

POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Visual Servoing Control for Robot-Assisted Vascular Ultrasound Imaging and Reconstruction

Tesi di Laurea Magistrale in
Automation and Control Engineering - Ingegneria dell'Automazione

Author: **Leonardo Tuberti**

Student ID: 10576323
Advisor: Prof. Elena De Momi
Co-advisors: Junling Fu
Academic Year: 2024-25

Abstract

Ultrasound imaging is a key tool in vascular diagnostics due to its real-time visualization, portability, low cost, and safety. However, conventional procedures rely heavily on operator skill, making image acquisition inconsistent and hard to replicate. This limits access to high-quality diagnostics, especially in remote areas. Robotic-assisted ultrasound systems (RUSS) offer a solution by automating and standardizing the scanning process.

This thesis presents a semi-autonomous robotic system for vascular ultrasound imaging based on image-based visual servoing (IBVS). The platform uses a 7-DOF KUKA LWR IV+ robotic arm and a Telemed MicrUs EXT-1H linear probe. It dynamically adjusts probe alignment using real-time feedback from ultrasound images to track a target blood vessel.

A central aspect is the comparison of four tracking strategies for detecting the vessel's center ("proxy point"): (i) a U-Net-based segmentation extracting the vessel contour and center; (ii) a YOLO-based detector identifying bounding boxes and centers; (iii) a sparse optical flow (Lucas-Kanade) applied to YOLO outputs for frame-to-frame tracking; and (iv) a dense optical flow (Farnebäck) initialized from YOLO, offering pixel-level motion estimation.

The system is built using the Robot Operating System (ROS) to ensure modularity and real-time control. Visual servoing adjusts the robot's motion based on image error between the current and desired vessel position. Experiments on a vascular phantom assess tracking accuracy, stability, robustness, and computational efficiency. Results show U-Net offers high accuracy but is slower; YOLO is fast but less stable.

Keywords: robotic ultrasound, visual servoing, vessel tracking, U-Net, YOLO, optical flow, KUKA LWR IV+, vascular imaging.

Abstract in lingua italiana

L'ecografia è uno strumento fondamentale per la diagnostica vascolare, grazie alla visualizzazione in tempo reale, alla portabilità, al basso costo e all'assenza di radiazioni. Tuttavia, i metodi tradizionali dipendono fortemente dall'abilità dell'operatore, causando variabilità nelle immagini e bassa ripetibilità. Questo limita l'accesso a diagnosi di qualità, soprattutto in aree remote. I sistemi ecografici robotici (RUSS) offrono una soluzione automatizzando il processo di scansione.

Questa tesi presenta un sistema robotico semi-autonomo per imaging vascolare basato su visual servoing guidato da immagini (IBVS). Il sistema utilizza un braccio robotico KUKA LWR IV+ a 7 gradi di libertà e una sonda lineare Telemed MicrUs EXT-1H, regolando dinamicamente la posizione della sonda tramite feedback visivo in tempo reale per seguire un vaso sanguigno.

Sono confrontate quattro strategie di tracking del centro del vaso ("proxy point"): (i) segmentazione U-Net per estrarre contorno e centro; (ii) rilevamento YOLO con box e centro; (iii) optical flow sparso (Lucas-Kanade) su output YOLO; (iv) optical flow denso (Farnebäck) inizializzato da YOLO per il tracking pixel-wise.

Il sistema si basa su Robot Operating System (ROS), garantendo modularità e controllo in tempo reale. Il visual servoing guida il movimento robotico in base all'errore tra posizione corrente e desiderata del vaso. I test su un fantoccio vascolare valutano accuratezza, stabilità, robustezza ed efficienza. I risultati mostrano che U-Net è preciso ma lento, YOLO veloce ma meno stabile.

Parole chiave:ecografia robotica, visual servoing, tracciamento vascolare, U-Net, YOLO, optical flow, KUKA LWR IV+, imaging vascolare.

Contents

Abstract **i**

Abstract in lingua italiana **iii**

Contents **v**

1 Introduction **1**
1.1 Clinical Background . . . 1
1.2 State-Of-The-Art . . . 2
1.2.1 Ultrasound Imaging . . . 2
1.2.2 Autonomy . . . 3
1.2.3 Visual Servoing . . . 4
1.3 Motivation and Contribution . . . 6

2 Materials and Methods **9**
2.1 Robotic System . . . 10
2.1.1 Technical Specifications . . . 10
2.1.2 Software Environment . . . 13
2.2 Ultrasound Imaging System . . . 15
2.2.1 Device Overview . . . 15
2.2.2 Technical Specifications . . . 15
2.2.3 HDMI Image Acquisition via Magewell USB Capture Gen 2 . . . 16
2.3 Ultrasound image segmentation and ROI tracking . . . 17
2.3.1 U-NET . . . 18
2.3.2 Optical Flow . . . 20
2.3.3 YOLO/CNN . . . 26
2.4 System calibration and Coordinate Transformation . . . 29
2.5 Robot Visual Servoing Control . . . 30
2.6 Position controller . . . 31

2.6.1 Close-loop Control 32

3 Experiments **35**

3.1 Experimental Setup 35

3.2 Performance metrics 38

3.3 Experiment Results 40

3.3.1 Experiment 1: Visual Servoing Tracking Under Linear External Disturbance 40

3.3.2 Experiment 2: Visual Servoing Tracking Under Sinusoidal External Disturbance 42

3.3.3 Experiment 3: Visual Servoing Tracking Under Manual Movement of the Phantom 45

3.4 Reconstruction results 47

4 Conclusions and future developments **49**

Bibliography **51**

List of Figures **55**

List of Tables **57**

Acknowledgements **59**

1 | Introduction

1.1. Clinical Background

Medical imaging plays a fundamental role in modern healthcare, providing non-invasive methods to visualize internal anatomical structure and aiding in disease diagnosis, treatment planning, and surgical navigation. Among the most widely used imaging modalities are Computed Tomography (CT)[1] and Magnetic Resonance Imaging (MRI)[2].

CT scans utilize X-rays to generate cross-sectional images of the body, offering high-resolution visualization of bones, soft tissues, and vasculature. However, they expose patients to ionizing radiation. MRI, on the other hand, uses strong magnetic fields and radio waves to produce detailed images, particularly excelling in soft tissue differentiation. Despite their high imaging quality, both CT and MRI require large, stationary equipment and often lengthy scanning procedures, limiting their real-time and bedside applicability.

In contrast Ultrasound imaging (US)[3] provides a portable, real-time, and non-invasive solution, making it particularly valuable for vascular imaging. Unlike CT and MRI, ultrasound does not rely on radiation or strong magnetic fields but instead uses high-frequency sound waves to generate images of internal structures. This makes it ideal for applications such as vascular assessment, blood flow monitoring, and guiding interventional procedures. However, traditional manual ultrasound scanning is highly operator-dependent, requiring extensive experience to obtain high-quality images and maintain probe stability.

Robot assisted ultrasound (US) imaging, is an innovative approach in medical diagnostics, combining the efficiency of advanced imaging techniques with robotics to improve consistency, accuracy and accessibility of ultrasound procedures. Traditional ultrasound diagnosis requires the physical presence of a trained sonographer to manipulate the probe, on the other hand, robot-assisted systems have the potential for remote operations. In this way also with the absence of a trained clinician it is possible to expand medical care also in underserved or geographically remote areas. Ultrasound-imaging is a widely used non-invasive diagnostic tool known for its real-time capabilities, low cost and safety due to the absence of ionizing radiation. However, its effectiveness often relies heavily on the

operator skills and experiences to position the probe and to maintain the contact with the patient. This operator dependence can lead to high variability in the results and limits the availability of high-quality ultrasound services [4]. The use of robotics in ultrasound imaging addresses these problems by increasing precision, stability, remote operations, and also to decrease the workload of the human operator. Furthermore, there are different studies that have shown that these non-ergonomic procedures create a high percentage of work-related muscle disorders (WRMSDs) in the sonographer[5]. Despite its potential remote robot-assisted ultrasound imaging faces significant challenges[6] which are listed below:

- **Probe positioning:** Ensuring the robot capability to adjust probe's position and orientation based on real-time ultrasound images is complex due to the dynamic and deformable nature of human tissue.
- **Human-Robot interaction:** Autonomy is valuable but human expertise remains crucial in interpreting ultrasound images and making fine adjustments.Therefore, a shared control paradigm between human and robot is needed.
- **Latency and Safety:** Remote operation introduces delay in communication that can compromise the safety of the procedure.

This thesis focuses on the development of a visual servoing control framework for robot-assisted vascular ultrasound imaging and reconstruction. The proposed system leverages real-time image processing techniques, including U-Net, Optical Flow, and YOLO, to detect and track veins, enabling autonomous robot motion to maintain optimal probe alignment.

1.2. State-Of-The-Art

1.2.1. Ultrasound Imaging

Ultrasound imaging has long been very important in medical diagnostics, valued for its non-invasive nature, absence of ionizing radiation, and real-time imaging capabilities. Traditionally, ultrasound examinations are performed manually by skilled sonographers, a process that can be highly operator-dependent and subject to variability. To mitigate these challenges, the integration of robotic systems into ultrasound imaging has emerged as a significant advancement. Robotic Ultrasound Systems (RUSS) aim to enhance reproducibility, improve dexterity, and provide intelligent, anatomy-aware imaging.These systems hold the potential to deliver consistent diagnostic outcomes and address the

shortage of experienced sonographers.

In the realm of vascular imaging[7], ultrasound serves as a critical tool for assessing blood flow and detecting vascular abnormalities. However, manual ultrasound examinations can be limited by operator dependency and variability in image acquisition. The integration between robotic systems and ultrasound imaging (US) not only improves diagnostic accuracy but also facilitates minimally invasive procedures by providing real-time, image-guided navigation.

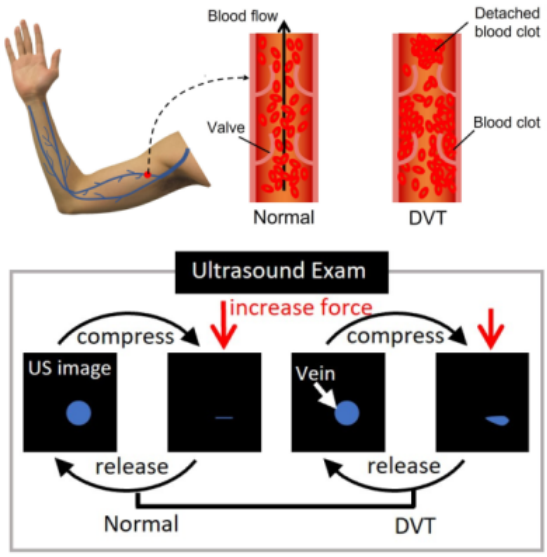


Figure 1.1: Deep Venous Thrombosis Ultrasound Examination[8]

1.2.2. Autonomy

The field of robotic ultrasound imaging categorizes systems based on their Level of Robot Autonomy (LORA) [9]. This classification goes from teleoperated systems to fully autonomous ones. Based on LORA we have the classification of the systems as follow:

- **Teleoperated Systems LORA(1-2):** in this type of system the sonographer directly controls the robot's motion, usually using joystick or an haptic device. These systems are designed for remote diagnostics, minimizing the need for on-site expertise. An example of this type of system is the MELODY (AdEchoTech) which is a remote-controlled robotic probe holder with limited autonomy. It requires an on-site assistant for coarse probe positioning, while fine adjustments are handled remotely.



Figure 1.2: Melody AdEchoTech

- **Collaborative Assistance Systems LORA(3-4):** these systems are partially automated, they provide decision support or execute pre-defined actions. The robot collaborates with the operator to enhance imaging accuracy or efficiency. These systems are mainly used in image optimization, adjusting the probe's angle or pressure to improve image quality based on real-time feedback or in trajectory planning, automating repetitive scans over specific anatomical regions, such as vascular imaging or spine reconstruction.
- **Autonomous Systems LORA(5-7):** High-level autonomy, robots independently plan and execute tasks such as probe positioning, trajectory planning, and image acquisition. With adaptive controls the robot adjusts its action dynamically based on ultrasound feedback or external sensors. Exploited above all in 3D image reconstruction in which the systems autonomously acquire volumetric images, using algorithms to adjust probe positioning for optimal coverage or in therapy guidance.
- **Full Autonomy LORA (8-9) - future directions**: the goal is to develop systems that are capable of completely independent operations, from probe placement to diagnosis. Indeed, it's crucial to incorporate AI and machine learning for real-time interpretation of ultrasound images and automatic adjustment of probe parameters.

