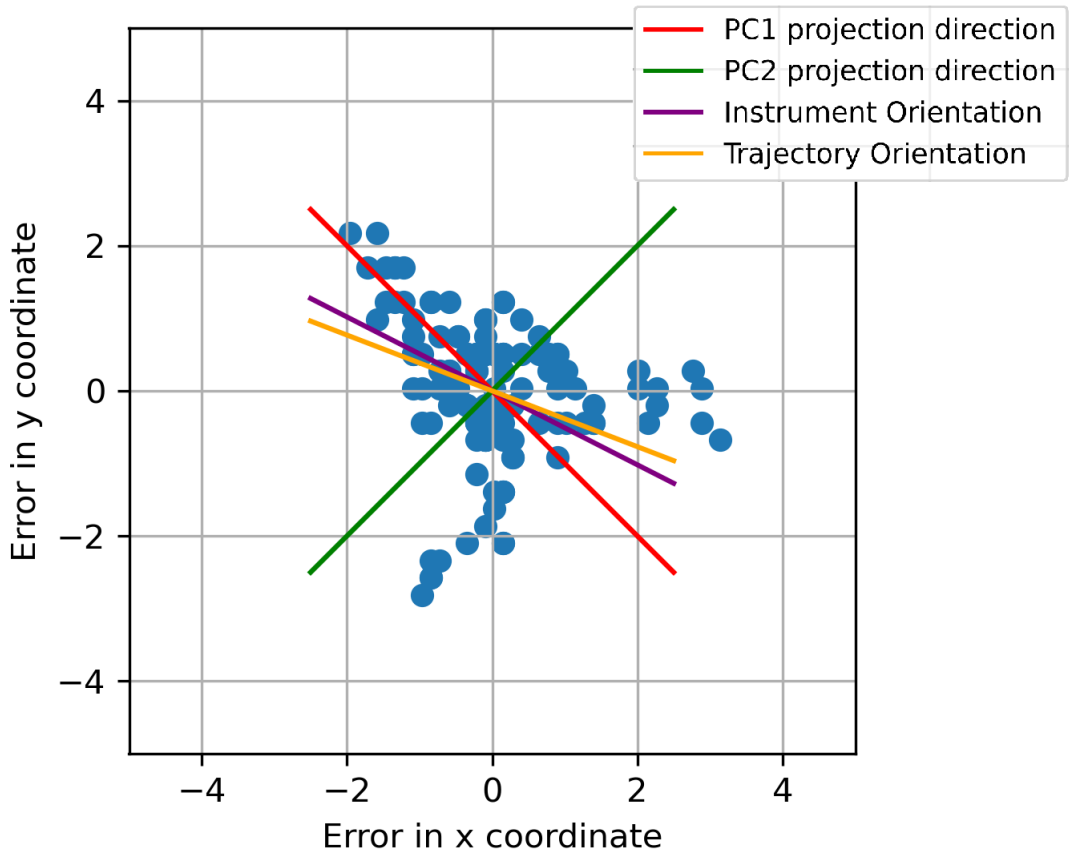
For US video 1 (see Fig. 3.27), both the instrument and trajectory orientation are similar and the first PC loading direction is close to both orientations. The first PC in this US video explains around 60% of the total variance, thus, both instrument trajectory and orientation may impact the error distribution.

Observing the PCA on US video 2 (see Fig. 3.28), the instrument orientation is almost orthogonal to the trajectory orientation. The first PC loading direction, which explains around 70% of the variance, is really close to the trajectory orientation. The second PC loading direction, which explains around 30% of the error variance, is close to the instrument orientation. Again, both the instrument and trajectory orientation seem to impact the error distribution, but in this case, the trajectory orientation has more influence on the distribution.

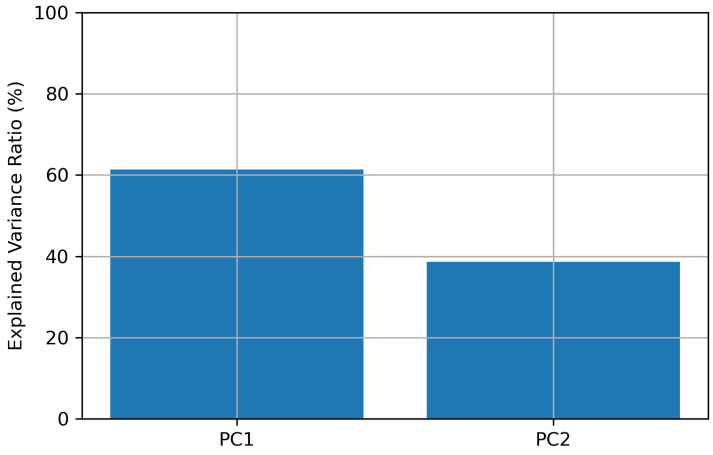
Finally, the PCA on US video 3 (see Fig. 3.29) shows that the instrument orientation has much more influence on the error distribution than the trajectory. The first PC loading direction is really close to the instrument orientation, while the trajectory orientation is distant from both loadings directions.

In conclusion, both the trajectory orientation and instrument orientation have an impact on the error distribution during the fetoscope tip tracking. However, the instrument orientation has a higher overall impact on the systematic error.

The systematic error can be explained by observing the OEU-Net segmentation output, which is similar to the other deep learning models outputs, in comparison with the ground truth segmentation mask (see Fig. 3.30). The deep learning model is not able to properly segment the ending of the instrument, which generates a deviation in the estimated tip position. This deviation is the source of the systematic error in the instrument orientation direction.

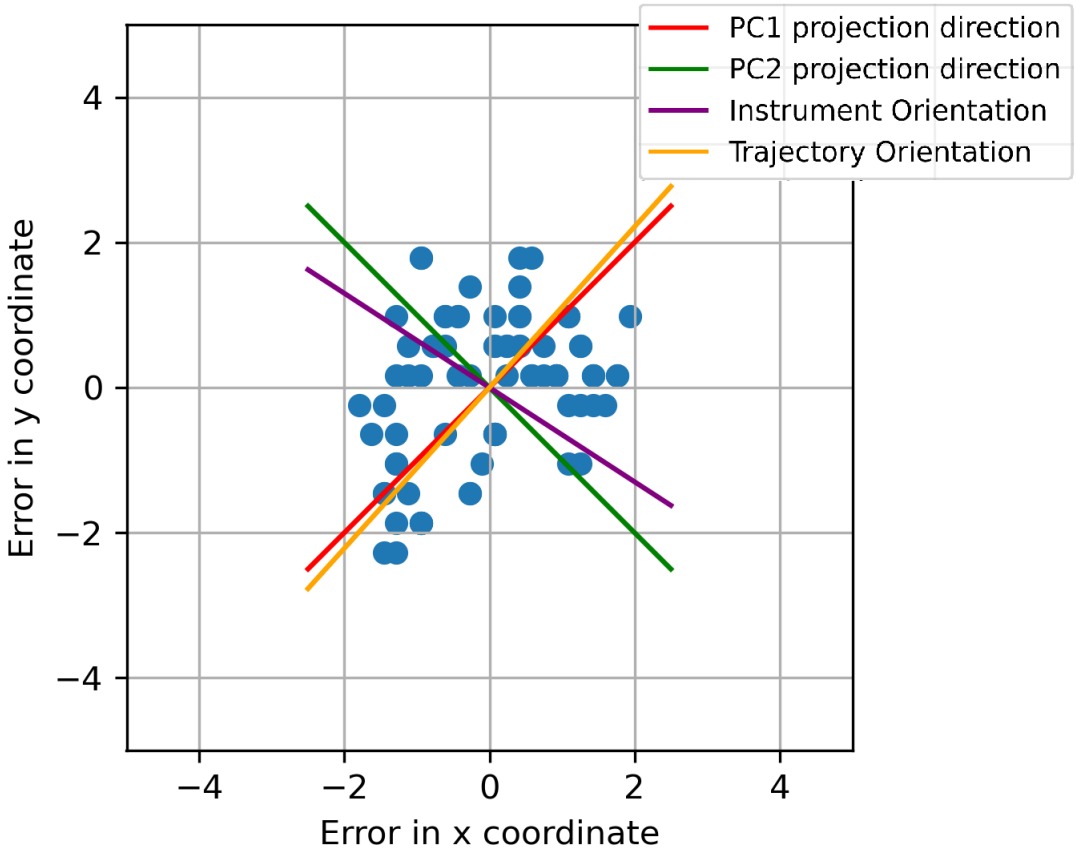


(a) Error distribution, PC directions, instrument orientation, and trajectory orientation

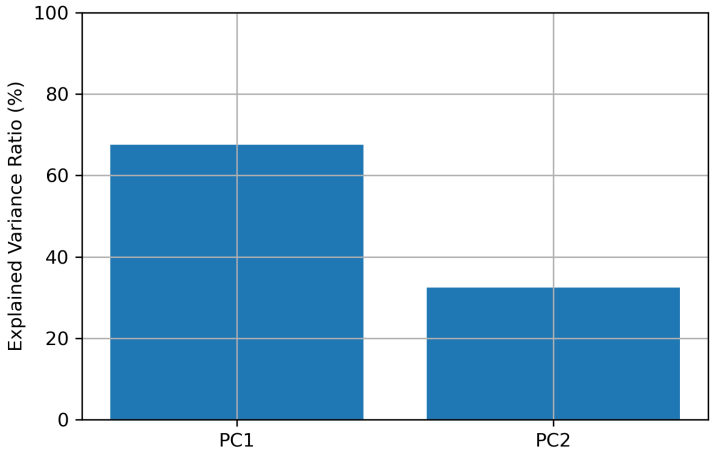


(b) Explained variance ratio

Figure 3.27: Principal Component Analysis for the error distribution from the OEU-Net method in US video 1.

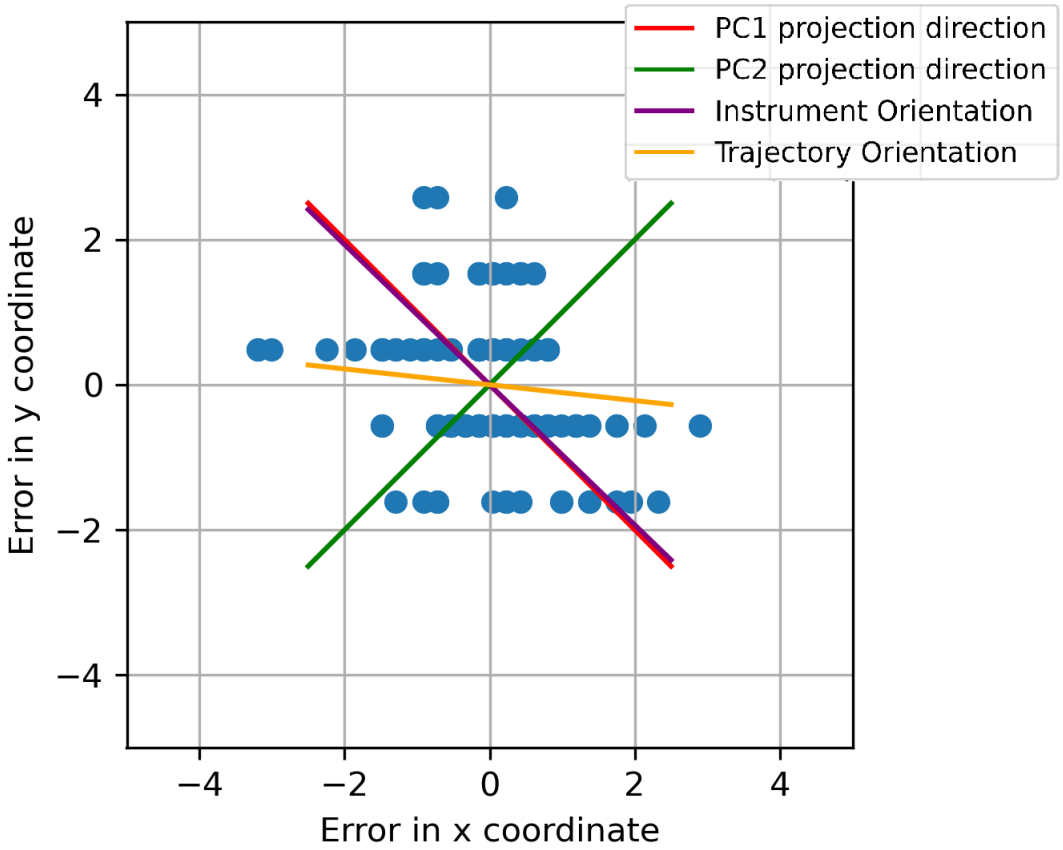


(a) Error distribution, PC directions, instrument orientation, and trajectory orientation

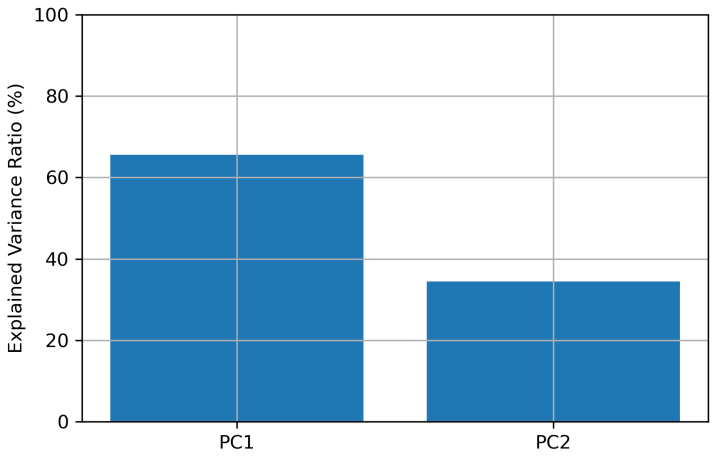


(b) Explained variance ratio

Figure 3.28: Principal Component Analysis for the error distribution from the OEU-Net method in US video 2.