1.2.3. Visual Servoing

Visual servoing is a control technique that utilizes visual information to guide the movements of robots, particularly in tasks requiring precise positioning and interaction with dynamic environments. By integrating real-time feedback from vision sensors, visual servoing enables robots to adjust their actions based on observed changes, enhancing adaptability and accuracy. Robots process live images to extract features such as points, lines,

or textures, which inform control decisions. Visual Servoing techniques are commonly classified into Image-Based Visual Servoing(IBVS) and Position-Based Visual Servoing (PBVS) [10].Both rely on visual feedback to guide robot control but differ fundamentally in their approach to interpreting and using visual data.

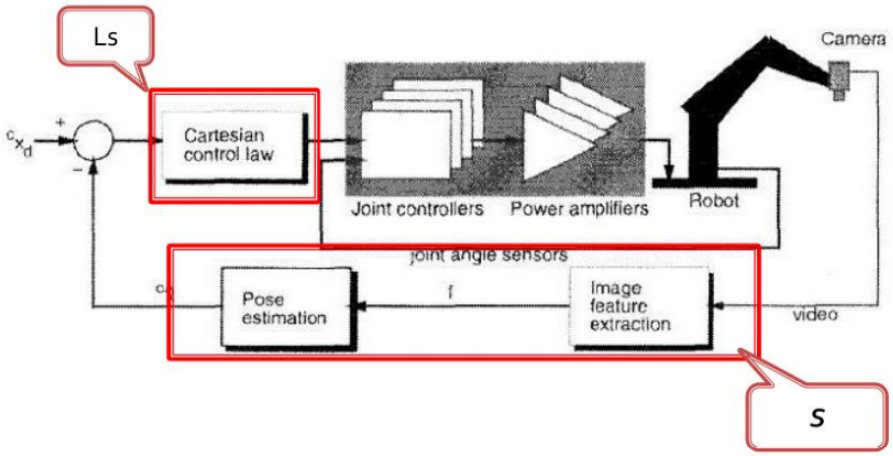


Figure 1.3: Position-Based Visual Servoing

Image-based Visual Servoing uses image features directly extracted from the camera's image plane (e.g., points, lines, contours) to control the robot, on the other hand Position-Based Visual Servoing reconstructs the 3D pose (position and orientation) of the target object in Cartesian space and computes the robot's control commands in the 3D space. There are two types of camera configuration Eye-in-Hand in which the camera is mounted on the robot's end-effector, providing a dynamic point of view and Eye-to-Hand in which the camera is fixed in the environment, observing the robot's workspace from a stationary position.

Visual servoing is utilized in various domains, including industrial automation, medical robotics, and autonomous navigation, where precise control relative to visual targets is essential. Incorporating visual servoing into robotic systems enhances their ability to interact with complex and changing environments, making it a pivotal technology in advancing autonomous and semi-autonomous robotic applications.

In this thesis, we explore a novel application of visual servoing that leverages ultrasound imaging as the primary visual feedback modality, particularly suited for tasks in medical robotics where conventional optical vision is limited or infeasible. Unlike traditional

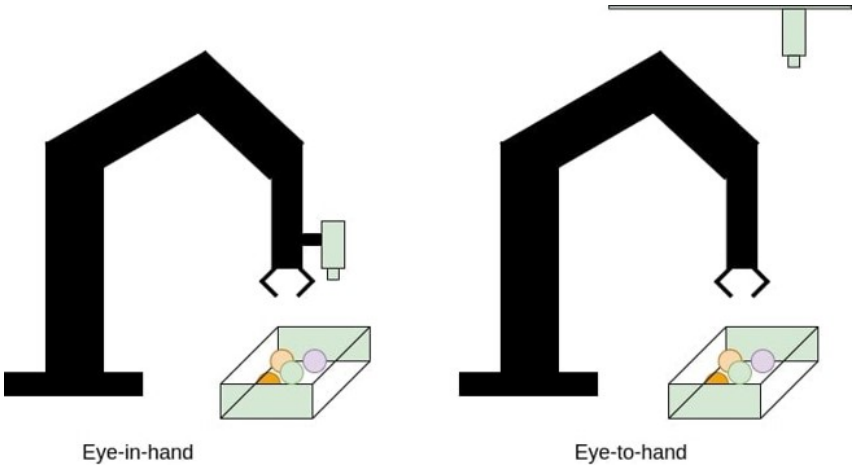


Figure 1.4: Eye-in-Hand vs Eye-to-Hand

visual servoing methods that rely on optical cameras, ultrasound-based visual servoing utilizes real-time ultrasound scans to track internal anatomical features, such as tissues or organs, during robotic procedures. Ultrasound images are processed to extract relevant features—such as contours, landmarks, or intensity patterns—that guide the robot's movement toward a desired target within the patient's body. This approach enables precise positioning in minimally invasive interventions, where visual access is restricted and tissue motion (e.g., due to breathing or heartbeat) introduces dynamic complexity. The integration of ultrasound imaging into the visual servoing loop presents unique challenges, including low signal-to-noise ratio and image artifacts, but also offers the critical advantage of real-time, non-ionizing, deep-tissue visualization, making it a powerful tool for safe and adaptive robotic assistance in clinical environments.

1.3. Motivation and Contribution

The use of robotic systems in ultrasound imaging has shown great potential in addressing critical clinical challenges, such as operator dependency, low repeatability of images, and the physical strain placed on sonographers. In the specific case of vascular ultrasound imaging, precision and consistency are particularly crucial, as slight deviations in the probe's position or orientation can compromise diagnostic quality or result in missed diagnoses. Within this context, my work aims to enhance robotic ultrasound systems by developing a method for real-time detection of vascular structures and identification of their proxy points (i.e., the central points of veins). This approach enables the robotic system to dynamically adapt its trajectory to maintain continuous alignment with the target vessel during scanning. By addressing this critical gap in existing robotic systems, my work provides a more autonomous, accurate, and reliable solution for ultrasound

imaging. The main steps of the project:

- Acquisition of a series of images of the vessel to train the model,in the way to always recognize a vessel when a probe approach the patient
- Using different algorithm for the detection of the proxy point
- Implementation of a control system able to precisely track the proxy point (Region of Interest) and keep it as stable as possible on the central line during all the scanning procedure

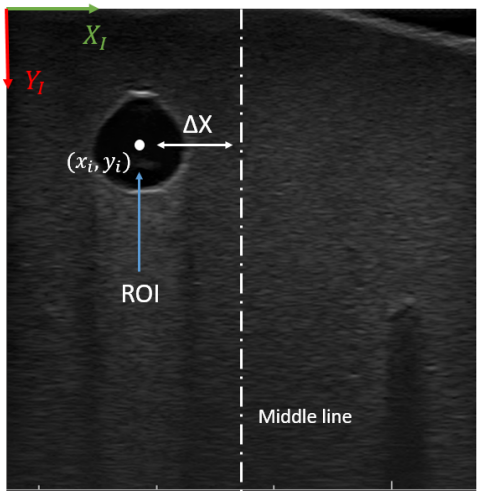


Figure 1.5: Region of Interest and displacement Δx

Following what has been told previously, my contribution is:

- **Real-Time Detection Algorithm:**I developed a robust algorithm to detect veins and identify their proxy points, ensuring that the robot can dynamically follow vascular structures.
- **Enhanced Visual Servoing Control:** I integrated the vein detection system with advanced visual servoing control, allowing the probe's trajectory to be continuously adjusted during scanning.
- **Application-Specific Optimization:** I designed the system to meet the specific requirements of vascular ultrasound imaging, where maintaining the precise alignment of the probe with the vein is essential to ensure diagnostic quality.
- **Experimental Validation:** I conducted experimental evaluations to validate the ability of the system to track vascular structures under various anatomical conditions, demonstrating its feasibility for real-world clinical applications.

The thesis is organized as follows:

- Chapter 2 presents the materials adopted for the project, for the detection of the vessel and then the integration with the robot, and the methods used to achieve the goal.
- Chapter 3 presents the experimental protocol and the performance of the set-up.
- Chapter 4 is characterized by the discussion, final conclusion and future goal.

2 | Materials and Methods

Ultrasound imaging is a widely used technique for vascular assessment due to its non-invasive nature, real-time capabilities, and high availability in clinical settings. However, traditional manual acquisition is highly operator-dependent and subject to variability, which can affect the accuracy and reproducibility of the results. In this work, a robotic system was developed to address these limitations by enabling automated and precise ultrasound imaging for vascular applications.

The proposed system integrates a KUKA LWR IV+ robotic manipulator with a Telemed MicrUs EXT-1H ultrasound probe, combining advanced control strategies and real-time image processing techniques. This integration ensures accurate probe positioning, stable contact with the patient's skin, and controlled scanning trajectories, which are essential for consistent and high-quality image acquisition.

To enhance vascular imaging and facilitate real-time analysis, several image processing and segmentation techniques were implemented. These include deep learning-based approaches, optical flow methods, and real-time object detection algorithms. Each of these methods contributes to the robust identification of vascular structures and the extraction of key features necessary for further processing.

By leveraging robotic automation and advanced imaging techniques, the system aims to improve the reliability and reproducibility of vascular ultrasound examinations, reducing operator dependency and enhancing diagnostic capabilities.

2.1. Robotic System

The robotic system used in this study is based on the KUKA LWR IV+, a lightweight robotic manipulator with seven degrees of freedom (DOFs). Designed for versatility and precision, the LWR IV+ combines advanced mechanical design with state-of-the-art sensing capabilities, making it an ideal platform for robot-assisted medical applications, including vascular ultrasound imaging.



Figure 2.1: KUKA LWR IV+

2.1.1. Technical Specifications

The KUKA LWR IV+ is a compact and lightweight manipulator, weighing approximately 16 kg, and supports a payload of up to 7 kg. The robot's structure consists of seven revolute joints configured in an anthropomorphic layout. This setup allows it to achieve high dexterity and redundancy, making it particularly suitable for tasks requiring precise control in confined or complex environments, such as medical imaging.The main axis components are denoted by A1, A2, E1, A3, A4, A5, A6. Figure 2.2 shows the robot in its zero position, together with the link frames chosen according to the Denavit-Hartenberg convention. Each joint is powered by brushless motors that are coupled to harmonic drives to ensure smooth, precise, and backlash-free movements. Additionally, the joints are equipped with integrated sensors for torque and position measurements, enabling accurate feedback control and ensuring the robot's suitability for tasks that require high sensitivity to external forces. Unlike classic industrial robots, the KUKA LWR IV+ was designed to be lighter and more versatile[11].

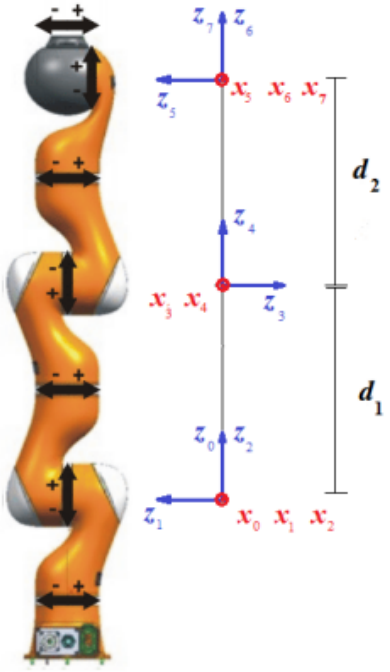


Figure 2.2: KUKA LWR-IV+[12]

In the following table are reported joint position limits and joint torque limits.

joints	q_{min} (degree)	q_{max} (degree)	τ_{max} Nm
A1	-170	170	176
A2	-120	120	176
E1	-170	170	100
A3	-120	120	100
A4	-170	170	100
A5	-120	120	38
A6	-170	120	38

Table 2.1: KUKA LWR IV+ joints position and torque limits

The manipulator architecture mirrors that of a human arm, with three joints forming the shoulder, one joint representing the elbow, and three additional joints forming the wrist. This configuration grants the robot six DOFs necessary for controlling the end-effector's position and orientation in 3D space, with the seventh DOF providing redundancy for resolving constraints, such as avoiding obstacles or optimizing force application.