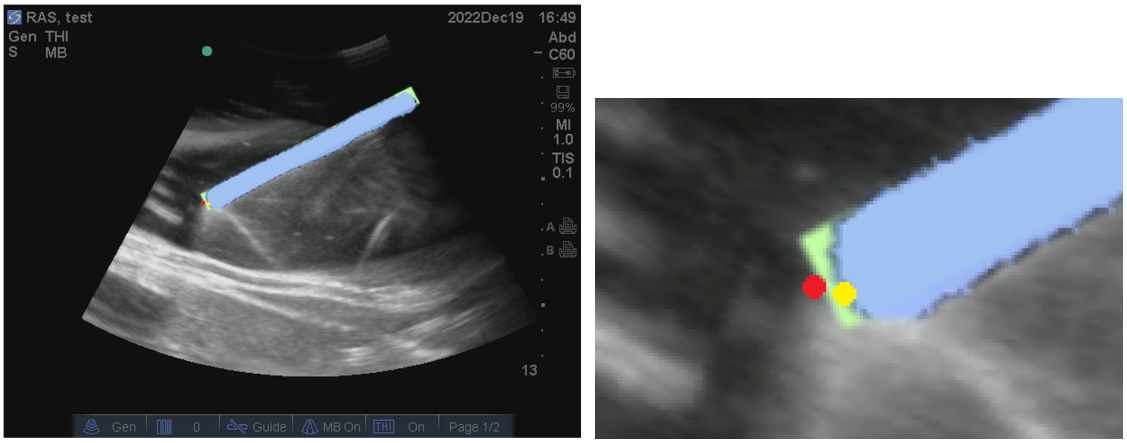


(a) Error distribution, PC directions, instrument orientation, and trajectory orientation



(b) Explained variance ratio

Figure 3.29: Principal Component Analysis for the error distribution from the OEU-Net method in US video 3.



(a) Normal view (b) Zoomed view

Figure 3.30: Ultrasound image from test dataset overlayed with the (green) ground-truth segmentation mask , the (blue) OEU-Net model segmentation output, (red) the ground-truth tip position, and (yellow) estimated tip position.

Chapter 4

Robotic Ultrasound-Based Instrument Tracking Framework

This chapter describes the development of an ultrasound-based instrument tracking framework for fetoscope tracking during FETO. The framework is based on the OEU-Net localization method described in chapter 3 and on a 6-DoF robot for autonomous robotic ultrasound imaging tracking. By using the information given by the OEU-Net method, which is processing the acquired US images, the robot performs an automatic tracking of the fetoscope.

The chapter starts with a presentation of the experimental setup, including a description of the robot and the software used to control it. Next, the complete ultrasound-based tracking framework is described. Finally, the results of the instrument tracking performance are reported and discussed.

4.1 Experimental setup

The experimental setup regarding the ultrasound image acquisition, optical tracking system, fetoscope, and amniotic cavity phantom corresponds to exactly the same setup detailed in section 3.1. There is only an additional component which is a 6-DoF *Haption Virtuose 6D RV* robotic arm (HAPTION SA, 2023) (see Fig. 4.1). The US probe is the end-effector of the robot which attached to the robot wrist.

The data produced by the US machine is managed by the same interface used in the experimental setup presented in the previous chapter (Section 3.1). Thus, the ROS middleware manages the data flow, while running in a computer with Ubuntu 20.04 operative system.

The robot also has to communicate with the computer. Hence, three separated subsystems are defined in order to better organize and comprehend the experimental setup (see Fig. 4.2). The first system corresponds to the vision system which is composed by the US machine, providing information regarding vision of the working environment. The robotic system is defined by the

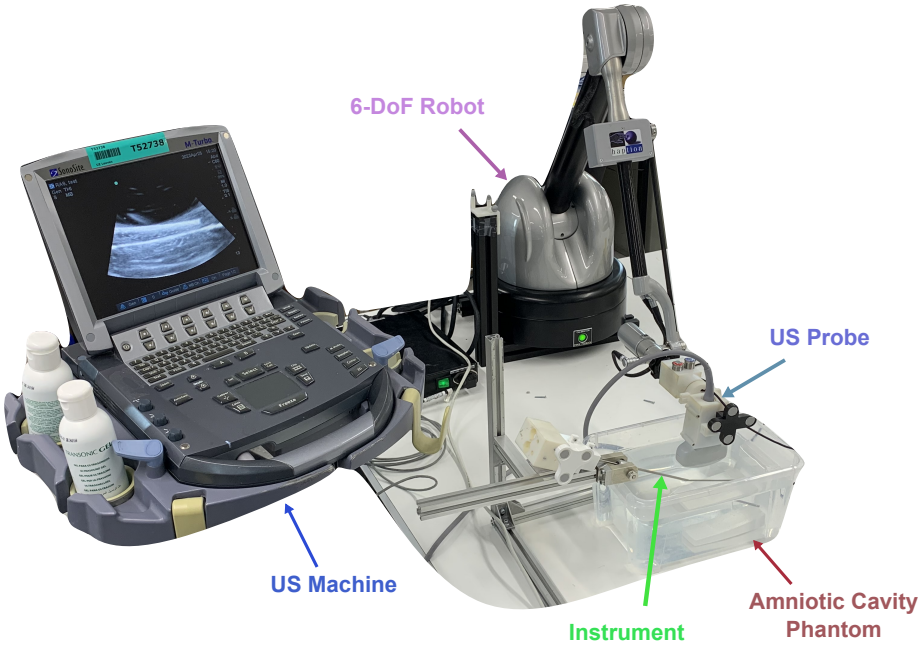


Figure 4.1: Experimental setup for ultrasound-based fetoscope tracking

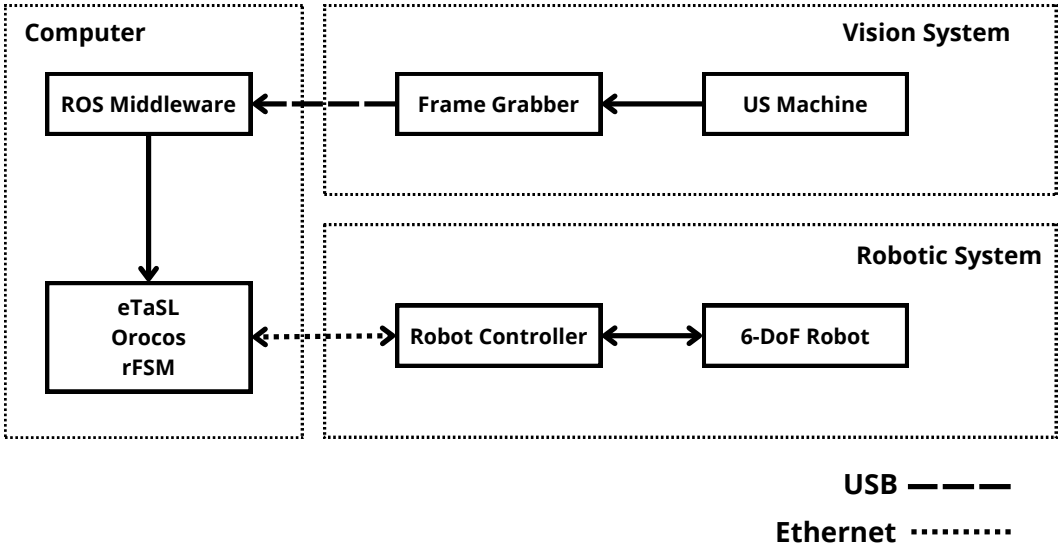


Figure 4.2: Subsystems of the experimental setup: vision system, robotic system, and computer

6-DoF Robotic Arm and its controller. Finally, the third system is the computer, which contains the software programs needed to process all the data and control the robotic arm.

The libraries/middleware used to control the robotic arm and process the US images are ROS (Open Robotics, 2022), expressiongraph-based Task Specification language (eTaSL) (Aertbeliën and De Schutter, 2014), Open robot control software (Orocos) (Soetens et al., 2020), and real-time finite state machine (rFSM) (Klotzbuecher, 2013). The following sections detail how these softwares were used in the ultrasound-based instrument tracking system.

4.1.1 Libraries and middleware

4.1.1.1 ROS

By using the *usb_cam* package (Open Robotics, 2022), the US frames transmitted by the frame grabber to the computer are published to a ROS topic called '\image_raw'. The ROS message for this topic is an Image message.

A topic named 'image_classifier' receives the raw images from the topic '\image_raw' and classifies them into two classes: fetoscope visible in US image or not. A neural network is used as the classifier model to perform this classification. The neural network is explained in section 4.1.1.2.

The OEU-Net localization method from section 3.3.2 is implemented by means of two ROS nodes. The node 'image_segmentation' receives the image data from the topic '\image_raw' and proceeds to perform the instrument segmentation by using the OEU-Net model. The OEU-Net model outputs a segmentation mask containing the instrument location. The image containing this segmentation mask is published to the topic '\segmented_image'.

Since OEU-Net cannot provide an accurate instrument orientation relative to the US image plane (Section 3.4.2), the area of the segmented mask is used to determine if the instrument is parallel or not to the US image plane. The area of the segmented mask correspond to the number of pixels in the mask. If this area is higher than 2500 pixels, the instrument is considered to be parallel with the image plane. In case the area is between 300 and 2500 pixels, the instrument is tangent to the image plane. If the area is below 300 pixels, the instrument is not visible in the ultrasound image.

In case the instrument is aligned and parallel to the image plane, the post-processing of the OEU-Net method to estimate the instrument tip position can be performed. This post-processing is performed by the ROS node 'tip_tracking', which subscribes to the '\segmented_image' topic, performs the localization of the tip, and publishes the average of the last 20 tip position values in the x-axis of the US image frame (see Fig. 4.3) to a topic denominated '\instrument_tip'. The ROS message used in this topic was Float32. The tip position is equal to zero if it is located in the center of the image in the horizontal direction (x-axis).

If the instrument is tangent to the image plane, the centroid of the segmented instrument mask is calculated. The centroid is simply the arithmetic mean of the pixel positions from the segmented mask. This centroid is calculated by the node 'align_instrument', which receives the image from

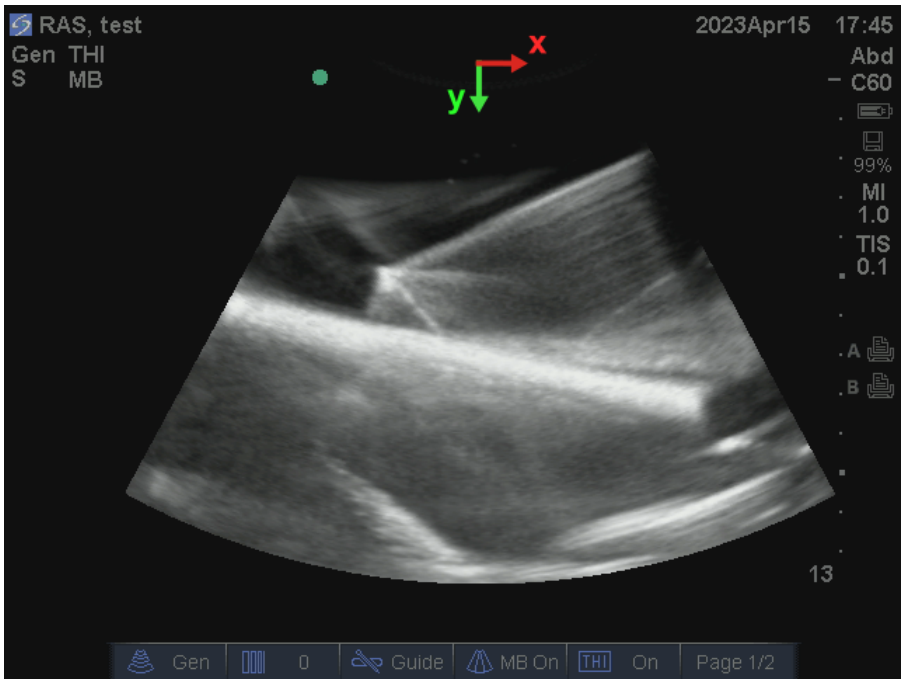


Figure 4.3: Ultrasound image frame: (red) x-axis and (green) y-axis

'\segmented_image' topic and publishes the centroid position in the x-axis of the US image frame (see Fig. 4.3) to the topic '\instrument_centroid'. The ROS message of this topic was Float32.