Control and Interface

The robot operates through the KUKA Robot Controller (KRC), which manages all motion and peripheral interactions. The KRC supports real-time control via Ethernet-

based communication, enabling integration with external computing systems and custom software. The controller also supports the KUKA Robot Language (KRL), which provides scripting capabilities for defining robot operations and communication protocols. Below a short description of the KUKA Robot Controller (KRC) and if the KUKA Control Panel (KCP).

- **KUKA Control Panel (KCP):** it serves as the robot's manual interface.The KCP includes a graphical user interface and hardware buttons for joint-level or Cartesian control. It offers three primary modes of operation:

  1. Manual Mode: allows direct control of joints or Cartesian movements using predefined velocity settings (T1 and T2).
  2. Gravity Compensation Mode: enables the operator to freely move the robot's arm without resistance, useful for calibrations or trajectory demonstrations.
  3. Automatic Mode: enables the operator to freely move the robot's arm without resistance, useful for calibrations or trajectory demonstrations.



Figure 2.3: KUKA Control Panel (KCP)

- **KUKA Robot Controller (KRC):** The KRC is the central unit responsible for managing and controlling the movements of the KUKA robot. It integrates both hardware and software components, allowing precise real-time control of the robot's actions. The KRC communicates with the robot's motors and sensors via an Ethernet connection, executing movement algorithms and ensuring smooth, accurate operations.

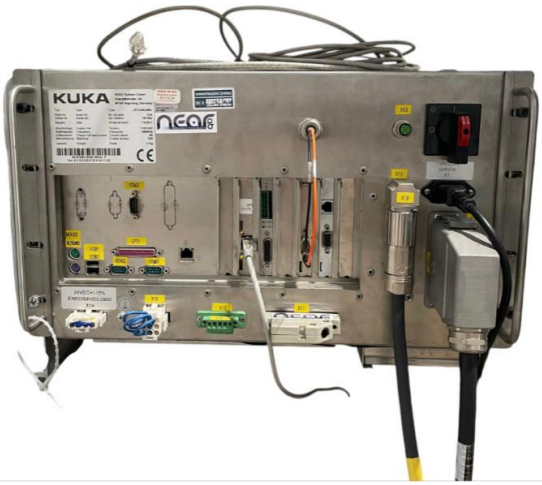


Figure 2.4: KUKA Robot Controller (KRC)

2.1.2. Software Environment

The System integrates several software tools programming, control and data processing.

Fast Research Interface (FRI): FRI is a real-time communication library that allows direct interaction with the robot's motion kernel, enabling low-latency control over the manipulator. The FRI uses a UDP (User Datagram Protocol) socket for communication, balancing real-time performance and reliability.

Robot Operating System (ROS)[13]: ROS plays a central role in the software architecture, providing a modular framework for developing, deploying, and managing control algorithms. Its open-source nature and extensive library ecosystem make it particularly well-suited for robot-assisted medical applications. In this study, ROS facilitates communication between the robotic manipulator and other system components through a publisher-subscriber model, ensuring robust and scalable data exchange. The ROS nodes used in this project communicate with the robot at 100 Hz on a computer with Ubuntu 16.04 LTS operating system, allowing for efficient data exchange and control execution.

Kinematic and Dynamics Library (KDL): KDL is employed for real-time computation of the robot's forward kinematics and Jacobian matrices, which are essential for trajectory planning and control.

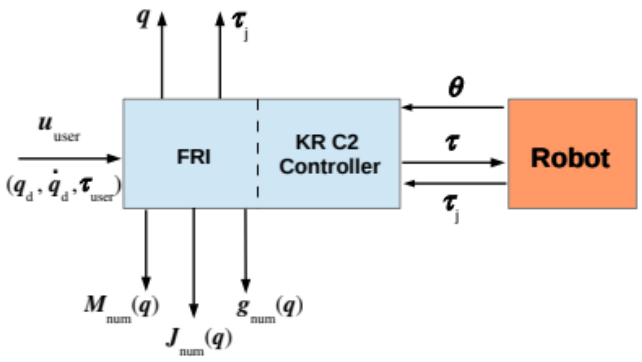


Figure 2.5: Signal flow between KUKA LWR IV+ and the KRC controller

2.2. Ultrasound Imaging System

The Telemed MicrUs EXT-1H is a highly portable and versatile ultrasound probe that I utilized in my work for robot-assisted vascular ultrasound imaging and reconstruction. This compact device is designed for high-resolution imaging and provides real-time data acquisition, making it suitable for applications in medical diagnostics, research, and robotic integration.



Figure 2.6: Ultrasound Probe: Telemed MicrUs EXT-1H

2.2.1. Device Overview

The MicrUs EXT-1H is a USB-powered ultrasound probe developed by Telemed, a leading company in portable ultrasound technology. It supports high-quality B-mode imaging with advanced signal processing and is equipped with a linear transducer, making it ideal for superficial imaging applications such as vascular studies. Its compact and lightweight design ensures easy integration with robotic systems, while its compatibility with standard imaging software offers flexibility in data acquisition and analysis.

2.2.2. Technical Specifications

The key technical specifications of the MicrUs EXT-1H include:

- **Frequency Range:** Typically ranging between 5 and 12 MHz, providing high-resolution imaging suitable for superficial tissues and vascular structures.
- **Image Modes:** Supports B-mode imaging with customizable parameters such as depth, gain, and dynamic range.
- **Connectivity:** Utilizes USB 2.0 or USB 3.0 for power and data transfers, simpli-

fying integration wirh computers and robotic platforms.

- **Dimensions and Weight:** This device is portable an lightweight which is ideal for procedures where size and portability are critical.

In this work, the MicrUs EXT-1H was mounted on the robotic arm to facilitate real-time ultrasound imaging during the scanning process. Its lightweight design allowed for precise manipulation by the robotic system, while its USB connectivity ensured seamless data transfer to the processing unit. The probe's high-frequency transducer enabled the acquisition of high-resolution images of vascular structures, which were critical for subsequent image segmentation, reconstruction, and navigation.

2.2.3. HDMI Image Acquisition via Magewell USB Capture Gen 2



Figure 2.7: Magewell USB Capture Gen 2

Due to the incompatibility of the Telemed MicrUs EXT-1H ultrasound probe with Linux-based systems, direct integration with the Ubuntu workstation—used for robot control and image processing—was not feasible. To address this, a hybrid acquisition pipeline was implemented using a Windows PC.

On the Windows machine, a custom Python script developed in PyCharm was used to interface with the Telemed API and acquire real-time ultrasound images from the probe. The script also included the drawing of the vertical central line of the ultrasound image window. This visual marker was added to facilitate downstream processing on the Ubuntu system, specifically to compute the distance between the central vertical line of the frame and the detected center of the target vein.

The processed video output, including the B-mode ultrasound and the overlaid central line, was sent via HDMI output from the Windows PC. This HDMI signal was then

captured using a Magewell USB Capture HDMI Gen 2 device, which provided a plug-and-play USB 3.0 interface compatible with Ubuntu. On the Ubuntu side, the Magewell device exposed the video stream as a standard webcam input, allowing for easy frame-by-frame acquisition using OpenCV.

This setup enabled the Ubuntu control node to receive and analyze the incoming video stream in real time, performing vessel center detection and calculating its displacement from the image's central axis. The Magewell USB Capture HDMI Gen 2 was selected for its reliability, low latency, and cross-platform compatibility, ensuring a stable and efficient data transfer between systems.

2.3. Ultrasound image segmentation and ROI tracking

Image segmentation is a fundamental step in computer vision and medical imaging, as it involves partitioning an image into meaningful regions corresponding to specific structures of interest. In vascular ultrasound imaging, segmentation plays a crucial role in isolating vascular structures from surrounding tissue, enabling accurate vein detection and localization. This process is essential for subsequent tasks such as trajectory planning and real-time probe adjustment, which are key for robotic-assisted scanning.

Tracking the Region of Interest (ROI), such as the central point of a vein, is equally important to ensure that the robotic system maintains continuous alignment with the target structure during scanning. Inconsistent tracking can lead to poor image quality, misdiagnosis, or failure to fully capture the target area. Real-time ROI tracking allows for dynamic adjustments of the probe's position and orientation, enabling the system to compensate for anatomical variations and patient movements.

To achieve robust and reliable segmentation and ROI tracking, this work employs a combination of deep learning-based segmentation, motion estimation techniques, and real-time object detection algorithms. The following methods were implemented:

- **U-NET[14]:**A convolutional neural network architecture widely used for medical image segmentation. U-Net is employed to automatically segment vascular structures in ultrasound images, providing a precise and consistent representation of veins.
- **Optical Flow[15][16]:**Both dense and sparse (Lucas-Kanade) optical flow techniques are used to estimate the motion of vascular structures in consecutive ul-

trasound frames. This information is leveraged to dynamically adjust the probe's trajectory, ensuring smooth and continuous tracking.

- **YOLO (You Only Look Once)[17]:**A real-time object detection algorithm applied to identify and localize veins within the ultrasound image. Its speed and efficiency make it particularly suitable for real-time robotic-assisted scanning.

2.3.1. U-NET

The first method that I adopted in my work for segmentation is U-Net (U-shaped Network). U-Net is a convolutional neural network architecture specifically designed for image segmentation tasks. First introduced in 2015 by Olaf Ronneberger et al.[14], U-Net has become a benchmark in medical imaging due to its ability to achieve precise segmentation with limited training data. Its architecture is highly effective for biomedical applications, where accurate delineation of anatomical structures is crucial for diagnosis, treatment planning, and robotic interventions.

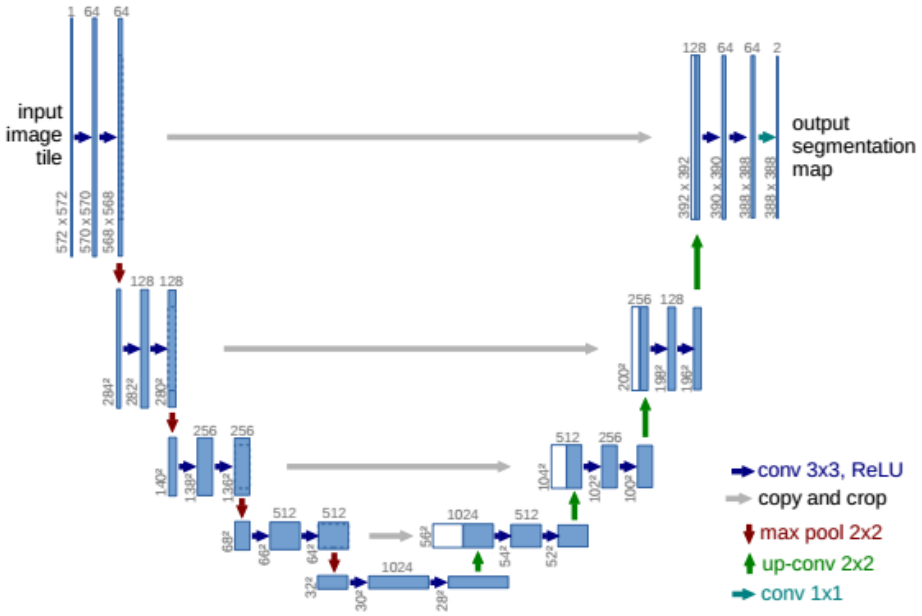


Figure 2.8: U-net architecture

In figure 2.5 is depicted the U-net architecture in which each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

U-net follows an encoder decoder structure:

- **Encoder Path (Contracting Path):** this part captures the context of the image by applying successive convolutional and max-pooling layers. It reduces the spatial dimensions while increasing the depth of feature maps, allowing the network to learn high-level abstract features.
- **Decoder Path (Expanding Path):** this reconstructs the segmentation map by up-sampling feature maps and applying transposed convolutions. It restores spatial resolution while integrating contextual information learned by the encoder.
- **Skip Connections:** These connect corresponding layers in the encoder and decoder paths, ensuring that fine-grained spatial details from early layers are preserved in the segmentation process. This feature is particularly valuable for medical imaging, where small structures need to be accurately segmented.

In medical imaging, segmentation is essential for extracting meaningful information from complex data. For tasks like vascular ultrasound imaging, precise segmentation of veins ensures that robotic systems can follow target structures accurately. U-Net's ability to delineate boundaries and detect subtle features is critical for identifying vascular structures and tracking proxy points.

Furthermore, U-Net has been widely validated in literature and clinical studies, demonstrating its reliability across numerous use cases, including tumor segmentation, organ delineation, and vascular imaging. Its robust performance and adaptability make it a gold standard for segmentation tasks in medical imaging.

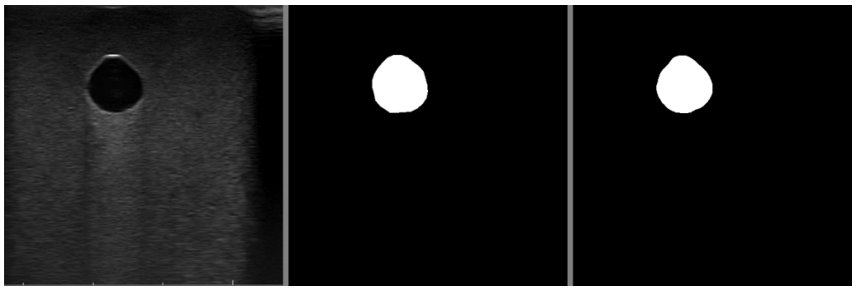


Figure 2.9: Vessel Segmentation Results using U-Net

U-net is a good choice in medical imaging for different reasons:

1. **Accuracy with limited data:** medical imaging datasets are often small due to the high cost and complexity of data annotation. U-Net's architecture is designed to perform well with relatively few labeled samples.
2. **Localization and Precision:** the skip connections in U-Net allow the model to

combine high-level semantic information with low-level spatial information, resulting in precise localization of structures. This is critical in medical imaging where even slight inaccuracies can impact clinical outcomes.

3. **Versatility:** U-Net is highly adaptable and can be fine-tuned for various imaging modalities (e.g., ultrasound, Magnetic Resonance Imaging (MRI), Computed Tomography (CT)). Its generalizability makes it a reliable choice across different medical applications.

4. **Real-Time capability:** U-Net can produce segmentation maps quickly, making it suitable for real-time applications such as guiding robotic systems during procedures.

5. **Robustness to noise:** U-Net is resilient to noise and variations in imaging conditions, such as the speckle noise often present in ultrasound images.

2.3.2. Optical Flow

Optical flow is a technique used in computer vision to estimate the motion of objects, surfaces, or edges between consecutive frames in a sequence. It calculates the apparent motion of pixels in an image, providing information about how points in a scene change over time. In medical imaging, and particularly in robotic-assisted ultrasound applications, optical flow is invaluable for tracking dynamic structures, such as veins, in real time.

In the context of vascular ultrasound imaging, where both patient movement and probe motion can introduce variability, optical flow helps to ensure that the Region of Interest (ROI) is consistently tracked. By analyzing pixel motion, it is possible to dynamically adjust the probe's trajectory and orientation, enabling robust scanning performance even in challenging conditions.

In the robotic-assisted ultrasound system, where maintaining alignment with vascular structures is critical, optical flow provides a complementary approach to segmentation. By analyzing motion information, optical flow can refine the tracking process, ensuring that the probe remains focused on the vein despite patient movement or other disruptions. Both dense[15] and sparse[16] optical flow methods were explored in this study to balance accuracy and computational efficiency, ensuring reliable performance across different scenarios.

Optical flow is particularly suited for applications like vein tracking because:

1. **Real-Time Capability:** it provides a frame-by-frame estimation of motion, mak-

ing it ideal for real-time systems.

2. **Complementarity:** optical flow complements segmentation methods like U-Net, offering a motion-based perspective to refine ROI tracking.
3. **Robustness to variability:** it can handle small and gradual motions caused by probe adjustments or patient breathing, maintaining accurate tracking of vascular structures.
4. **Low computational overhead:** using sparse optical flow with the Lucas-Kanade method it simplifies optical flow techniques, making them useful in scenarios with limited processing power.

Now let's focus on the the two major approaches that I adopted for this project, dense and sparse (Lucas-Kanade) optical flow.

Sparse Optical Flow focuses on estimating motion for a limited set of significant points or features within an image, rather than calculating motion for every pixel (as in dense optical flow). In my case the key points are estimated on the external surface of the vessel, and then based on these key points I track the center of the vessel. This approach is computationally more efficient and is particularly useful when tracking specific regions or points of interest, such as the central axis of a vein in vascular ultrasound imaging. One of the most widely used algorithms for sparse optical flow is the Lucas-Kanade method, which provides a reliable and computationally efficient way to estimate motion between consecutive frames.

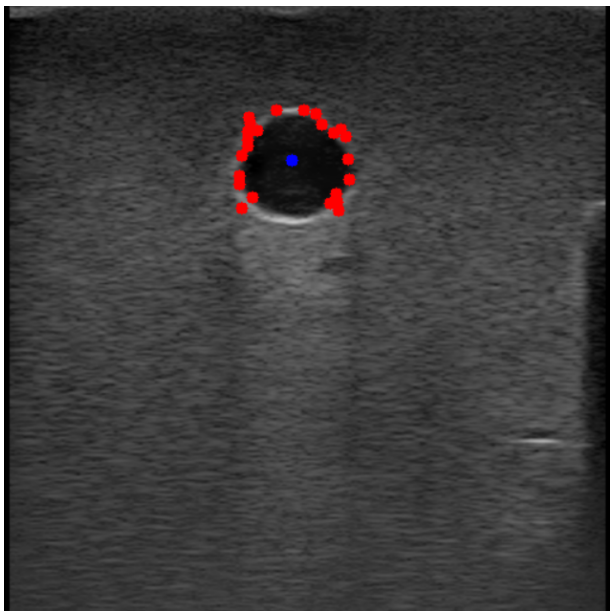


Figure 2.10: Vessel Tracking with Sparse Optical Flow

The Lucas-Kanade method is based on the assumption that the motion within a small local neighborhood of pixels is approximately constant. By analizing the intensity patterns in this local region, the algorithm estimates the motion of selected feature points.

The core idea of optical flow[18] relies on the brightness constancy assumption, which states that the intensity of a pixel remains constant over time:

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \tag{2.1}$$

where $I(x, y, t)$ is the intensity of pixel at position (x, y) at time t. Using a first-order Taylor expansion, this can be seen as:

$$\frac{\delta I}{\delta x}u + \frac{\delta I}{\delta y}v + \frac{\delta I}{\delta t} = 0 \tag{2.2}$$

where:

- $\frac{\delta I}{\delta x}$ and $\frac{\delta I}{\delta y}$ are the spatial gradients of the image.
- $\frac{\delta I}{\delta t}$ is the temporal gradient.
- u and v are the horizontal and vertical components of the optical flow (motion vector).

This equation represents one constraint per pixel, but since there are two unknowns (u, v) the system is undetermined. To solve this, the Lucas-Kanade method[19][20] considers a local window around pixels around each feature point and assumes that all pixels within the window have the same motion (u, v) This results in an overdetermined system of linear equations, which is solved using least squares:

$$A \cdot v = b \tag{2.3}$$

where:

- A is the matrix of spatial gradients:

$$A = \begin{bmatrix} \frac{\delta I}{\delta x} & \frac{\delta I}{\delta y} \\ \vdots & \vdots \\ \frac{\delta I}{\delta x} & \frac{\delta I}{\delta y} \end{bmatrix}$$

- $[u, v]^T$ is the optical flow vector

- b is the temporal gradient vector:

$$b = \begin{bmatrix} -\frac{\delta I}{\delta t} \\ \vdots \\ -\frac{\delta I}{\delta t} \end{bmatrix}$$

The solution for v is obtained as:

$$v = (A^T A)^{-1} A^T b \tag{2.4}$$

The Lucas-Kanade method is a powerful tool for sparse optical flow and is well-suited for tasks like vein tracking in ultrasound imaging. Its computational efficiency, robustness to noise, and ability to focus on relevant features make it an essential component for my project, ensuring reliable tracking of vascular structures in real-time.

Dense Optical Flow is a fundamental concept in computer vision and motion analysis, which estimates the motion of every pixel in an image sequence. Unlike sparse optical flow, where motion is tracked at selected feature points, dense optical flow calculates the displacement of each pixel between consecutive frames, providing a complete flow field that represents motion across the entire image.

It assumes that the brightness of the pixels remains constant between frames (equation 2.4) and that the movement between frames is small and smooth. The goal is to compute a flow vector for each pixel, which consists of the displacement in both the x and y directions.

The result of this process is a flow field where each pixel has an associated motion vector that describes its displacement between two frames. Dense optical flow is particularly useful when the motion of the entire scene is important and precise tracking is required for every part of the image.

There are several methods to compute dense optical flow, the most widely used are Horn-Schunck and Farneback. In this project I prefer the use of Farneback because it handles complex, non-rigid motions better, making it ideal for medical imaging where tissue and organ deformation needs to be tracked. It is more robust to noise and computationally efficient, enabling real-time processing. Additionally, it provides detailed flow information for large displacements, which is essential in applications like ultrasound imaging. Farneback method computes the optical flow using a polynomial expansion of the image signal. The image is first downsampled using a Gaussian pyramid, and then optical flow is computed at each level of the pyramid.

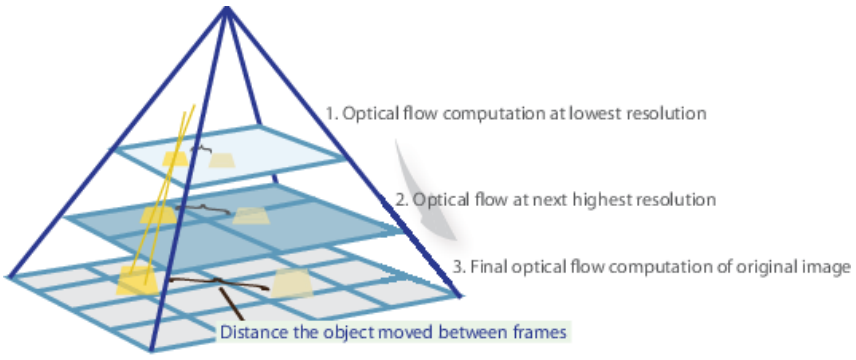


Figure 2.11: Farneback Pyramid

The Farneback algorithm[21] computes dense optical flow by first constructing a Gaussian pyramid of the input images, where each level represents a progressively downsampled version of the image. This pyramid structure allows the algorithm to capture motion at different scales and speeds, making it robust to large displacements. For each pixel, the algorithm approximates the local image neighborhood using a quadratic polynomial expansion, which models the motion in that region. At the coarser levels of the pyramid, the flow vectors are estimated more easily, as the motion is less complex. These initial estimates are then refined at higher resolution levels, where the flow vectors are propagated from the coarser to finer levels. This hierarchical refinement results in a smooth, dense flow field that accurately represents pixel displacement from one frame to the next. By using this multi-scale approach, Farneback is able to handle both large and subtle motion, making it particularly effective in applications like medical imaging where accurate tracking of complex movements is required.

In conclusion, the choice between dense and sparse optical flow depends on the application and the type of motion analysis required. Dense optical flow, as demonstrated by the Farneback algorithm, estimates motion for every pixel in the image, providing a complete and detailed representation of motion fields. This makes it highly suitable for tasks where subtle or complex motion patterns, such as tissue deformation in medical imaging, need to be captured accurately. On the other hand, sparse optical flow, such as the Lucas-Kanade method, focuses on tracking a set of distinctive points or features, which is much less computationally intensive but more limited in capturing global motion.

2.3.3. YOLO/CNN

The YOLO (You Only Look Once)[22][17] algorithm is a real-time object detection method that reframes detection as a single regression problem. Unlike traditional systems that rely on multiple stages, such as region proposals and separate classification, YOLO directly predicts bounding boxes and class probabilities in a single evaluation of a convolutional neural network. The image is divided into a grid, and each cell in the grid predicts bounding boxes, their confidence scores, and class probabilities, enabling the system to reason globally about the entire image.