The nodes 'image_classifier' and 'image_segmentation' can publish string ROS messages to the topic '\events'. The types of events that could be sent to this topic are described in Table 4.1.

Figure 4.4 shows a summary of the different ROS topics and nodes used in this work.

4.1.1.2 Image classification neural network

A pre-trained neural network ResNet-50 (He et al., 2015) was trained on ultrasound images which were separated into two classes: (class 1) contain instrument or (class 2) does not contain instrument. Thus, the model should learn how to classify the images into those two classes.

The training followed a cross-validation with random subsampling, where 80% of the training dataset was used for training, and 20% for validation. The training dataset had 1469 images of class 1 and 1147 of class 2. The loss used to train the classifier was the binary cross entropy, and

Table 4.1: Type of events published to ROS topic '\events'

Event Name	Publisher Name	Event Description
e_found	'image_classifier'	Fetoscope is visible in image
e_not_found	'image_classifier'	Fetoscope is not visible in image
e_found_aligned	'image_segmentation'	Segmented fetoscope area > 2500 pixels
e_found_not_aligned	'image_segmentation'	300 pixels < Segmented fetoscope area < 2500 pixels
e_lost	'image_segmentation'	Segmented fetoscope area < 300 pixels

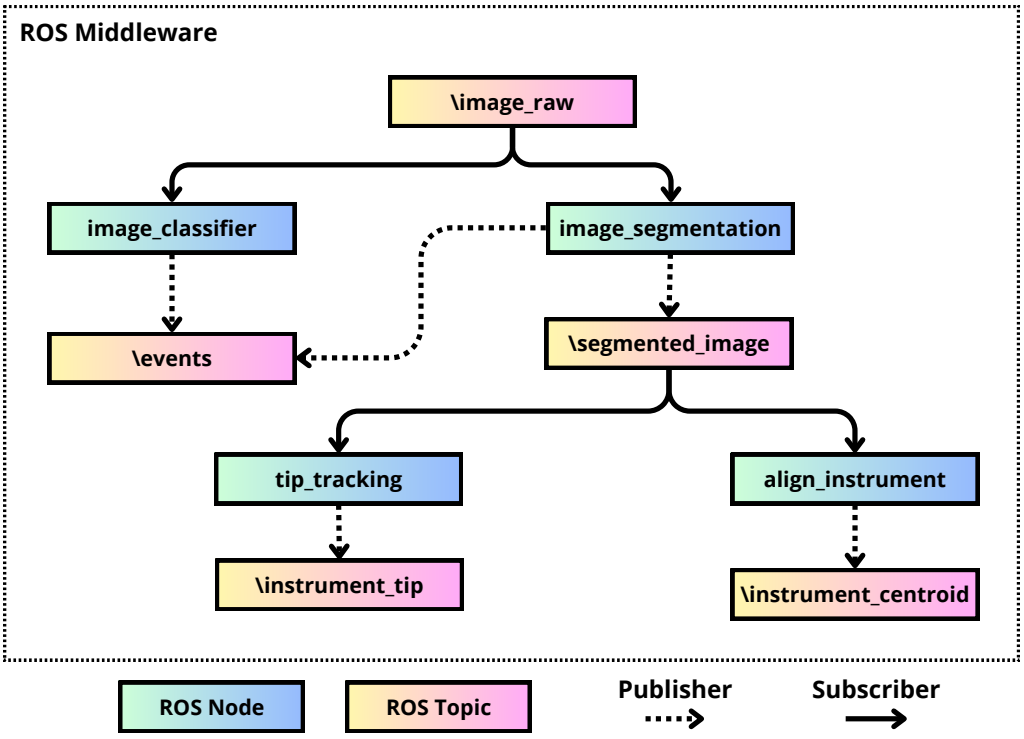


Figure 4.4: Communication scheme between ROS nodes and ROS topics

a stochastic gradient descent optimizer was applied to update the model weights with an initial learning rate equal to 0.001. The learning rate was multiplied by a factor of 0.1 every 5 epochs. 25 epochs were used to train the model.

The ResNet-50 model was able to achieve a 98.21% accuracy for the classification on the test dataset, which contained 251 images of class 1 and 252 images of class 2.

4.1.1.3 eTaSL

eTaSL is a Task Specification Language for reactive control of robotic systems implemented in Lua (Aertbeliën and De Schutter, 2014). Since the robot must be inserted in a dynamic scenario which is a surgical room, the reactive control applied by eTaSL allows for robustness against environment variations and can deal with human interactions. At each sample time, eTaSL generates an appropriate control signal based on the sensors, optimization problems solutions, kinematic model, and defined task.

The architecture of eTaSL follows a modular approach and is presented in Figure 4.5.

The task specification is captured by a C++ data structure denominated 'context'. eTaSL enables a human-friendly specification of the task in Lua, which is then translated into the context. eTaSL provides a controller called eTC. This controller provides a solver that translates the context for a numerical solver. The solver uses a velocity-resolved instantaneous optimization problem approach to generate a numerically specified optimization problem to the numerical solver.

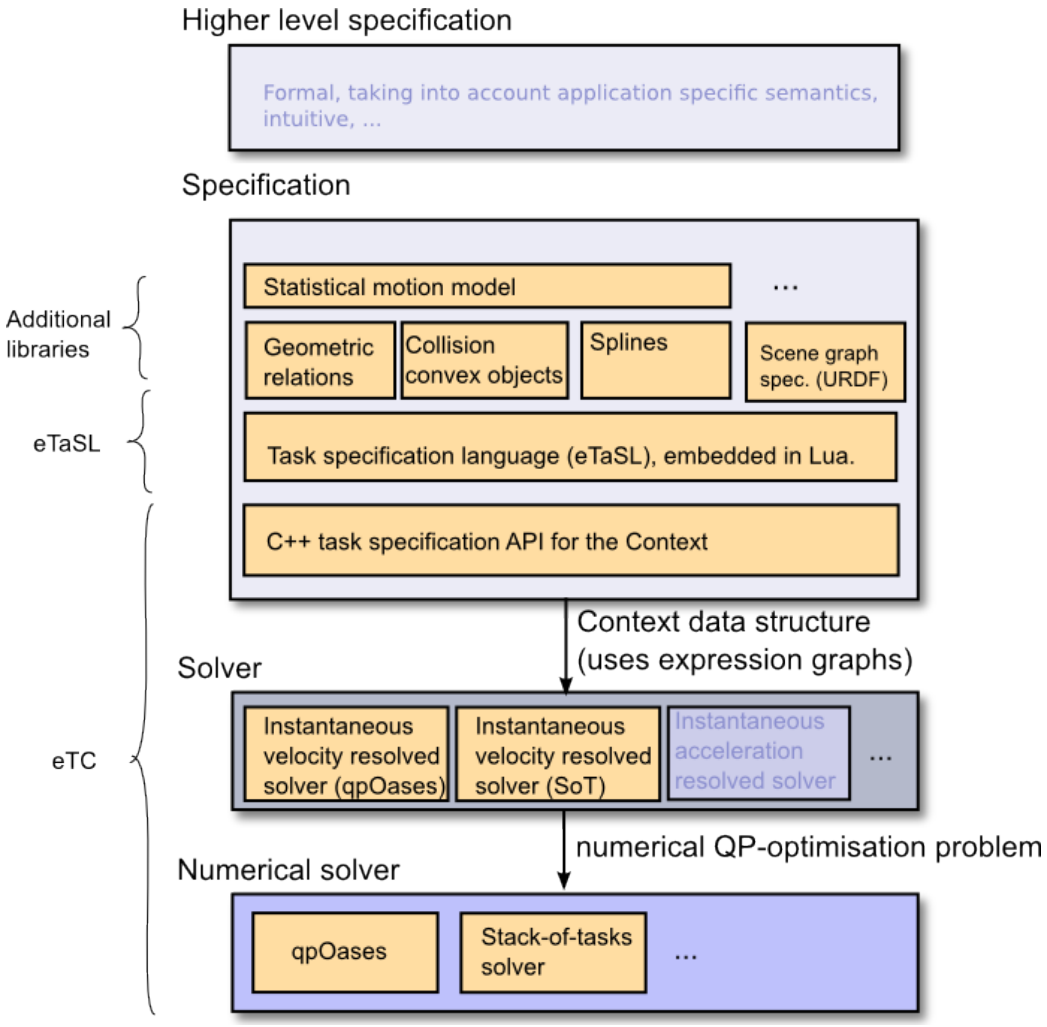


Figure 4.5: The architecture of eTaSL (Aertbeliën, 2020).

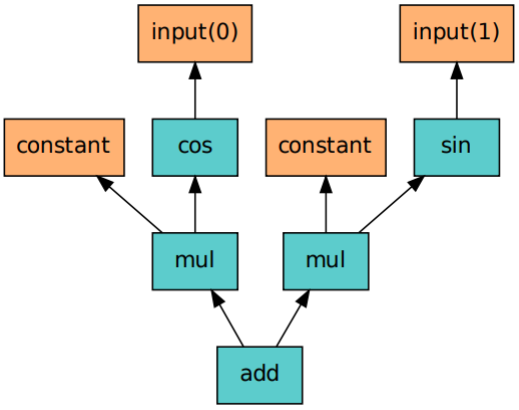


Figure 4.6: eTaSL expression graph example (Aertbeliën, 2020).

eTaSL uses expression graphs to symbolically represent an expression. These expression graphs are used to represent position-level expressions, such as transformation frames, vectors, or rotations matrices. An eTaSL expression graph representing the kinematic chain of a robot can be easily obtained from a Unified Robot Description Format (URDF) file for the specific robot. The transformation matrices between any frame of the robot are easily extracted from the expression built from the URDF file. Figure 4.6 shows an eTaSL expression graph for the expression "$a \cdot \cos(x) + b \cdot \cos(y)$", where a and b are constants, while x and y are the inputs.

The URDF file containing the complete description of the geometry and links of the 6-DoF robot was used to create the eTaSL expression representing the kinematic chain of the robot, including the US probe frame. Figure 4.7 presents the robot model running in RViz. RViz is a 3D visualizer for ROS (Open Robotics, 2022).

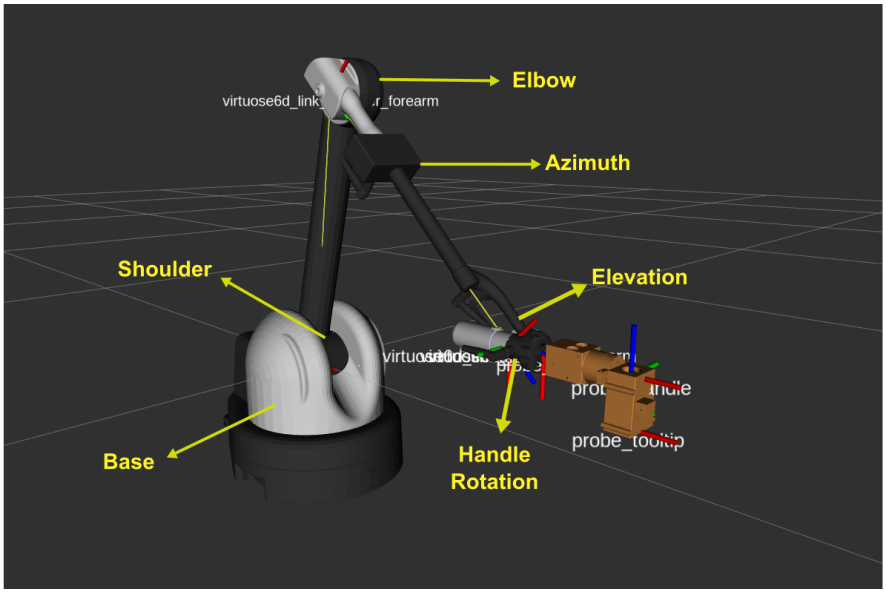


Figure 4.7: 6-DoF robotic arm model in RViz with robot joint names.