YOLO is exceptionally fast, processing images at 45 frames per second in its base model, with a smaller, optimized version achieving 155 frames per second. Its unified architecture also ensures end-to-end optimization for detection performance. While YOLO is slightly less precise in localizing small objects compared to region-based methods, it significantly reduces false positives and generalizes well to new domains, making it a robust choice for real-time detection tasks.

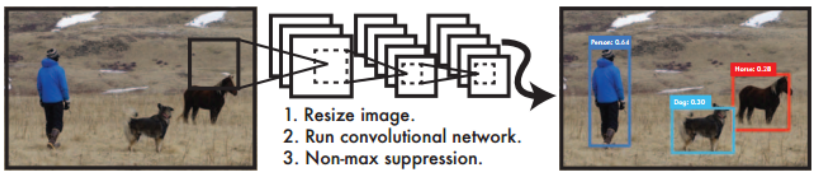


Figure 2.12: YOLO structure

Network Design

The YOLO network is a convolutional neural network (CNN) designed for end-to-end object detection. Its architecture combines feature extraction, bounding box prediction, and classification into a single model. The network consists of 24 convolutional layers followed by 2 fully connected layers. The convolutional layers are responsible for extracting features from the image, while the fully connected layers predict bounding box coordinates, confidence scores, and class probabilities.

The network design draws inspiration from GoogLeNet, using alternating 1×1 and 3×3 convolutional layers to reduce dimensionality and improve computational efficiency. For detection, the input image is divided into an $S \times S$. Each cell predicts B bounding boxes, their associated confidence scores, and C class probabilities. These predicitions are encoded as a tensor of size $S \times S \times (B \times 5 + C)$, where "5" corresponds to bounding box coordinates (x, y, w, h) and confidence.

YOLO reasons globally about the image during both training and inference, which reduces redundancy and ensures efficient use of the image context. By directly linking image pixels to predictions, YOLO achieves fast detection speeds and maintains a high level of accuracy.

Training

The training process involves optimizing a loss function that captures three key objectives: bounding box regression, confidence prediction, and classification accuracy. The loss function is defined as:

$$\begin{aligned}\text{Loss} = \lambda_{coord} &\sum_{i=1}^{S^2}\sum_{j=1}^{B} 1_{ij}^{obj}\left[(x_i-\hat{x}_i)^2+(y_i-\hat{y}_i)^2+(\sqrt{w_i}-\sqrt{\hat{w}_i})^2+(\sqrt{h_i}-\sqrt{\hat{h}_i})^2\right] \\ &+\sum_{i=1}^{S^2}\sum_{j=1}^{B} 1_{ij}^{obj}(C_i-\hat{C}_i)^2 \\ &+\lambda_{noobj}\sum_{i=1}^{S^2}\sum_{j=1}^{B} 1_{ij}^{noobj}(C_i-\hat{C}_i)^2 \\ &+\sum_{i=1}^{S^2} 1_{i}^{obj}\sum_{c\in\text{classes}}(p_i(c)-\hat{p}_i(c))^2.\end{aligned}$$

where:

- 1_{ij}^{obj} is 1 if the j-th boundig box in grid cell i responsible for predicting an object, and 0 otherwise.
- (x, y, w, h) are the center coordinates, width, and height of the bounding box.
- C_i is the confidence score, and $p_i(c)$ represents the class probabilities.

Key aspects of the training process:

1. **Bounding Box Prediction:**The loss penalizes the difference between predicted and ground truth box coordinates. To emphasize accuracy, the width and height are predicted as square roots to reduce the impact of large bounding boxes.
2. **Confidence Loss:** The model learns to predict the confidence score as the Intersection over Union (IoU) between the predicted and ground truth boxes.
3. **Class Prediction:** The network predicts conditional class probabilities for objects within each grid cell.

To improve stability during the training, YOLO increases the loss for coordinate predictions ($\lambda_{coord} = 5$ and reduces the loss for confidence predictions for cells without object ($\lambda_{noobj} = 0.5$). YOLO is pretrained on the ImageNet dataset for feature extraction, and then fine-tuned for object detection tasks using dataset like PASCAL VOC. During training, extensive data augmentation, such as random scaling, translation, and color adjustments, is used to enhance robustness. The network also employs dropout layers to prevent overfitting. This streamlined design and training approach enable YOLO to detect objects efficiently while maintaining high accuracy across a variety of real-world scenarios.

This is the core principal of YOLO, the first one that was developed. The one that I use is YOLOv8 nano, which is a little bit different and updated but the general principles remains the same. YOLOv8 is part of the latest evolution of YOLO models and incorporates improvements in architecture, efficiency, and training methods. YOLOv8 Nano, in particular, is optimized for lightweight, resource-efficient deployment without sacrificing much accuracy, making it ideal for real-time applications and devices with limited computational power.

While the original YOLO used a single convolutional network with 24 convolutional layers and 2 fully connected layers, YOLOv8 Nano employs a lightweight CSPNet-based (Cross-Stage Partial Net-based) architecture optimized for speed and resource efficiency. YOLOv8 nano also uses a decoupled detection head for separate classification and regression tasks, an anchor-free design for simpler bounding box prediction, and modern techniques like depthwise separable convolutions to reduce computational load. In training, YOLOv8 nano incorporates advanced loss functions (e.g., CIoU), Mosaic augmentation, and transfer learning to improve generalization and performance, whereas the original YOLO relied on simpler squared-error-based loss and basic augmentation. These advancements make YOLOv8 Nano significantly faster, more accurate, and better suited for real-time applications on low-power devices.

2.4. System calibration and Coordinate Transformation

In robotic ultrasound imaging, precise system calibration and coordinate transformation[23] are essential to ensure accurate spatial alignment between the robotic manipulator, the ultrasound probe, and the acquired images. This section describes the transformation framework and the calibration procedure used in my system.

The system consists of multiple coordinate frames:

- **{B} - Robot Base Frame:** The fixed reference frame attached to the base of the robotic arm.
- **{E} - End-Effector Frame:** The coordinate system attached to the last link of the robotic arm, where the probe is mounted.
- **{P} - Probe Frame:** The coordinate frame located at the ultrasound probe tip.
- **{I} - Ultrasound Image Frame:** The 2D coordinate system representing the ultrasound image.

A series of transformations are required to map a point in the ultrasound image to the robot's frame base. The transformation from the robot base to the ultrasound image can be decomposed as follows:

$$ {}^{B}_{I}T = {}^{B}_{E}T * {}^{E}_{P}T * {}^{P}_{I}T * P^{I} \tag{2.5} $$

where:

- ${}^{B}_{I}T \in \mathbb{R}^{4\times4}$ represents the transformation from robot base {B} to US image {I}.
- ${}^{B}_{E}T \in \mathbb{R}^{4\times4}$ is the transformation from robot base {B} to the end-effector {E}.
- ${}^{E}_{P}T \in \mathbb{R}^{4\times4}$ is the transformation from the end-effector {E} to the probe tip {P}.
- ${}^{P}_{I}T \in \mathbb{R}^{4\times4}$ maps the probe tip {P} to the ultrasound image plane {I}.

The ultrasound image plane is represented by points P^{I} in homogeneous coordinates:

$$ P^{I} = \begin{bmatrix} x_I \\ y_I \\ 0 \\ 1 \end{bmatrix} $$

The transformation from the probe to the ultrasound image is given by:

$$ {}^{P}_{I}T = \begin{bmatrix} \frac{L_p}{L_I} & 0 & 0 & -\frac{L_P}{2} \\ 0 & 0 & -1 & 0 \\ 0 & \frac{D_I}{W_I} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Where:

- L_I, W_I are the length and width of the ultrasound image (in pixels).
- D_I is the depth of the ultrasound image.
- L_P is the length of the probe.

Calibration plays a fundamental role in visual servoing by ensuring accurate vein tracking, compensating for motion errors and enabling precise closed-loop control. Through proper transformation can consistently follow the detected veins in the ultrasound image while maintaining correct positioning in real-world coordinates. Additionally, calibration minimizes errors caused by probe misalignment or variations in imaging parameters, enhancing the reliability of the system. By accurately mapping images features to the robot's coordinate frame, the system can dynamically adjust its motion, allowing for continuous and precise navigation throughout the scanning process.

2.5. Robot Visual Servoing Control

Robotic Ultrasound Imaging requires precise and adaptive control strategies to ensure accurate probe positioning and consistent contact with the patient's skin. Visual Servoing Control is a widely used approach in robotic systems that leverages real-time image feedback to dynamically adjust the robot's movements. By continously analyzing ultrasound images, the system can autonomously adapt to anatomical variations, patient movements, and external disturbances, enhancing both scanning accuracy and repeatability.

In this chapter, the implementation of a visual servoing-based control strategy for robotic-assisted vascular ultrasound imaging is presented. The proposed approach utilizes image-based feedback to regulate the motion of the robotic manipulator, ensuring that the ultrasound device maintains proper alignment with the target vascular structure. Various control schemes are explored to achieve robust trajectory tracking and real-time adjustments, enabling autonomous and reliable ultrasound scanning.

The following section details the methodology adopted, including the integration of im-

age processing techniques with robotic control, the formulation of control laws, and the evaluation of system performance in dynamic scanning conditions.

2.6. Position controller

Position control is a widely used control strategy in robotic systems, where the primary objective is to guide the robot along predefined trajectories with high spatial accuracy. Unlike hybrid force/position control—which simultaneously regulates position along certain directions and force along others—position control focuses solely on achieving and maintaining the desired position across all degrees of freedom. This simplifies the control architecture, making it particularly suitable for scenarios where interaction with the environment is limited or does not require active force regulation.

In this work, position control is employed to guide the robotic manipulator along specified trajectories in order to perform precise movements within the designated workspace. The trajectory is generated through temporal interpolation, resulting in a continuous sequence of target positions, which are then tracked using a closed-loop PI controller. This approach ensures high accuracy in trajectory execution while maintaining system stability, even in the presence of small disturbances or model uncertainties.

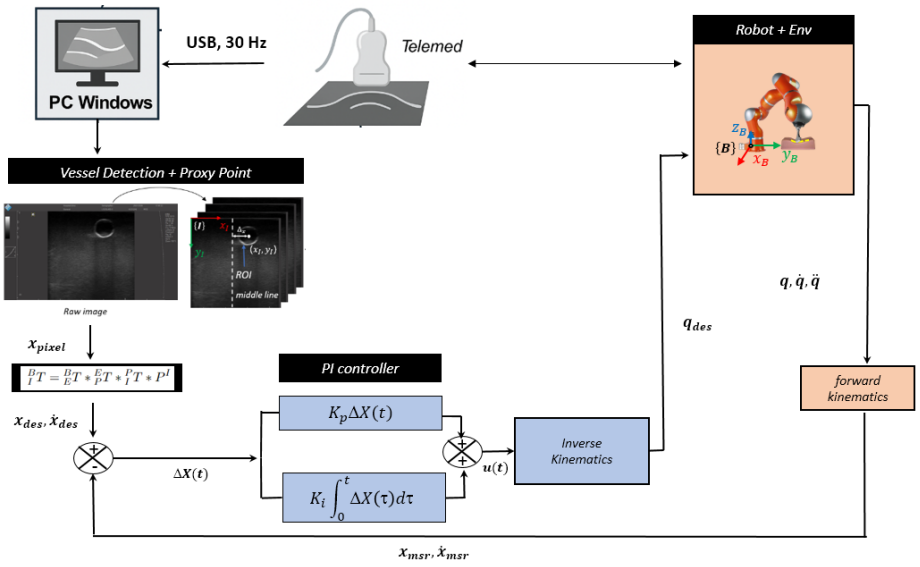


Figure 2.13: Visual-servoed control architecture for robot-assisted vascular ultrasound imaging

2.6.1. Close-loop Control

In robotic control systems, the Proportional-Integral (PI) controller is a widely adopted feedback control strategy due to its implementation simplicity, robustness to disturbances, and effectiveness in minimizing steady-state errors. A PI controller generates the control input based on the current and accumulated tracking error, offering both immediate corrective action and long-term compensation for bias. Mathematically, the control law is defined as:

$$u(t) = K_p \cdot \Delta X(t) + K_I \int_0^t \Delta X(\tau)\, d\tau \tag{2.6}$$

where:

- $\Delta X(t)$ is the error between the desired and current value (e.g., position of the vessel center in the ultrasound image)
- K_P is the proportional gain, controlling the response to the current error
- K_I is the integral gain, eliminating steady-state errors by considering the cumulative error over time.