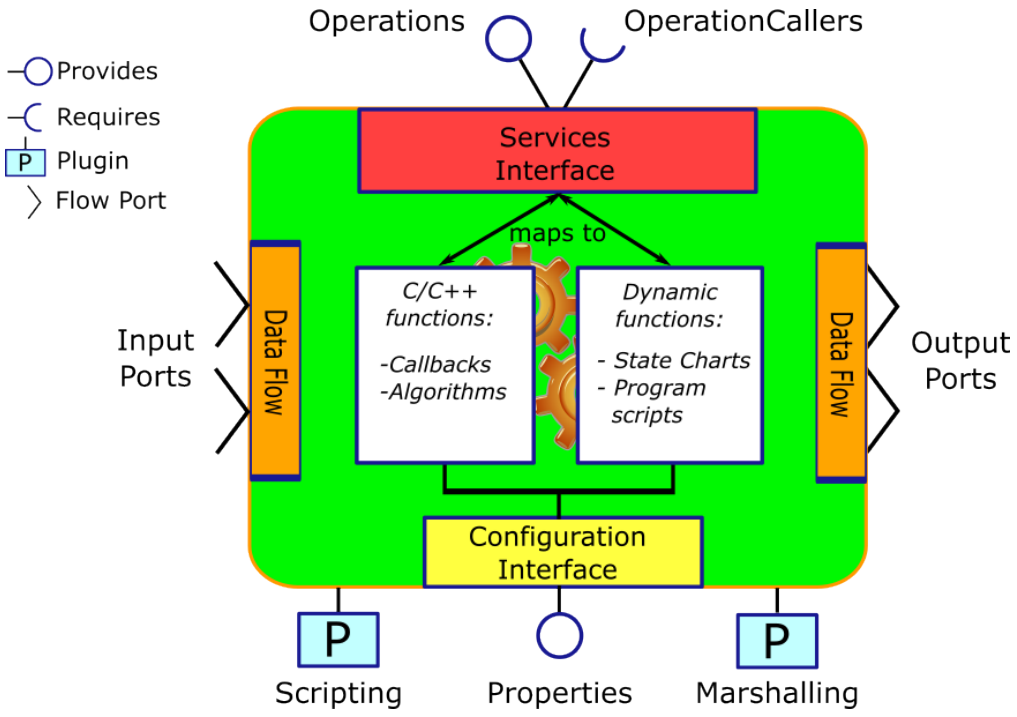


Figure 4.8: Schematic overview of a TaskContext (Soetens et al., 2020).

The tasks performed by the robot were performed with a trapezoidal motion profile. Using some pre-defined parameters eTaSL automatically computes the joint velocities and send the control signals to the robot controller.

4.1.1.4 Orocos

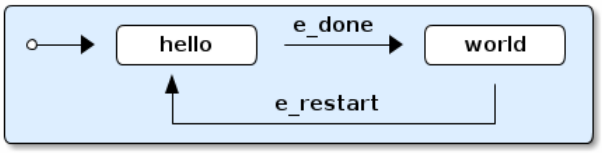
Orocos is composed by different libraries and toolkits. The main parts are the Orocos Real-Time Toolkit (RTT) and the Orocos Component Library (OCL). RTT allows the writing of real-time C++ components, while OCL gives the necessary tools to start an application and interact with it in real-time. Thus, Orocos allows the building of real-time software components (Soetens et al., 2020).

The components in Orocos are instances of a C++ class denominated TaskContext and its interface consists in attributes, properties, operations, and data flow ports (see Fig. 4.8).

The eTaSL RTT component offers the eTaSL functionalities to an Orocos environment. How and what it communicates with other Orocos components is flexibly configured (Aertbeliën, 2020).

Orocos components were build and used to transmit the data produced in ROS (section 4.1.1.1) to eTaSL. Thus the events, instrument tip position, or instrument centroid can be used to specify the tasks in eTaSL for controlling the robotic system.

Orocos was also used to transmit the events generated by ROS to rFSM. This communication allowed the transition between robot tasks to be dependent of the US image processing performed in ROS.




```
1:  return rfsm.state {
2:     hello = rfsm.state { exit=function() print("hello") end },
3:     world = rfsm.state { entry=function() print("world") end },
4:
5:     rfsm.transition { src='initial', tgt='hello' },
6:     rfsm.transition { src='hello', tgt='world', events={ 'e_done' } },
7:     rfsm.transition { src='world', tgt='hello', events={ 'e_restart' } },
8:  }
```

Figure 4.9: Example of simple finite state machine using rFSM (Klotzbuecher, 2013).

4.1.1.5 rFSM

eTaSL is not able to generate discrete-event switching or sequencing of robot tasks (Aertbeliën, 2020). Hence, rFSM is used for this purpose, enabling the development of a finite state machine based robot control.

rFSM is a Statechart implementation, written in pure Lua and mainly designed for coordination of complex systems (Klotzbuecher, 2013). Some of the rFSM features are:

- Hierarchical states
- Completion events
- Parametrizable and reusable states
- Easy to build statemachines by composing existing states/state machines

Figure 4.9 exemplifies the implementation of a simple finite state machine with only two states and two events for transition between states.

rFSM was used in this work to build the finite state machine used to operate the 6-DoF robot for fetoscope tracking.

4.2 Ultrasound-Based instrument tracking framework

The instrument tracking framework is based on the information that is retrieved from processing the ultrasound image that is being gathered in real-time by the US probe, which is positioned by the 6-DoF robot.

The ultrasound images are processed by ROS. Then, ROS generates a certain event or a number depending on the image processing output. The generated events which are described in Table 4.1 are sent to rFSM in order to perform state transitions in the finite state machine defined to operate the robot. These state transitions cause a change in the robot task that needs to be performed.

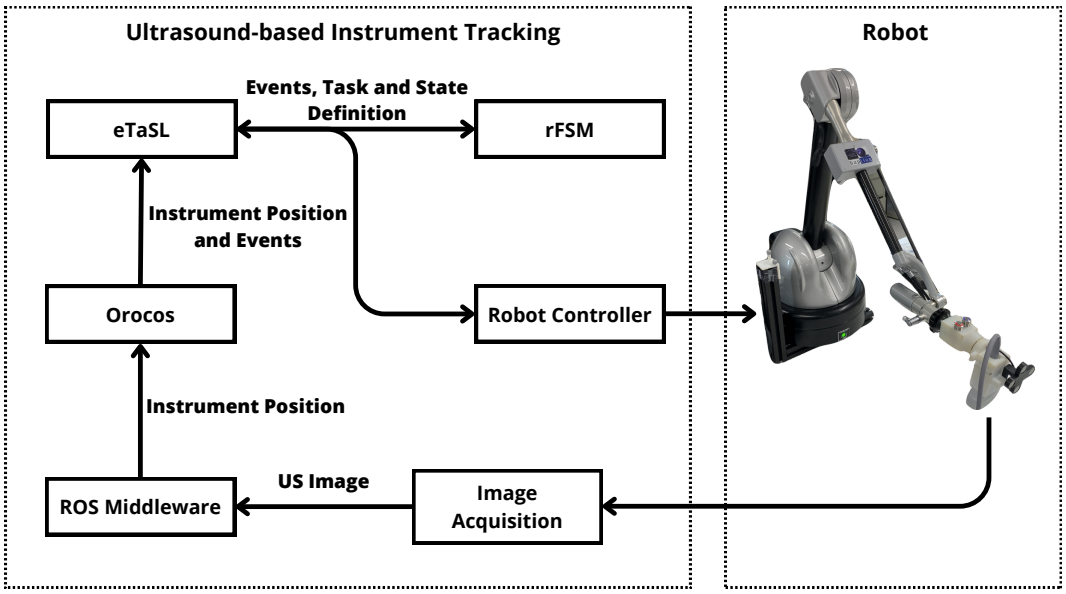


Figure 4.10: Schematic view of the ultrasound-based instrument tracking system

The numbers that can be produced by ROS are the instrument tip position or centroid position. The instrument tracking is performed by moving the US probe based on these positions. All the motion profiles defined for the tracking system have a maximum velocity equal to 0.006 m/s and maximum acceleration equal to 0.003 m/s^2.

Figure 4.10 presents a simple schematic representation of the tracking system.

4.2.1 Finite state machine description

The main component of the robot control and tracking system is the finite state machine (see Fig. 4.11). This state machine defines when and which robot tasks must be performed. The events that are present in Table 4.1 can only cause a switching between states in case the same event is produced 30 times in a row. Each state is described in the next sections.

4.2.1.1 State *configured* and *idle*

The first two states are used to load necessary files such the robot expression definition, the data ports, and Orocos components. The robot is not moved in this phase, it serves to test if the files can be correctly loaded without any errors. The *configured* state automatically transitions to the *idle* state, which transitions to the *wait* state.

4.2.1.2 State *wait*

In the *wait* state, the robot is set to not move. All the states have a connection with this state through the *e_stop* event. In case the operator needs to stop any robot task, the event *e_stop* is used to stop the robot and the state is transitioned to the *wait* state.

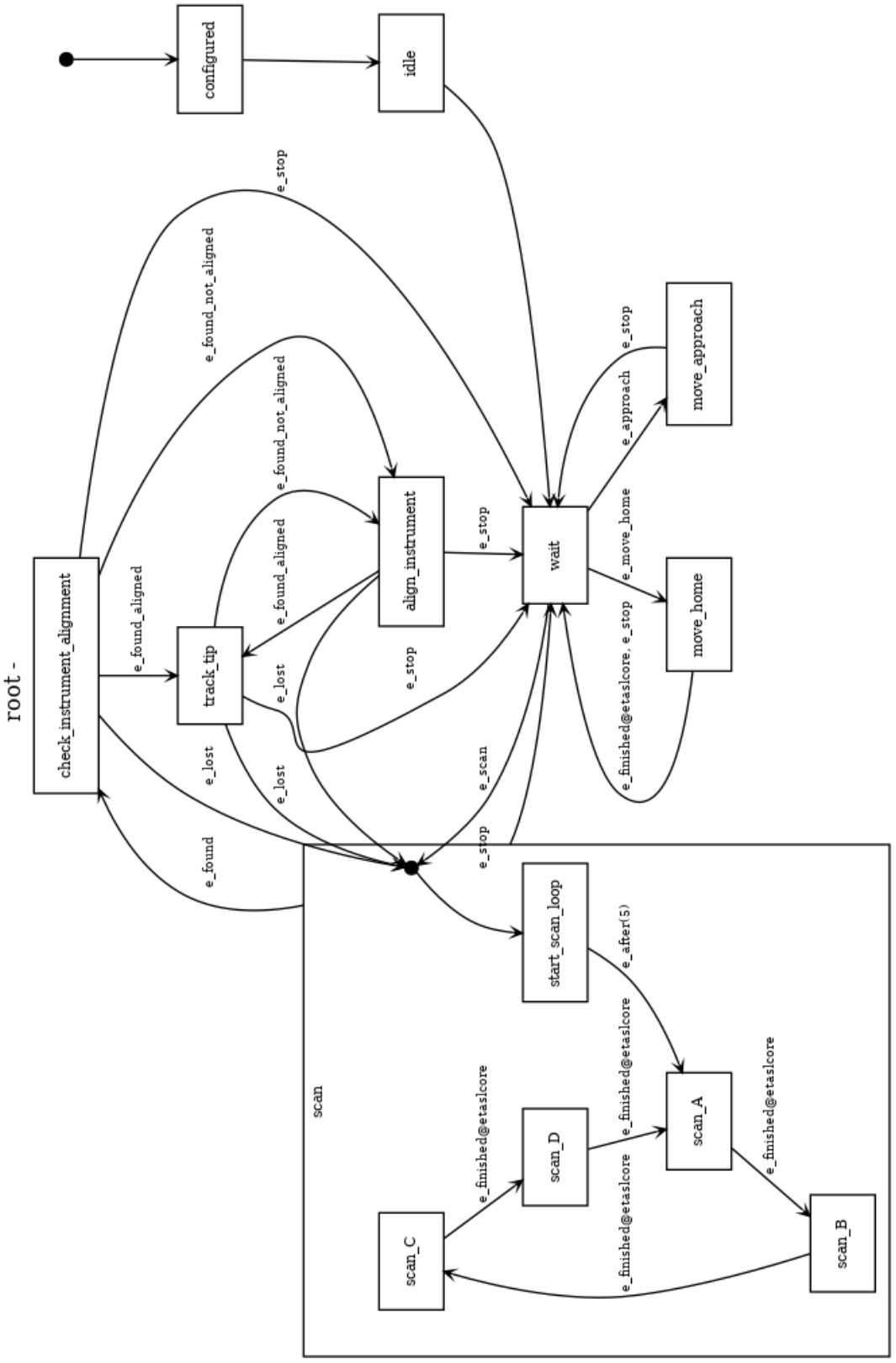


Figure 4.11: Finite State Machine States and Transitions for the Ultrasound-Based Instrument Tracking System

Table 4.2: Graphical User Interface Options, Generated Events, and State Transitions

Option	Event	Source State	Target State
Move Home	*e_move_home*	*wait*	*move_home*
Approach	*e_approach*	*wait*	*move_approach*
Start Scan	*e_scan*	*wait*	*scan*
Stop	*e_stop*	any state	*wait*

The robot operator can make some state transitions by generating an event through the options available in a graphical user interface. The options with the respective event generated and the state transitions involved are shown in Table 4.2.

4.2.1.3 State *move_home*

The home position of the robot is defined in the joint space. Each joint receives a specific value, then eTaSL uses the forward kinematic equations obtained from the URDF file to define a trapezoidal motion profile for each joint. Then, the control signals are sent to the robot.

Table 4.3 presents the values used for each robot joint.

4.2.1.4 State *move_approach*

This state is used to move the US probe downwards. The robot task is defined in the task space by using the end-effector frame. The initial pose of the end-effector frame is the current pose, the final pose corresponds to a frame that presents a displacement of -0.26m in the z-axis direction of the world frame. The z-axis of the world frame is orthogonal to the table where the experimental setup is placed (see Fig. 4.1). The rotation of the frame is kept the same. Then, eTaSL generates the necessary trapezoidal motion profiles for each joint.

4.2.1.5 State *scan*

The *scan* state has several substates that are in loop and only stops if the event *e_stop* or the event *e_found* are generated. The *scan* state is the one that starts the autonomous fetoscope tracking framework.

Table 4.3: Robot joint values for home position defined in joint space

Joint Name	Value (degrees)
Base	0
Shoulder	-75
Elbow	120
Azimuth	0
Elevation	45
Handle Rotation	0

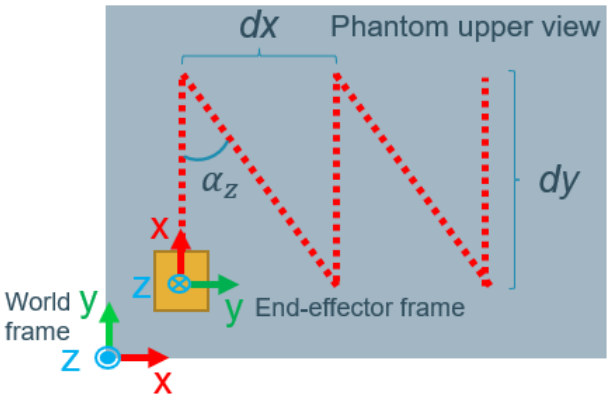


Figure 4.12: US probe scanning routine path

The first substate is the *start_scan_loop*, which is used to start the US images acquisition by the ROS middleware. There is a delay of 5 seconds before starting the US probe scan routine in order to guarantee that the images are being acquired. After the delay, the scanning loop starts.

The scan loop is used to systematically move the US probe until the fetoscope is found inside the amniotic cavity phantom. The scan routine consists in four substates that contain robot tasks defined in the task space. The robot tasks are based on the end-effector pose and the objective is to produce a Z-shaped path (see Fig. 4.12).

The first substate of the scan loop is *scan_A*. In this state the end-effector is moved dx meters in the x-axis direction of the world frame (see Fig. 4.12). Next, state *scan_B* defines a rotation around the z-axis with a specific angle of α_z radians. State *scan_C* produces a translation movement in both x and y axes where the end-effector must move $-dx$ meters in the x-axis and dy meters in the y-axis. Finally, state *scan_D* produces a rotation of $-\alpha_z$ radians around the z-axis in order to recover the original orientation and start the loop again. The transitions between states is automatically done by the event *e_finished@etaslcore*, meaning that as soon as a substate finishes, the next one starts.

The value dx was set to 0.04 m, dy to 0.02 m, and α_z to $\frac{\pi}{8}$ radians.

4.2.1.6 State *check_instrument_alignment*

The system stays in this state without generating any control signals to the robot until an event is received from ROS (Table 4.1). Upon receiving the *e_lost event*, the system transitions to the *scan* state. In case the *e_found_aligned* event is received, the state switches to track_tip, while if the event *e_found_aligned* is transmitted, the state switches to *align_instrument*.

4.2.1.7 State *align_instrument*

In the *align_instrument* state, the probe is simply rotated around its z-axis in order to align the US image plane with the fetoscope.

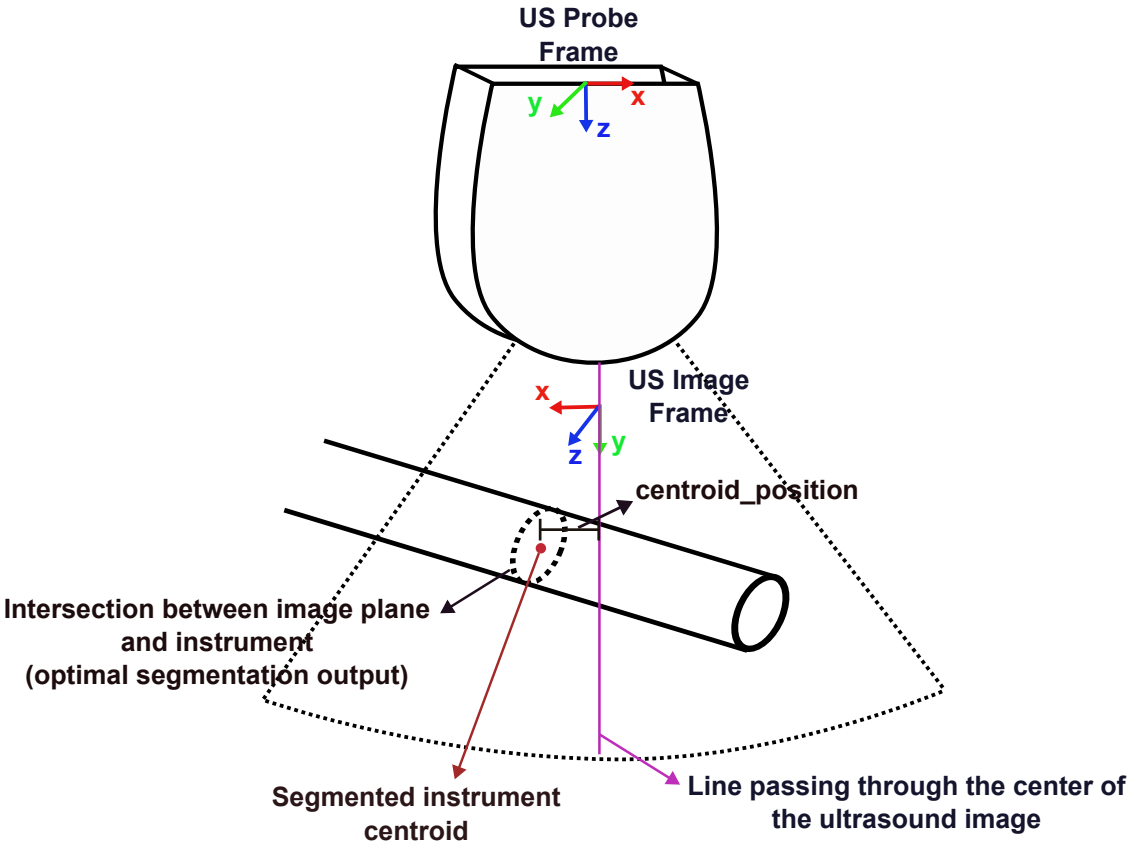


Figure 4.13: Representation of the segmented instrument centroid position relative to the ultrasound image frame and ultrasound probe frame.

Firstly, the US probe moves in the x-axis direction of the US image frame in order to have the segmented instrument centroid in the center of the image. Figure 4.13 shows the centroid position relative to the line passing through the image center, the distance between the segmentation centroid and this line corresponds to the distance which the probe must move. However, the distance produced by ROS and published to the '\instrument_centroid' topic is relative to the US image frame (see Fig. 4.4). Thus, it is necessary to transform this distance along the x-axis of the image frame into a distance along the x-axis of the probe frame. This is done by applying the rotation matrix from the image frame to the probe frame.

$$ {}^{probe}dist = {}^{probe}R_{image} \cdot {}^{image}dist \tag{4.1} $$

$$ {}^{probe}dist = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} centroid\_position \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -centroid\_position \\ 0 \\ 0 \end{bmatrix} \tag{4.2} $$

where ${}^{probe}R_{image}$ denotes the rotation matrix from the image to the probe frame, ${}^{image}dist$ the instrument centroid position in the image frame, ${}^{probe}dist$ the instrument centroid position in the probe frame, and *centroid_position* the value received from the topic '\instrument_centroid'.

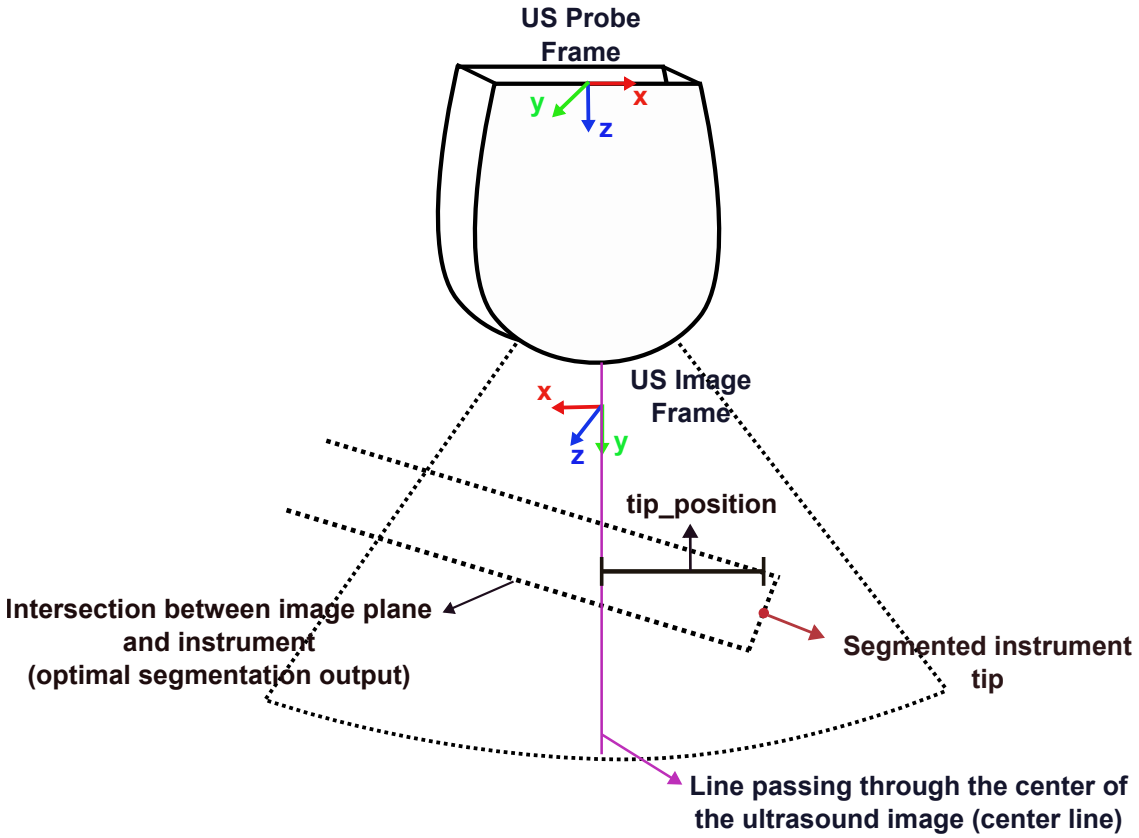


Figure 4.14: Representation of the segmented instrument tip position relative to the ultrasound image frame and ultrasound probe frame.

Hence, the robot task is to translate *-centroid_position* meters in the x-axis of the probe frame.

After the translation, a rotation of 20 degrees is done to one direction, followed by a rotation of -20 degrees to the opposite direction. This rotation is done around the z-axis of the probe frame. The rotation stops when the *e_found_aligned* event or the *e_lost* event is sent from ROS.

4.2.1.8 State *track_tip*

The *track_tip* state is similar to the first stage of the *align_instrument* state. The instrument tip position is received from ROS where the position is published to the topic '\instrument_tip' (see Fig. 4.4). The tip position corresponds to the distance between the tip and the line passing through the ultrasound image center (see Fig. 4.14).

The robot task in this state is to keep the instrument tip within a margin of 1 cm from the center line. This task is accomplished by moving the US probe in the x-axis direction of the probe frame. The tip position coming from ROS is relative to the image frame, so it needs to be transformed to the probe frame by using the same transformation from the *align_instrument* state (Equation 4.2). But, instead of having a *centroid_position*, there is a *tip_position*.

Figure 4.15: Ultrasound probe position, instrument position, and ultrasound image before starting the scanning routine.