In the context of this work, the PI controller is implemented within the Image-Based Visual Servoing (IBVS) control loop. The main objective is to maintain the alignment between the detected position of the vessel (or its visual proxy) and the center of the ultrasound image frame. The image-space error, computed in real-time, serves as the input to the PI controller.

The proportional term provides a fast corrective response to any instantaneous deviation, making the system responsive and reactive to rapid changes in the visual input. The integral term, on the other hand, addresses longer-term discrepancies caused by factors such as mechanical imperfections, unmodeled dynamics, sensor noise, or calibration drift. By integrating the error over time, the controller compensates for biases that would otherwise prevent convergence to the desired target.

Once the control signal is computed in the image space, it is transformed into Cartesian velocity commands through a coordinate transformation pipeline, which maps image coordinates to robot base frame coordinates. This ensures that the robotic end-effector's motion is spatially consistent with the error observed in the ultrasound image, allowing

accurate and stable positioning during the visual servoing process.

The use of a PI controller in this setup strikes a balance between responsiveness and accuracy, making it particularly suitable for real-time, closed-loop control applications such as robotic navigation based on medical imaging.

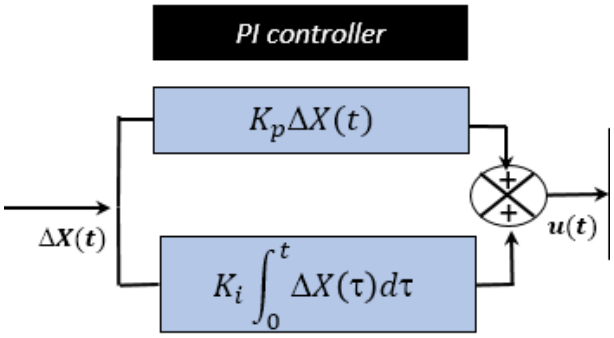


Figure 2.14: Control Scheme of a PI controller

3 | Experiments

3.1. Experimental Setup

To assess the performance of the proposed vessel tracking and visual servoing strategies, a set of three controlled experiments was conducted using a vascular phantom. The experiments were designed to evaluate the system's ability to track a moving vessel under varying motion dynamics and levels of automation.

In the first two experiments, a linear actuator was integrated into the setup to simulate consistent vessel motion along the horizontal (X) axis[24]. The actuator was driven by a stepper motor controlled via an Arduino board, enabling precise control over the phantom's movement. The ultrasound probe, mounted on the KUKA LWR IV+ robotic arm, was used to acquire real-time images of the vessel during the experiments.

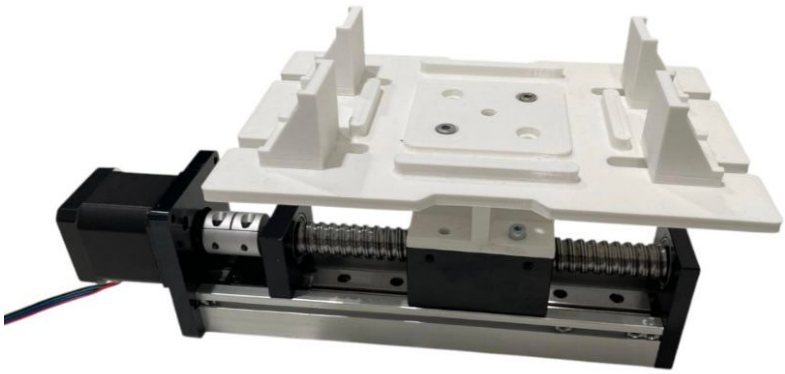


Figure 3.1: Actuator for Vascular Phantom Motion Simulation

- **Experiment 1: Visual Servoing Tracking Under Linear External Disturbance**

  The phantom was programmed to move along a straight horizontal path with a constant velocity of 2 mm/s. The total travel distance was 2.5 cm in each direction (forward and backward), and the full cycle was repeated three times to assess tracking repeatability and stability under uniform motion conditions.

- **Experiment 2: Visual Servoing Tracking Under Sinusoidal External Disturbance**

  To introduce a more complex motion profile, the phantom followed a sinusoidal trajectory along the same horizontal axis. The average speed was increased to 4 mm/s, while the spatial coverage remained consistent with the linear case (2.5 cm amplitude). This experiment tested the system's ability to cope with continuous velocity variations and curvature-induced motion.

- **Experiment 3:Visual Servoing Tracking Under Manual Movement of the Phantom**

  In this final experiment, the phantom was moved manually along the X axis to emulate the kind of motion one might observe during a real clinical procedure. Although manually executed, the movement followed a defined pattern: starting from the central position, the phantom was moved 10 cm upward, then 20 cm downward, followed by another 20 cm upward, and finally 10 cm downward to return to the initial position. The purpose of this test was to evaluate how the tracking system performs under less stable and less repeatable conditions, introducing variability such as small tremors, speed fluctuations, and inconsistent trajectories that naturally occur with human operation. This experiment provided valuable insights into the system's resilience and practicality in real-world scenarios, where patient movement or operator imprecision may affect image acquisition.

All experiments were conducted under identical imaging and processing conditions to allow for a fair comparison of the vessel tracking strategies (U-Net, YOLO, and Optical Flow).For each experiment, a total of six test runs were performed per tracking method. These individual tests were subsequently concatenated into a single sequence to allow a more consistent and comprehensive evaluation across the different strategies.

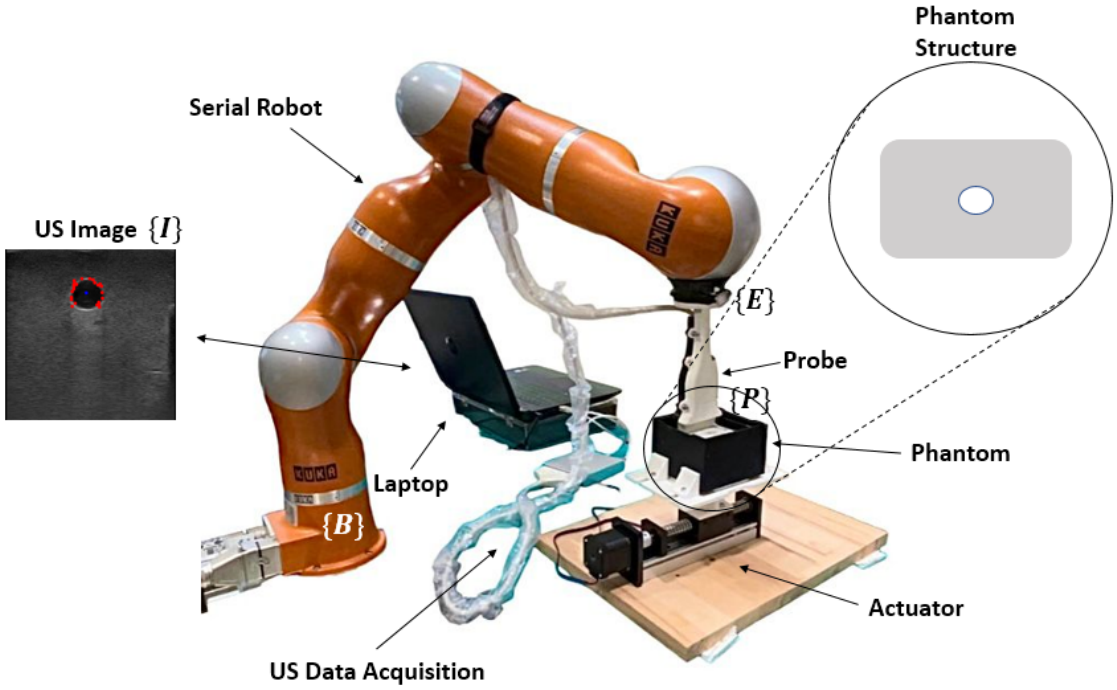


Figure 3.2: System Components of the Visual Servoing Control framework for Robot-Assisted vascular Ultrasound Imaging and Reconstruction

Since the nature of the error signal and system response varies depending on the tracking method employed (e.g., segmentation-based vs. detection-based), different sets of proportional and integral gains were tuned for each method to optimize control stability and responsiveness. These gain values were empirically selected based on preliminary tests to ensure smooth and accurate tracking performance in each case.

The values of the proportional gain Kp and integral gain Ki used in the PI controller for each method are reported in the following table:

Method	K_p	K_i
YOLO	3.5	1.7
Sparse Optical Flow	2.5	1.7
Dense Optical Flow	1.0	1.5
U-NET	1.3	1.7

Table 3.1: Proportional (K_p) and Integral (K_i) gain values used in the PI controller for each method.

3.2. Performance metrics

To evaluate the performance of the vessel tracking and visual servoing system, a set of quantitative metrics was defined based on the real-time tracking error recorded throughout each experiment. The tracking error was defined as the Euclidean distance, in millimeters, between the detected center of the target vessel (proxy point) and the vertical midline of the ultrasound image.

The raw error values were computed in pixels and then converted to millimeters using the known physical width of the ultrasound image. Specifically, the probe's imaging width was $lp = 0.04$ m (i.e., 40 mm), and the image resolution was W=420 pixels, leading to a scale factor:

$$scale = \frac{lp}{W} = \frac{40mm}{420} \approx 0.0952\,\text{mm/pixel} \tag{3.1}$$

This conversion factor was applied to all pixel-based tracking errors, ensuring consistency and clinical relevance.

To assess the distribution of the tracking error data, a Kolmogorov–Smirnov test [25] was performed. The test results indicated that the error distributions did not follow a Gaussian profile. Therefore, the analysis primarily relied on robust statistical metrics, namely:

- **Median (MED):**
  The middle value of the error distribution, offering resistance to outliers and skewed data.

- **Interquartile Range (IQR):**
  The difference between the 75th percentile (Q_3) and 25th percentile (Q_1), providing an estimate of variability:

$$\text{IQR} = Q_3 - Q_1 \tag{3.2}$$

In addition to these robust measures, the Root Mean Squared Error (RMSE) was also computed for all tracking sequences to provide a single scalar value summarizing the overall error magnitude:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(e_i)^2} \tag{3.3}$$

where e_i is the tracking error in millimeters at frame i, and N is the total number of frames.

While MED and IQR were used for comparative evaluation across methods, the RMSE was included to provide an additional indicator of mean performance over time, especially useful for summarizing the overall trend of each algorithm.

These metrics were computed for each of the six test repetitions per experiment and then aggregated by tracking method (U-Net, YOLO, Sparse Optical Flow, Dense Optical Flow) to enable a consistent and meaningful comparison.

3.3. Experiment Results

This section presents the experimental results obtained from the three test scenarios described in the previous section. For each experiment, the system's performance is evaluated using the metrics introduced in Section 3.2.

The results are reported for each of the four tracking methods—U-Net, YOLO, Sparse Optical Flow, and Dense Optical Flow—and include both qualitative observations and quantitative performance metrics (MED, IQR).

3.3.1. Experiment 1: Visual Servoing Tracking Under Linear External Disturbance

In this first experiment, the vascular phantom was moved along a straight-line trajectory at a constant velocity of 2 mm/s, covering a distance of 2.5 cm forward and 2.5 cm backward, repeated three times. The ultrasound probe remained stationary while the phantom moved beneath it along the x axis, simulating a controlled and repeatable motion scenario.

This experiment served as a baseline for evaluating tracking performance under ideal conditions with minimal motion variability.

All four tracking methods successfully detected and followed the vessel center throughout the experiment. However, visual inspection and error signal analysis revealed notable differences:

- **YOLO** emerged as the most effective approach in this setting, offering fast, stable, and consistent detections across all test repetitions. Its real-time performance, coupled with sufficient spatial accuracy, made it the most balanced and robust solution.
- **Sparse Optical Flow (YOLO + Lucas-Kanade)** also performed well, slightly smoothing the error signal thanks to frame-to-frame motion estimation. However, in this low-variability scenario, the added computational overhead was not justified by a significant performance gain.
- **U-Net** produced highly accurate vessel center localizations due to its strong segmentation capabilities, but suffered from substantial computational delays. These delays introduced latency in the control loop, occasionally resulting in overshoots or lagging responses in the servoing behavior.
- **Dense Optical Flow (YOLO + Farnebäck)** showed the least stable perfor-

mance, with more frequent drift and error accumulation, likely due to sensitivity to noise in low-motion conditions.