4.3 Results and discussion

4.3.1 Scanning

The scanning routine was tested in order to check if the classification neural network was able to detect the ultrasound images containing the instrument. Five tests were done, having the instrument in different orientations. The tracking framework was able to detect when the instrument was present in the ultrasound image and stop the scanning routine in all five tests. Figure 4.15 shows an example of the experimental setup before starting the probe scanning, while Figure 4.16 shows the experimental setup when the scanning stops, meaning that the instrument was detected.

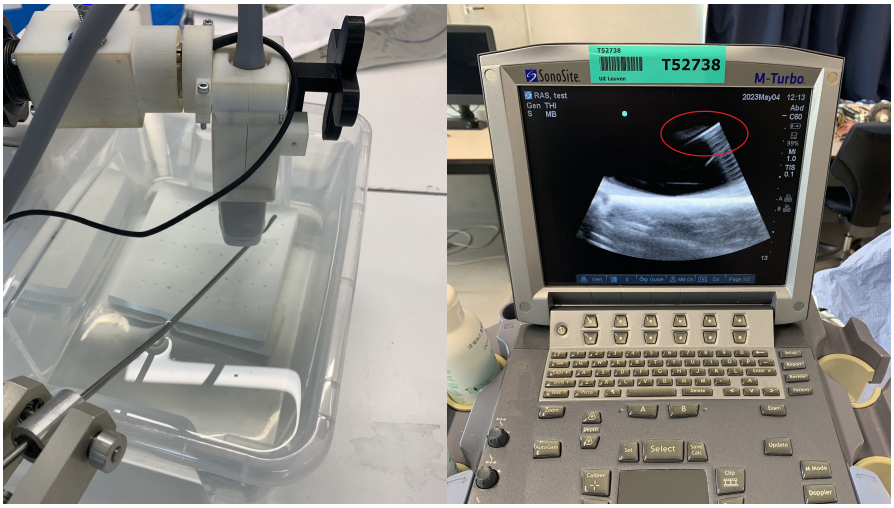
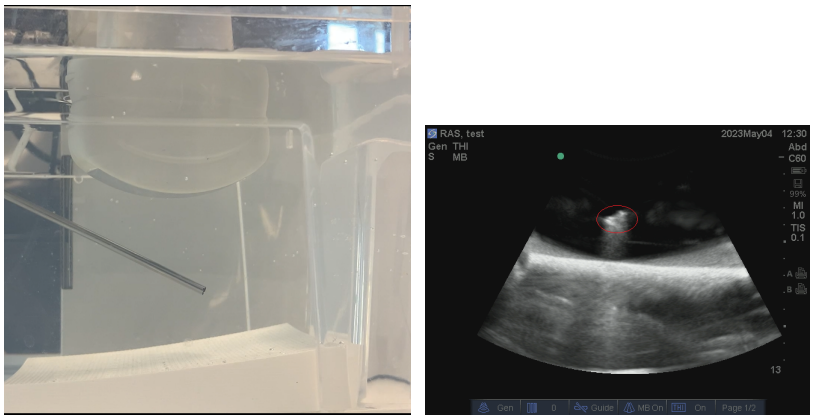


Figure 4.16: Ultrasound probe position, instrument position, and ultrasound image upon conclusion of the scanning routine. The instrument is present in the ultrasound image (red circle).



(a) Instrument and US probe positions (b) Ultrasound image with fetoscope cross section (red circle)

Figure 4.17: (a) Fetoscope position, probe position, and (b) ultrasound image before the instrument alignment.

This scanning step involves the *scan* state from the finite state machine (Section 4.2.1) and the 'image_classifier' node from the ROS middleware (Section 4.1.1.1). Since the instrument detection worked in all tests, it means that the ROS node, the classification neural network, and the *scan* state are working properly.

4.3.2 Instrument alignment

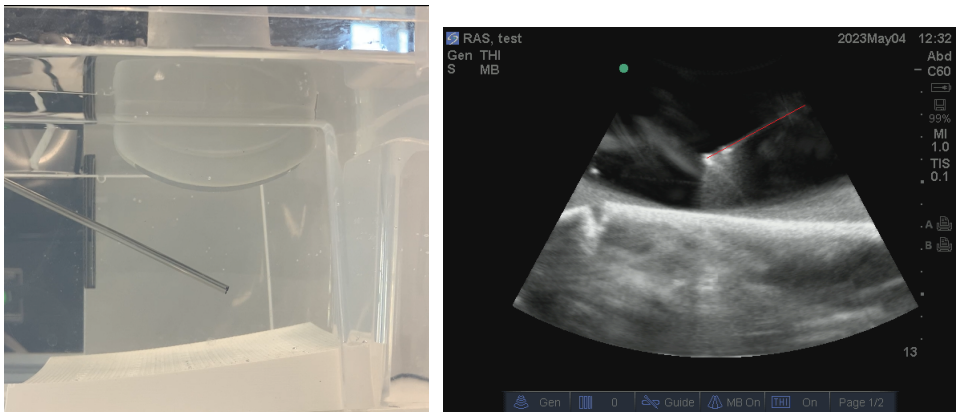
Next, the alignment capabilities of the tracking system was tested. The fetoscope was positioned in a way that the ultrasound image plane captured a cross section of the fetoscope. The objective is to assess if the tracking framework is able to correctly align the ultrasound probe with the fetoscope orientation. The states responsible for this task are the *check_instrument_alignment* and the *align_instrument* (Section 4.2.1).

Figure 4.17 shows the cross section of the fetoscope which is visible in the ultrasound image at the beginning of the instrument alignment test, while Figure 4.18 shows that the fetoscope axis is present in the US image, meaning that the instrument was aligned with the probe at the end of the test.

More tests should be performed to assess the robustness of this alignment procedure. Nevertheless, in the test performed the ultrasound-based instrument tracking system was able to correctly align the fetoscope in the ultrasound image by rotating the probe.

4.3.3 Instrument tracking

For assessing the system tracking performance it is necessary to have the position of the instrument position and of the ultrasound probe in the 3D physical space. By using these positions it is possible to check if the probe is tracking the instrument tip. The instrument and probe positions were obtained by using the *atracsys* optical tracking system.



(a) Instrument and US probe positions (b) Ultrasound image with fetoscope axis (red line)

Figure 4.18: (a) Fetoscope position, probe position, and (b) ultrasound image after the instrument alignment.

Two experiments were performed, the first one was used to assess the capability of keeping the instrument tip inside the 1 cm margin around the center line of the ultrasound image (see Fig. 4.14). In this first experiment, the instrument is kept aligned with the ultrasound image plane. Hence, the probe just need to perform in-plane motions.

The second experiment was used to assess the tracking performance of the whole ultrasound-based fetoscope tracking framework by moving around the fetoscope in different directions. Thus, the probe motion must include both in-plane and out-of-plane motions, while the system goes through different states.

Figure 4.19 shows the result obtained for the first test. The instrument tip position was kept inside the margin of 1 cm around the center line for 43.5 seconds, which represents 58% of the total experiment time of 75 seconds. Moreover, the highest distance from the fetoscope tip to the center line was 4.76 cm. The tracking performance is considerably good taking into consideration that the instrument tip was visible during the whole duration of the experiment.

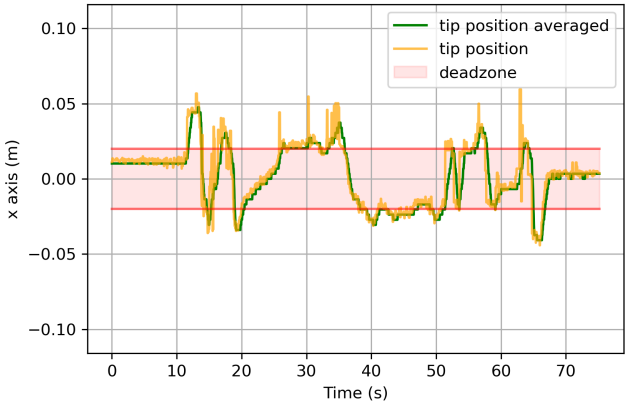


Figure 4.19: Tip position in ultrasound image frame x-axis during tracking of the instrument tip.

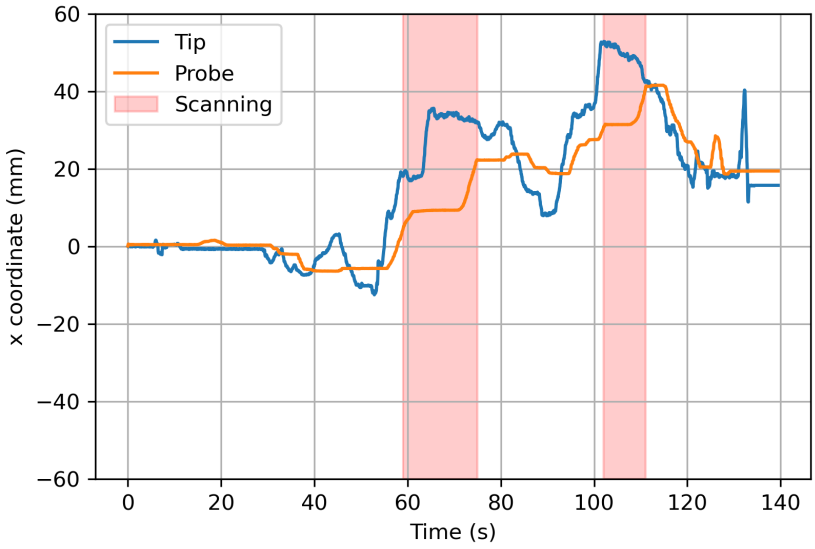


Figure 4.20: Tip position and probe position relative to the *initial frame* x-axis during tracking of the instrument tip.

Figure 4.19 also shows the effect of using the average of the tip position instead of the raw signal. There are a lot of peaks in the raw signal that are filtered out, allowing a smoother motion of the US probe.

Figures 4.20, 4.21, and 4.22 present the results of the second experiment. These figures present the positions of both the instrument tip and US probe relative to a fixed frame with its origin at the initial probe position, that is at the moment when the tracking started. This fixed frame is denominated *initial frame*. The z-axis of this frame is perpendicular to the phantom surface and the probe movements are performed in the x-y plane. Fig 4.22 shows that the probe position is maintained practically constant on the z-axis.

The red zones in the figures represent the moments where the finite state machine is in the scan state. During the scanning, it is possible to observe the delay of 5 seconds from the *start_scan_loop* substate, where the probe is not moving. The delay is followed by a linear translation of the probe which stops when the fetoscope is present in the ultrasound image.

The objective of the instrument tracking system is to have the probe position overlapping the instrument tip position in the x-y plane. Figures 4.20 and 4.21 show that the system can track the tip with some delay and a small error. The mean error between the tip position and the probe position in the x and y axes is presented in Table 4.4. The RMSE between the tip and the probe position was 8.61 mm.

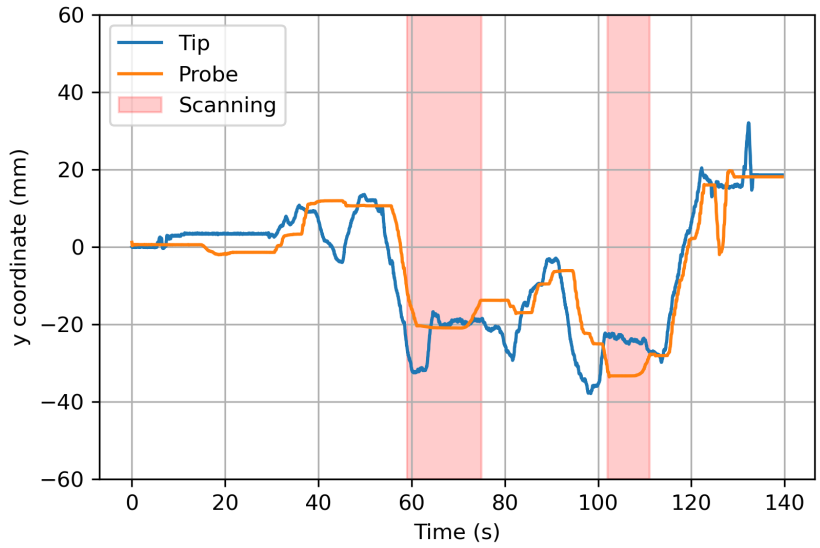


Figure 4.21: Tip position and probe position relative to the *initial frame* y-axis during tracking of the instrument tip.