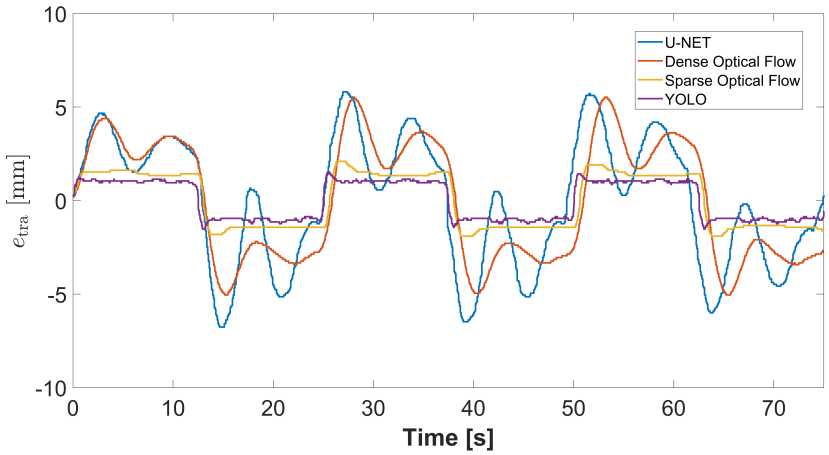


Figure 3.3: Experiment results of visual servoing tracking of vascular anatomies under linear external motion disturbance

The first experiment involved linear motion at 2 mm/s.

$$x(t) = \begin{cases} v \cdot t, & 0 \leq t < T/2 \\ x_{\max} - v \cdot (t - T/2), & T/2 \leq t < T \end{cases}$$

with: $v = 2$ mm/s, $\quad x_{\max} = 25$ mm, $\quad T = 25$ s

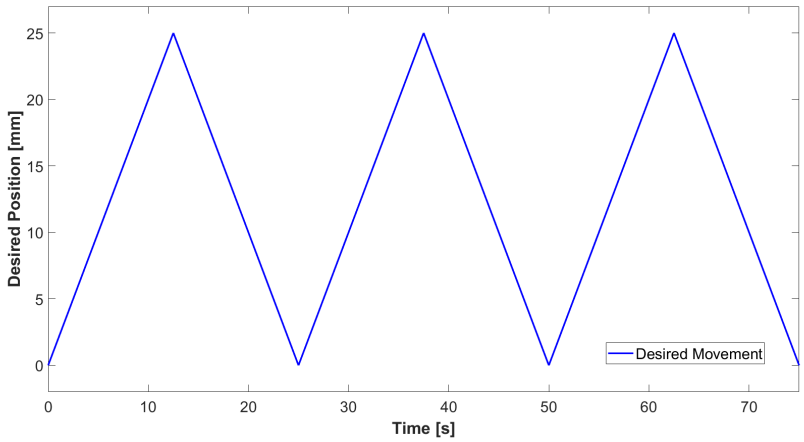


Figure 3.4: Linear Movement

Method	MED [mm]	IQR [mm]
YOLO	1.048	0.095
Sparse Optical Flow	1.429	0.190
Dense Optical Flow	2.857	1.429
U-NET	2.286	2.857

Table 3.2: Median error and IQR under linear external motion disturbance

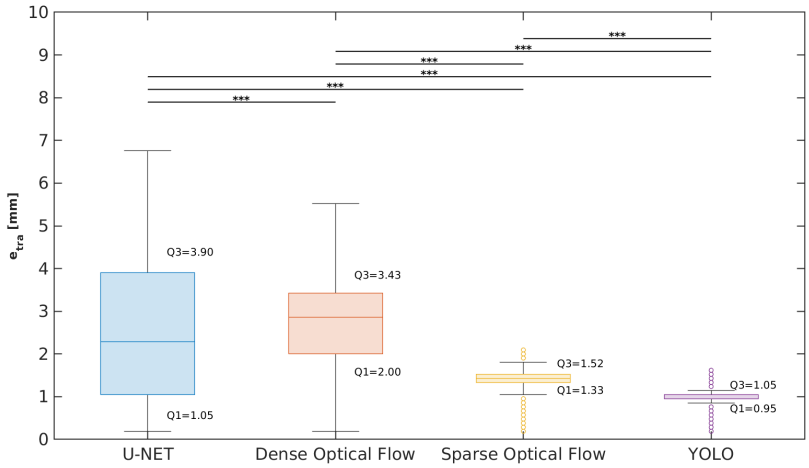


Figure 3.5: Box Plot of the Linear Trajectory Experiment

3.3.2. Experiment 2: Visual Servoing Tracking Under Sinusoidal External Disturbance

In the second experiment, the vascular phantom was displaced following a sinusoidal trajectory along the X axis, covering the same spatial range of 5 cm (2 × 2.5 cm) at an average velocity of 4 mm/s. This motion profile introduces periodic changes in both direction and speed, simulating more dynamic and realistic vessel movement patterns.

- **YOLO** maintained a relatively low and stable error throughout the experiment. Despite its frame-by-frame operation without temporal continuity, it handled the sinusoidal motion well, likely due to its fast inference time and consistent detection accuracy.
- **Sparse Optical Flow** shows slightly larger oscillations and occasional lag spikes,

reflecting a trade-off between temporal smoothing and responsiveness. Its performance is good, but it consistently underperforms compared to YOLO in terms of peak and median error.

- **Dense Optical Flow** produces broad error oscillations, with peaks reaching ±10 mm. While it captures the periodic nature of motion, the larger amplitude suggests less stability and sensitivity to noise or model drift.
- **U-Net** was excluded from this test due to performance limitations observed in Experiment 1. Its computational latency introduced a noticeable delay in the control loop, making it incapable of responding in real time to the increased velocity and curvature of the sinusoidal trajectory. Preliminary attempts confirmed that the system could not maintain alignment, with the probe persistently lagging behind the vessel's position.

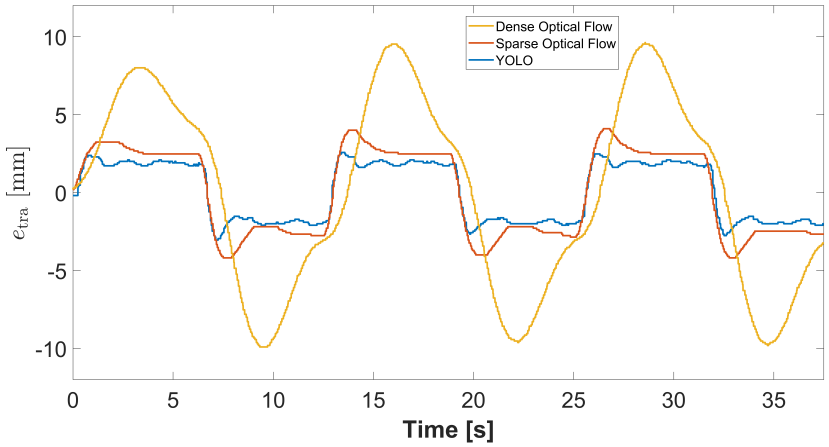


Figure 3.6: Experiment results of visual servoing tracking of vascular anatomies under sinusoidal external motion disturbance

The second method used a sinusoidal trajectory with an amplitude of 12.5 mm, and a period of 12.5 seconds. Starting from position 0, the system moves to the peak position of 2.5 cm in 6.25 seconds and returns to the starting point in the next 6.25 seconds, completing one full cycle.

$$x(t) = A \cdot \left(1 - \cos\left(\frac{2\pi t}{T}\right)\right)$$

with: $A = 12.5$ mm, $\quad T = 12,5$ s

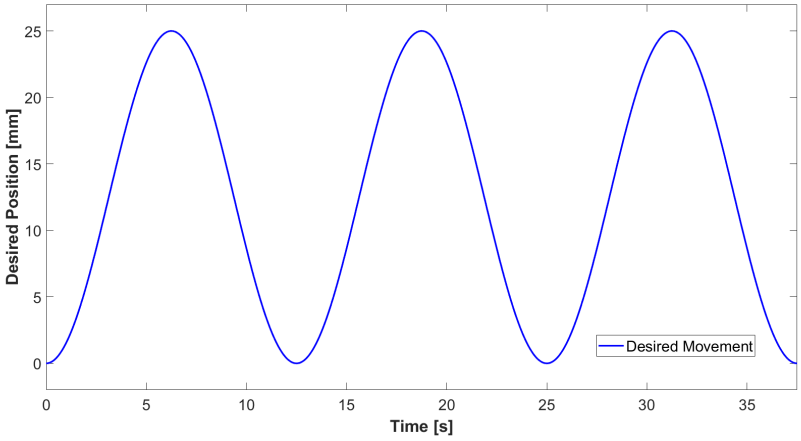


Figure 3.7: Sinusoidal Movement

Method	MED [mm]	IQR [mm]
YOLO	1.905	0.286
Sparse Optical Flow	2.476	0.952
Dense Optical Flow	3.524	6.286
U-NET	N/A	N/A

Table 3.3: Median error and IQR under sinusoidal external motion disturbance

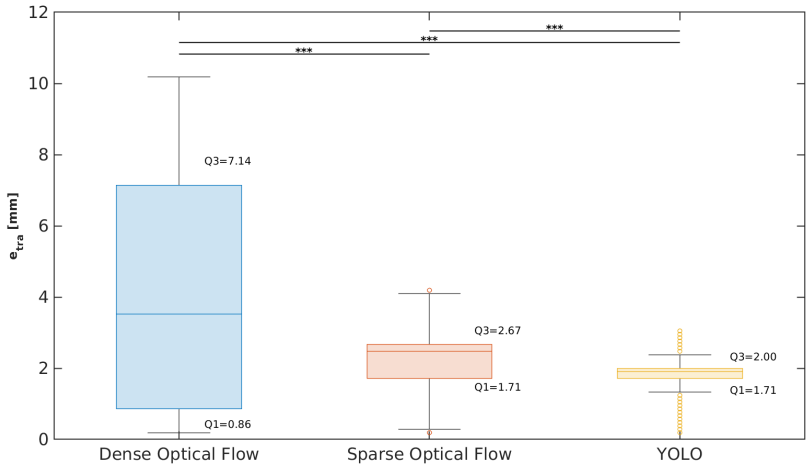


Figure 3.8: Box Plot of the Sinusoidal Trajectory Experiment

3.3.3. Experiment 3: Visual Servoing Tracking Under Manual Movement of the Phantom

In the third experiment, the phantom was moved manually along the X axis to simulate real-world conditions, where patient motion or operator adjustments introduce unpredictability. Although the movement followed a predefined pattern—10 cm upward, 20 cm downward, 20 cm upward, and finally 10 cm downward—the execution was affected by small irregularities, velocity variations, and human-induced tremors.

This experiment was designed to assess how robust and adaptive each tracking method is when faced with inconsistent, non-repetitive motion patterns. Unlike the previous tests, which were highly structured, this scenario introduces variable motion dynamics that more closely resemble clinical situations.

- **YOLO** maintained a relatively stable performance, with moderate fluctuations but consistent centering. While affected by the irregularity of motion, it continued to provide reliable tracking throughout the sequence.
- **Sparse Optical Flow** exhibited higher error spikes and slower adaptation to sharp movement changes. Its temporal smoothing, beneficial in predictable motion, became a limitation under erratic shifts.
- **Dense Optical Flow** presented significant noise and instability, particularly during rapid motion segments. The method suffered from cumulative drift and frequent overshooting.
- **U-Net** showed the most unstable behavior. Although spatially accurate when static, the heavy computational load induced a pronounced delay, causing the system to consistently lag behind the vessel position during rapid transitions. The result was a visibly erratic and delayed error response.