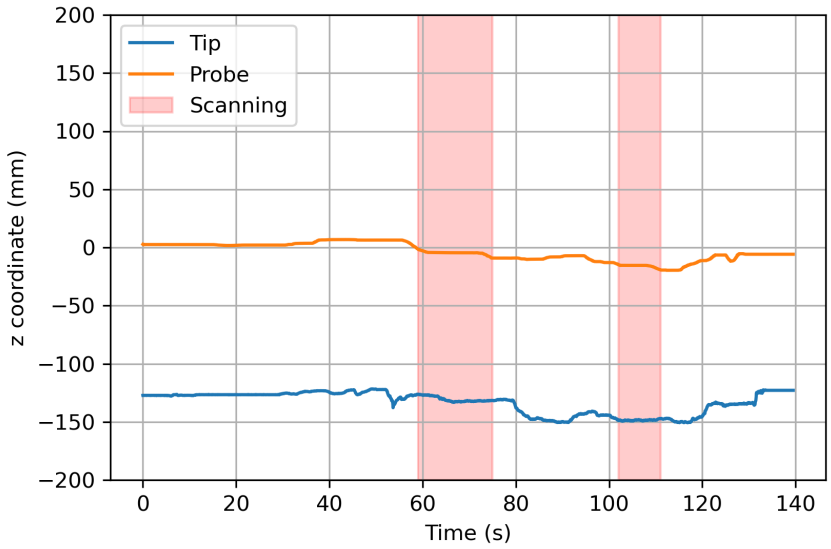


Figure 4.22: Tip position and probe position relative to the *initial frame* z-axis during tracking of the instrument tip.

Table 4.4: Mean error between tip position and probe position during the ultrasound-based instrument tracking

Mean Error Tip Tracking - x axis (mm)	**Mean Error Tip Tracking - y axis (mm)**
3.28 $\pm$ 9.21	-0.54 $\pm$ 7.23

The framework for ultrasound-based instrument tracking presented a good performance for tracking the fetoscope tip by means of controlling a 6-DoF robotic arm for the US probe positioning. The mean error average was below 4 mm. When the fetoscope was visible in the image, the tracking system was able to align the fetoscope axis with the US image plane. Furthermore, when the instrument was out of the image plane, the scanning routine would start and accomplish its objective by finding the instrument. The US-based fetoscope tracking system still needs to go through more experiments in order to assess its robustness and capability of tracking the instrument in a real clinical scenario.

Chapter 5

Concluding Remarks and Future Work

5.1 Conclusion

This thesis presents the description of the research done towards the development of a real-time robotic ultrasound-based instrument tracking framework during FETO. The methods and results were extensively discussed in the different chapters. The clinical background was provided in the first chapter, followed by the state-of-the-art on instrument localization algorithms in US images and on autonomous robotic US imaging tracking. Next, a comparison between different developed fetoscope localization algorithms in 2D ultrasound images was performed. Finally, Chapter 4 described the ultrasound-based instrument tracking framework.

The results of this master's thesis are related to two objectives. The first objective is the development of a fetoscope localization algorithm in ultrasound images and the second is the development of a framework for ultrasound-based fetoscope tracking.

For the first objective, five different algorithms were developed based on the state-of-the-art. The first method was based on a Gabor filter, while the other four were based on deep learning models for image segmentation. The deep learning algorithms were compared in terms of instrument segmentation performance and computation time. The W-Net was the deep learning model with best results for segmentation with an IoU score of 57.2% on the test dataset. W-Net was also the slowest model, with an average of computation time required to perform the segmentation of 76.25 ms. The fastest model was the EU-Net, with an average computation time of 24.87 ms. Next, the instrument tracking performance of all the developed localization algorithms was assessed. The objective of the instrument tracking was to localize the ground-truth position of the instrument tip in three different ultrasound videos. The Gabor method was the algorithm with the best performance, presenting a maximum average tip error of 3.52 mm, while also being the fastest method. However, the Gabor method is semi-automatic, which hampers its real-time capabilities. Thus, the OEU-Net was considered the preferred methodology for performing the instrument tip localization. The OEU-Net was the second best deep learning model for the segmentation task and was the best one regarding the instrument tracking performance among the deep learning methods. Moreover, the OEU-Net has similar computation times to the Gabor method, not being slow as the

W-Net model. It was also concluded that the deep learning methodologies presented a systematic error due to the instrument segmentation output that is produced by the neural networks.

Since there were no available ultrasound videos from FETO procedures and the labeling of the ultrasound images pixels for training the neural networks for segmentation is a very time consuming task. The optical tracking system was used to automatically produce the ground-truths for segmentation and tip tracking. The optical tracking system is not free from errors and the computation of the transformations between different frames may lead to some numerical approximations. Thus, the ground-truth may not be correctly overlaid with the real instrument localization in the ultrasound image. This may have caused a poorer segmentation and tracking performance of the localization algorithms. Moreover, the images acquired in the phantom does not present the complexity that is present in the ultrasound images from the FETO intervention, where different anatomical structures are present. Hence, the localization algorithms may have had an overestimated performance.

The second objective aims to develop a framework for ultrasound-based fetoscope tracking by using an autonomous robotic US imaging tracking system. This framework was developed using a finite state machine approach which was implemented with ROS, eTaSL, Orocos, and rFSM softwares. The tracking system starts by scanning the phantom until it detects the instrument in the ultrasound image by means of a classification neural network. This detection procedure worked for all the experiments done. One of the states of the state machine deals with the alignment between the US probe and the instrument orientation. This alignment worked correctly, by rotating the probe until the instrument axis was present in the ultrasound image. Another state deals with the instrument tip tracking while the instrument is aligned with the image plane. Thus, only in-plane motions are performed with the probe. The objective of the tracking is to keep the instrument tip close to the center of the ultrasound image. The system was able to maintain the instrument tip visible within a 1 cm margin around the center of the ultrasound image. When performing an experiment with more dynamic movements, where the instrument was moving outside and within the image plane, the tracking system had a good performance. The mean error between the probe position that was being manipulated by a 6-DoF robot and the instrument tip position was lower than 4 mm.

More experiments should be performed with different instrument orientations and movements to assess the robustness of the ultrasound-based fetoscope tracking framework. Furthermore, the simple and static environment of the phantom may not be representative of the real scenario during a FETO intervention. Thus, the tracking system performance may be overestimated. Moreover, the instrument was kept still during some experiments, which may not be the case during FETO.

The equipment for the experimental setup was not always available, which had an impact on the number of experiments performed. Moreover, certain aspects of the research may have been constrained by the limited time frame, potentially impacting the overall depth of the findings. Nevertheless, an effort was done in order to maximize the value and rigor of this work within the given time frame.

In conclusion, it can be stated that the results of this master's thesis are a promising building

block for the automation of the FETO procedure. The results showed that it is possible to localize the instrument in 2D ultrasound images and use this information to manipulate the US probe position by using a 6-DoF robot. The implementation of this automation for fetoscope tracking in hospitals should reduce the physical and cognitive burden on the sonographer.

5.2 Future work

The ultrasound-based fetoscope tracking framework developed in this thesis must go through some additional experiments in order to optimize its parameters, have a better assessment of the tracking performance, and evaluate its robustness. In case the information gathered from the developed fetoscope localization algorithm is not sufficient for having a good tracking performance, a way to obtain the instrument pose should be developed. The instrument pose could be retrieved from a multi task neural network such as the OEU-Net or from additional information provided by another tracking system such as an electromagnetic tracking system. Moreover, a hybrid force and position control of the US probe should be implemented. The force control would be used to maintain the contact of the US probe with the mother's belly during FETO. Then, the tracking system should be tested with a more realistic phantom. After the tracking performance evaluation and safety assessment in the realistic phantom, the system could be implemented and tested in a clinical scenario.

Appendix A

Gabor Filter Algorithm

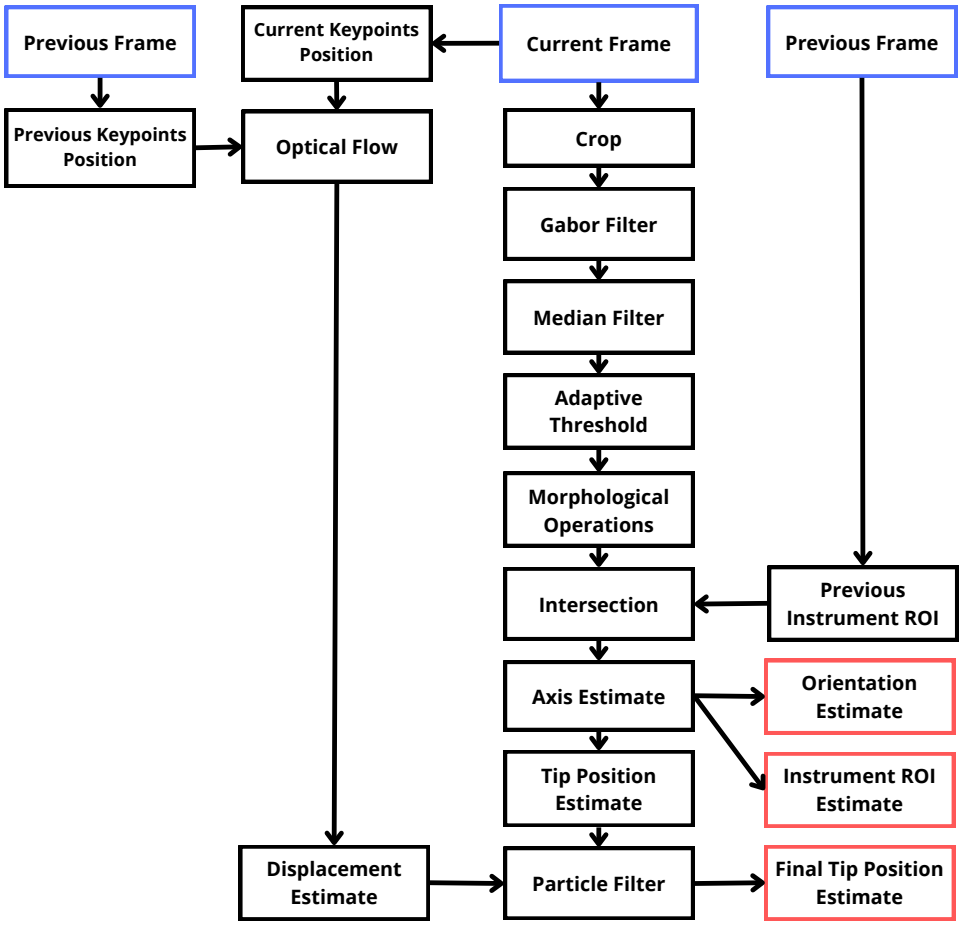


Figure A.1: Gabor filter algorithm having inputs in blue rectangles and outputs in red rectangles

References

Aertbeliën, E. (2020). etasl documentation. https://etasl.pages.gitlab.kuleuven.be/contents.html.

Aertbeliën, E. and De Schutter, J. (2014). Etasl/etc: A constraint-based task specification language and robot controller using expression graphs. pages 1540 – 1546. IEEE. 10.1109/IROS.2014.6942760.

Ahmad, M. A., Ourak, M., Gruijthuijsen, C., Deprest, J., Vercauteren, T., and Poorten, E. V. (2020). Deep learning-based monocular placental pose estimation: towards collaborative robotics in fetoscopy. *International Journal of Computer Assisted Radiology and Surgery*, 15:1561–1571. 10.1007/s11548-020-02166-3.

Alsbeih, D., Daoud, M. I., Al-Tamimi, A. K., and Al-Jarrah, M. A. (2020). A dynamic system for tracking biopsy needle in two dimensional ultrasound images. volume 2020-October. IEEE Computer Society. 10.1109/MECBME47393.2020.9265166.

Antico, M., Sasazawa, F., Wu, L., Jaiprakash, A., Roberts, J., Crawford, R., Pandey, A. K., and Fontanarosa, D. (2019). Ultrasound guidance in minimally invasive robotic procedures. *Medical Image Analysis*, 54:149–167. 10.1016/j.media.2019.01.002.

Atracsys (2017). *fTk250-datasheet*. Puidox, Switzerland.

Beigi, P., Rohling, R., Salcudean, S. E., and Ng, G. C. (2017). Casper: computer-aided segmentation of imperceptible motion—a learning-based tracking of an invisible needle in ultrasound. *International Journal of Computer Assisted Radiology and Surgery*, 12:1857–1866. 10.1007/s11548-017-1631-4.

Cao, K., Mills, D., and Patwardhan, K. A. (2013). Automated catheter detection in volumetric ultrasound. pages 37–40. IEEE. 10.1109/ISBI.2013.6556406.

Chalabi, N. E., Attia, A., and Akrouf, S. (2021). Machine learning and deep learning: Recent overview in medical care. pages 223–231. 10.1007/978-3-030-49932-7_22.

Chandrasekharan, R., Rawat, M., Madappa, R., Rothstein, D. H., and Lakshminrusimha, S. (2017). Congenital diaphragmatic hernia: A review. *Maternal Health, Neonatology, and Perinatology*, pages 3–6. 10.1186/s40748-017-0045-1.

Chatelain, P., Krupa, A., and Marchal, M. (2013). Real-time needle detection and tracking using a visually servoed 3d ultrasound probe. pages 1676–1681. Institute of Electrical and Electronics Engineers Inc. 10.1109/ICRA.2013.6630795.

Chen, Q., Liu, F. W., Xiao, Z., Sharma, N., Cho, S. K., and Kim, K. (2019). Ultrasound tracking of the acoustically actuated microswimmer. *IEEE Transactions on Biomedical Engineering*, 66:3231–3237. 10.1109/TBME.2019.2902523.

Chen, S., Lin, Y., Li, Z., Wang, F., and Cao, Q. (2022). Automatic and accurate needle detection in 2d ultrasound during robot-assisted needle insertion process. *International Journal of Computer Assisted Radiology and Surgery*, 17:295–303. 10.1007/s11548-021-02519-6.

Conrad, S. A. (2010). *"Focused Echocardiography in the ICU", in Bedside Procedures for the Intensivist*. Springer New York, NY, USA, 1 edition.

Daoud, M., Khraiwesh, S., Zayadeen, A., and Alazrai, R. (2017). Accurate needle localization in two-dimensional ultrasound images. pages 578–582. https://ieeexplore.ieee.org/document/8266353.

Deprest, J., Barki, G., Ville, Y., Hecher, K., Gratacos, E., and Nicolaides, K. (2015). *ENDOSCOPY IN FETAL MEDICINE*. Endo Press, 3 edition.

Deprest, J., Gratacos, E., and Nicolaides, K. H. (2004). Fetoscopic tracheal occlusion (feto) for severe congenital diaphragmatic hernia: Evolution of a technique and preliminary results. *Ultrasound in Obstetrics and Gynecology*, 24:121–126. 10.1002/uog.1711.

Deprest, J., Nicolaides, K., Done', E., Lewi, P., Barki, G., Largen, E., Dekoninck, P., Sandaite, I., Ville, Y., Benachi, A., Jani, J., Amat-Roldan, I., and Gratacos, E. (2011). Technical aspects of fetal endoscopic tracheal occlusion for congenital diaphragmatic hernia. *Journal of Pediatric Surgery*, 46:22–32. 10.1016/j.jpedsurg.2010.10.008.

der Veeken, L. V., Russo, F. M., Catte, L. D., Gratacos, E., Benachi, A., Ville, Y., Nicolaides, K., Berg, C., Gardener, G., Persico, N., Bagolan, P., Ryan, G., Belfort, M. A., and Deprest, J. (2018). Fetoscopic endoluminal tracheal occlusion and reestablishment of fetal airways for congenital diaphragmatic hernia. *Gynecological Surgery*, 15. 10.1186/s10397-018-1041-9.

Ding, M. and Fenster, A. (2003). A real-time biopsy needle segmentation technique using hough transform. *Medical Physics*, 30:2222–2233. 10.1118/1.1591192.

Du, X., Kurmann, T., Chang, P. L., Allan, M., Ourselin, S., Sznitman, R., Kelly, J. D., and Stoyanov, D. (2018). Articulated multi-instrument 2-d pose estimation using fully convolutional networks. *IEEE Transactions on Medical Imaging*, 37:1276–1287. 10.1109/TMI.2017.2787672.

Duflot, L. A., Krupa, A., Tamadazte, B., and Andreff, N. (2016). Towards ultrasound-based visual servoing using shearlet coefficients. volume 2016-June, pages 3420–3425. Institute of Electrical and Electronics Engineers Inc. 10.1109/ICRA.2016.7487519.

FUJIFILM Sonosite (2023). Sonosite m-turbo | sonosite fujifilm. https://www.sonosite.com/products/sonosite-m-turbo.

Gruijthuijsen, C., Colchester, R., Devreker, A., Javaux, A., Maneas, E., Noimark, S., Xia, W., Stoyanov, D., Reynaerts, D., Deprest, J., Ourselin, S., Desjardins, A., Vercauteren, T., and Poorten, E. V. (2018). Haptic guidance based on all-optical ultrasound distance sensing for safer minimally invasive fetal surgery. *Journal of Medical Robotics Research*, 3. 10.1142/S2424905X18410015.

Gupta, V. S. and Harting, M. T. (2020). Congenital diaphragmatic hernia-associated pulmonary hypertension. *Seminars in Perinatology*, 44. 10.1053/j.semperi.2019.07.006.

HAPTION SA (2023). Virtuose 6d rv. https://www.haption.com/en/products-en/virtuose-6d-en.html.

Hasan, M. K., Calvet, L., Rabbani, N., and Bartoli, A. (2021). Detection, segmentation, and 3d pose estimation of surgical tools using convolutional neural networks and algebraic geometry. *Medical Image Analysis*, 70. 10.1016/j.media.2021.101994.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. 10.48550/arXiv.1512.03385.

Jani, J. C., Nicolaides, K. H., Gratacós, E., Valencia, C. M., Doné, E., Martinez, J. M., Gucciardo, L., Cruz, R., and Deprest, J. A. (2009). Severe diaphragmatic hernia treated by fetal endoscopic tracheal occlusion. *Ultrasound in Obstetrics and Gynecology*, 34:304–310. 10.1002/uog.6450.

Kaehler, A. and Bradski, G. (2016). *Learning OpenCV 3*. O'Reilly Media, Inc., 1 edition.

Kaya, M. and Bebek, O. (2014). Needle localization using gabor filtering in 2d ultrasound images. pages 4881–4886. Institute of Electrical and Electronics Engineers Inc. 10.1109/ICRA.2014.6907574.

Kaya, M., Senel, E., Ahmad, A., Orhan, O., and Bebek, O. (2015). Real-time needle tip localization in 2d ultrasound images for robotic biopsies. pages 47–52. Institute of Electrical and Electronics Engineers Inc. 10.1109/ICAR.2015.7251432.

Klotzbuecher, M. (2013). rfsm statechards v1.0. https://orocos.org/stable/documentation/rFSM/index.html.

Kora, P., Ooi, C. P., Faust, O., Raghavendra, U., Gudigar, A., Chan, W. Y., Meenakshi, K., Swaraja, K., Plawiak, P., and Acharya, U. R. (2022). Transfer learning techniques for medical image analysis: A review. *Biocybernetics and Biomedical Engineering*, 42:79–107. 10.1016/j.bbe.2021.11.004.

Li, K., Xu, Y., and Meng, M. Q. (2021a). An overview of systems and techniques for autonomous robotic ultrasound acquisitions. *IEEE Transactions on Medical Robotics and Bionics*, 3:510–524. 10.1109/TMRB.2021.3072190.

Li, R., Cai, Y., Niu, K., and Poorten, E. V. (2021b). Comparative quantitative analysis of robotic ultrasound image calibration methods. 10.1109/ICAR53236.2021.9659341.

Maier, A., Steidl, S., Christlein, V., and Hornegger, J., editors (2018). *Medical Imaging Systems*, volume 11111. Springer International Publishing. 10.1007/978-3-319-96520-8.

Mebarki, R., Krupa, A., and Chaumette, F. (2010). 2-d ultrasound probe complete guidance by visual servoing using image moments. *IEEE Transactions on Robotics*, 26:296–306. 10.1109/TRO.2010.2042533.

Monfaredi, R., Wilson, E., Koutenaei, B. A., Labrecque, B., Leroy, K., Goldie, J., Louis, E., Swerdlow, D., and Cleary, K. (2015). Robot-assisted ultrasound imaging: Overview and development of a parallel telerobotic system. *Minimally Invasive Therapy and Allied Technologies*, 24:54–62. 10.3109/13645706.2014.992908.

Mwikirize, C., Kimbowa, A. B., Imanirakiza, S., Katumba, A., Nosher, J. L., and Hacihaliloglu, I. (2021). Time-aware deep neural networks for needle tip localization in 2d ultrasound. *International Journal of Computer Assisted Radiology and Surgery*, 16:819–827. 10.1007/s11548-021-02361-w.

Mwikirize, C., Nosher, J. L., and Hacihaliloglu, I. (2017). Local phase-based learning for needle detection and localization in 3d ultrasound. pages 108–115. 10.1007/978-3-319-67543-5_10.

Nadeau, C. and Krupa, A. (2011). Improving ultrasound intensity-based visual servoing: Tracking and positioning tasks with 2d and bi-plane probes. pages 2837–2842. Institute of Electrical and Electronics Engineers (IEEE). 10.1109/iros.2011.6094886.

OpenAI (2023). Introducing chatgpt. https://openai.com/blog/chatgpt. (accessed Jun. 07, 2023).

Open Robotics (2022). Ros.org. http://wiki.ros.org/.

Open Source Computer Vision (2023). Tutorial optical flow. https://docs.opencv.org/3.4/d4/dee/tutorial_optical_flow.html.

Patel, N. and Kipfmueller, F. (2017). Cardiac dysfunction in congenital diaphragmatic hernia: Pathophysiology, clinical assessment, and management. *Seminars in Pediatric Surgery*, 26:154–158. 10.1053/j.sempedsurg.2017.04.001.

Popov, V. V., Kudryavtseva, E. V., Katiyar, N. K., Shishkin, A., Stepanov, S. I., and Goel, S. (2022). Industry 4.0 and digitalisation in healthcare. *Materials*, 15. 10.3390/ma15062140.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. pages 234–241. 10.1007/978-3-319-24574-4_28.

Sadler, T. W. (2011). *Langman's medical embriology*. Williams & Wilkins, 12 edition.

Shi, J. and Tomasi (1994). Good features to track. pages 593–600. IEEE Comput. Soc. Press. 10.1109/CVPR.1994.323794.

Silva, F. J. G., Santos, G., Ferreira, P., Lopes, M. P., Mcdermott, O., Foley, I., Antony, J., Sony, M., and Butler, M. (2022). The impact of industry 4.0 on the medical device regulatory product life cycle compliance. 10.3390/su.

Soetens, P., Issaris, T., Bruyninckx, H., Joyeux, S., Smits, R., et al. (2020). Orocos project documentation. https://docs.orocos.org/.

Sorriento, A., Porfido, M. B., Mazzoleni, S., Calvosa, G., Tenucci, M., Ciuti, G., and Dario, P. (2020). Optical and electromagnetic tracking systems for biomedical applications: A critical review on potentialities and limitations. *IEEE Reviews in Biomedical Engineering*, 13:212–232. 10.1109/RBME.2019.2939091.

Texas Children's Fetal Center (2023). Congenital diaphragmatic hernia (cdh). https://women.texaschildrens.org/program/texas-childrens-fetal-center/conditions-we-treat/congenital-diaphragmatic-hernia-cdh.

Tong, X. L., Wang, L., Gao, T. B., Qin, Y. G., Qi, Y. Q., and Xu, Y. P. (2009). Potential function of amniotic fluid in fetal development-novel insights by comparing the composition of human amniotic fluid with umbilical cord and maternal serum at mid and late gestation. *Journal of the Chinese Medical Association*, 72:368–373. 10.1016/S1726-4901(09)70389-2.

Tsao, K. J. and Lally, K. P. (2008). The congenital diaphragmatic hernia study group: a voluntary international registry. *Seminars in Pediatric Surgery*, 17:90–97. 10.1053/j.sempedsurg.2008.02.004.

Wang, J., Zhu, H., Wang, S.-H., and Zhang, Y.-D. (2021). A review of deep learning on medical image analysis. *Mobile Networks and Applications*, 26:351–380. 10.1007/s11036-020-01672-7/Published.

Yang, H., Shan, C., Kolen, A. F., and de With, P. H. N. (2022). Medical instrument detection in ultrasound: a review. *Artificial Intelligence Review*. 10.1007/s10462-022-10287-1.

Yang, L., Wang, J., Kobayashi, E., Liao, H., Yamashita, H., Sakuma, I., and Chiba, T. (2013). Ultrasound image-based endoscope localization for minimally invasive fetoscopic surgery. pages 1410–1413. 10.1109/EMBC.2013.6609774.

Zhao, Y., Liebgott, H., and Cachard, C. (2015). Comparison of the existing tool localisation methods on two-dimensional ultrasound images and their tracking results. *IET Control Theory and Applications*, 9:1124–1134. 10.1049/iet-cta.2014.0672.

Zhao, Y., Lu, Y., Lu, X., Jin, J., Tao, L., and Chen, X. (2022). Biopsy needle segmentation using deep networks on inhomogeneous ultrasound images. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, 2022:553–556. 10.1109/EMBC48229.2022.9871059.

Zhao, Y., Shen, Y., Bernard, A., Cachard, C., and Liebgott, H. (2017). Evaluation and comparison of current biopsy needle localization and tracking methods using 3d ultrasound. *Ultrasonics*, 73:206–220. 10.1016/j.ultras.2016.09.006.

Zhao, Y., Wang, Y., Yu, Y., Yang, F., and Shen, Y. (2020). Automatic recognition and tracking of liver blood vessels in ultrasound image using deep neural networks. volume 2020-December, pages 499–504. Institute of Electrical and Electronics Engineers Inc. 10.1109/ICSP48669.2020.9320944.