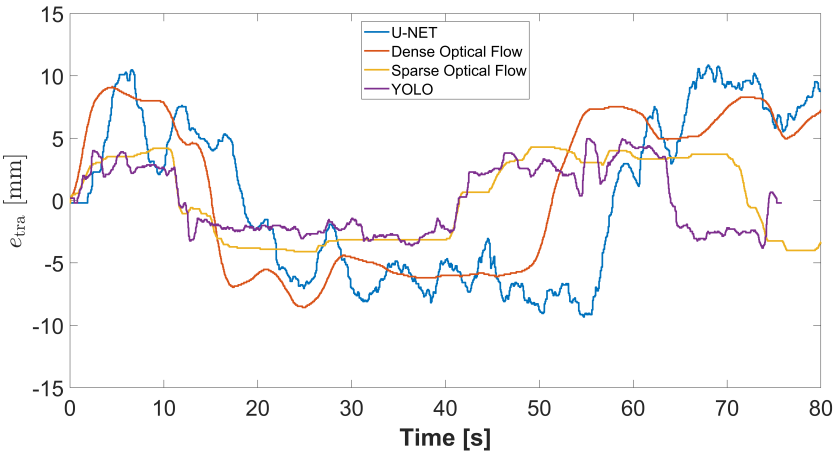


Figure 3.9: Experiment results of visual servoing tracking of vascular anatomies under manual motion

Method	MED [mm]	IQR [mm]
YOLO	2.381	0.857
Sparse Optical Flow	3.238	1.238
Dense Optical Flow	6.095	2.667
U-NET	6.000	3.429

Table 3.4: Median error and IQR under linear manual motion of the Phantom

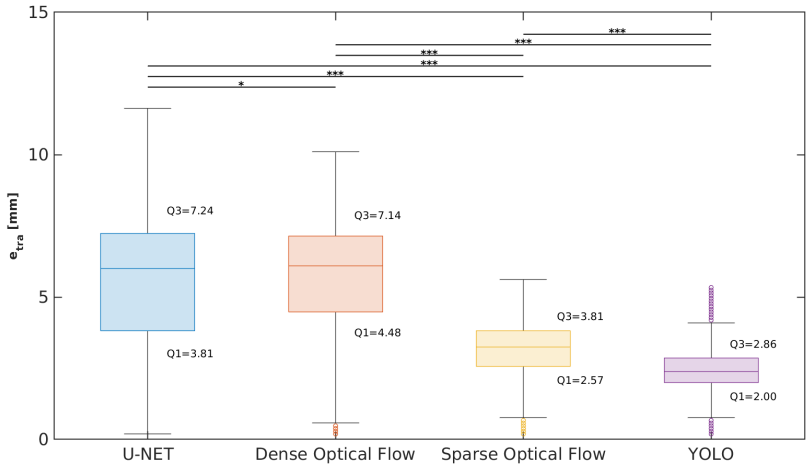


Figure 3.10: Box Plot of the Manual Movement Experiment

3.4. Reconstruction results

As a final part of the system, a module for real-time vessel reconstruction was developed. The goal was not just to track the vessel, but to reconstruct its 3D volume during scanning, using live data from the Telemed ultrasound probe and the robot's pose.

At each frame, the vessel (extracted using YOLO) was combined with the corresponding spatial position of the probe to generate a 3D representation of the vessel. This process was performed in real time, without relying on offline processing of 2D image stacks.

The reconstruction was implemented using the VTK (Visualization Toolkit) library, which allowed for the creation and rendering of simple 3D structures from the acquired data. VTK provided tools for point cloud visualization and basic volume representation.

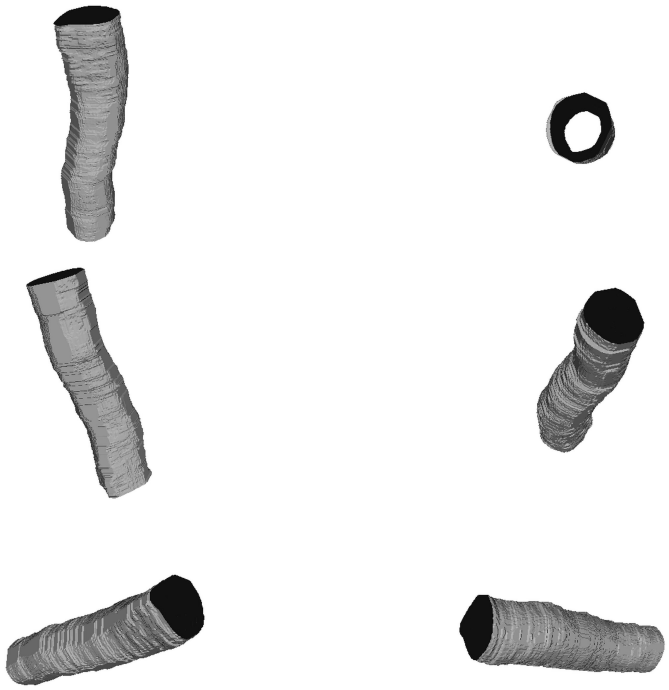


Figure 3.11: 3D Vessel Reconstruction

This reconstruction, while still basic, demonstrates the potential for integrating live ultrasound data with robotic motion to generate meaningful 3D anatomical information. Further refinement could include filtering, surface reconstruction, or integration with prior anatomical models.

4 | Conclusions and future developments

This thesis presented the development and experimental evaluation of a semi-autonomous robotic system for vascular ultrasound imaging, leveraging image-based visual servoing (IBVS) and state-of-the-art image processing techniques. By integrating real-time detection algorithms—based on U-Net, YOLO, and optical flow—with a 7-DOF robotic manipulator, the system demonstrated the ability to maintain stable alignment between the ultrasound probe and a target vessel. The comparative analysis of different tracking strategies highlighted the trade-offs between spatial accuracy, robustness, and computational efficiency, with YOLO showing strong real-time performance and the hybrid optical flow approaches offering enhanced stability.

Overall, the results confirm the feasibility of using visual servoing to automate ultrasound-guided scanning procedures and underscore its potential for improving image consistency and reducing operator dependency.

Looking ahead, the system could be further enhanced by validating its performance on anatomically realistic phantoms and human subjects. Improvements in the control strategy, such as incorporating adaptive or force-based feedback, may increase robustness during complex scanning tasks. Additionally, optimizing the image processing pipeline for real-time clinical use, and integrating the system into hospital workflows, will be key steps toward practical deployment. These future efforts will help bridge the gap between research and real-world application, moving robotic ultrasound closer to autonomous, accessible, and standardized diagnostic imaging.

Bibliography

[1] RHT Bates, Kathryn L Garden, and Terence M Peters. Overview of computerized tomography with emphasis on future developments. *Proceedings of the IEEE*, 71(3):356–372, 1983.

[2] Alexander Selvikvåg Lundervold and Arvid Lundervold. An overview of deep learning in medical imaging focusing on mri. *Zeitschrift für Medizinische Physik*, 29(2):102–127, 2019.

[3] Yu Wang, Xinke Ge, He Ma, Shouliang Qi, Guanjing Zhang, and Yudong Yao. Deep learning in medical ultrasound image analysis: a review. *IEEE Access*, 9:54310–54324, 2021.

[4] Songjie Xiao, Chaojun Wang, Yunze Shi, Jiawei Yu, Linfei Xiong, Chengzhong Peng, and Liangjing Yang. Visual optimization of ultrasound-guided robot-assisted procedures using variable impedance control. In *2021 WRC Symposium on Advanced Robotics and Automation (WRC SARA)*, pages 128–133. IEEE, 2021.

[5] Kevin Evans, Shawn Roll, and Joan Baker. Work-related musculoskeletal disorders (wrmsd) among registered diagnostic medical sonographers and vascular technologists: a representative sample. *Journal of Diagnostic Medical Sonography*, 25(6):287–299, 2009.

[6] Septimiu E. Salcudean, Hamid Moradi, David G. Black, and Nassir Navab. Robot-assisted medical imaging: A review. *Proceedings of the IEEE*, 110(7):951–967, 2022.

[7] Bahaa M Fadel, Bahaa Kazzi, and Dania Mohty. Ultrasound imaging of the superior vena cava: a state-of-the-art review. *Journal of the American Society of Echocardiography*, 36(5):447–463, 2023.

[8] Dianye Huang, Chenguang Yang, Mingchuan Zhou, Angelos Karlas, Nassir Navab, and Zhongliang Jiang. Robot-assisted deep venous thrombosis ultrasound examination using virtual fixture. *IEEE Transactions on Automation Science and Engineering*, 2024.

[9] Felix von Haxthausen, Sven Böttger, Daniel Wulff, Jannis Hagenah, Verónica García-Vázquez, and Svenja Ipsen. Medical robotics for ultrasound imaging: current systems and future trends. *Current robotics reports*, 2:55–71, 2021.

[10] S. Hutchinson, G.D. Hager, and P.I. Corke. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12(5):651–670, 1996.

[11] Svemir Popić and Branko Miloradović. Light weight robot arms-an overview. *INFOTEH-JAHORINA*, 14, 2015.

[12] Claudio Gaz, Fabrizio Flacco, and Alessandro De Luca. Identifying the dynamic model used by the kuka lwr: A reverse engineering approach. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 1386–1392. IEEE, 2014.

[13] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, Andrew Y Ng, et al. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan, 2009.

[14] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[15] Deqing Sun, Stefan Roth, John P Lewis, and Michael J Black. Learning optical flow. In *Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part III 10*, pages 83–97. Springer, 2008.

[16] Aurélien Plyer, Guy Le Besnerais, and Frédéric Champagnat. Massively parallel lucas kanade optical flow for real-time video processing applications. *Journal of Real-Time Image Processing*, 11:713–730, 2016.

[17] Muhammad Hussain. Yolov1 to v8: Unveiling each variant–a comprehensive review of yolo. *IEEE Access*, 12:42816–42833, 2024.

[18] Joachim Weickert, Andrés Bruhn, Thomas Brox, and Nils Papenberg. *A survey on variational optic flow methods for small displacements*. Springer, 2006.

[19] Simon Baker and Iain Matthews. Lucas-kanade 20 years on: A unifying framework. *International journal of computer vision*, 56:221–255, 2004.

[20] W Tong. Formulation of lucas–kanade digital image correlation algorithms for non-contact deformation measurements: a review. *Strain*, 49(4):313–334, 2013.

[21] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Image Analysis: 13th Scandinavian Conference, SCIA 2003 Halmstad, Sweden, June 29–July 2, 2003 Proceedings 13*, pages 363–370. Springer, 2003.

[22] J Redmon. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[23] Zhongliang Jiang, Nehil Danis, Yuan Bi, Mingchuan Zhou, Markus Kroenke, Thomas Wendler, and Nassir Navab. Precise repositioning of robotic ultrasound: Improving registration-based motion compensation using ultrasound confidence optimization. *IEEE Transactions on Instrumentation and Measurement*, 71:1–11, 2022.

[24] Frida Lindberg, Mattias Mårtensson, Christer Grönlund, and Lars-Åke Brodin. Evaluation of ultrasound tissue velocity imaging: a phantom study of velocity estimation in skeletal muscle low-level contractions. *BMC medical imaging*, 13:1–7, 2013.

[25] Zvi Drezner, Ofir Turel, and Dawit Zerom. A modified kolmogorov–smirnov test for normality. *Communications in Statistics—Simulation and Computation®*, 39(4):693–704, 2010.

List of Figures

1.1 Deep Venous Thrombosis Ultrasound Examination[8] . . . 3
1.2 Melody AdEchoTech . . . 4
1.3 Position-Based Visual Servoing . . . 5
1.4 Eye-in-Hand vs Eye-to-Hand . . . 6
1.5 Region of Interest and displacement Δx . . . 7

2.1 KUKA LWR IV+ . . . 10
2.2 KUKA LWR-IV+[12] . . . 11
2.3 KUKA Control Panel (KCP) . . . 12
2.4 KUKA Robot Controller (KRC) . . . 13
2.5 Signal flow between KUKA LWR IV+ and the KRC controller . . . 14
2.6 Ultrasound Probe: Telemed MicrUs EXT-1H . . . 15
2.7 Magewell USB Capture Gen 2 . . . 16
2.8 U-net architecture . . . 18
2.9 Vessel Segmentation Results using U-Net . . . 19
2.10 Vessel Tracking with Sparse Optical Flow . . . 21
2.11 Farneback Pyramid . . . 24
2.12 YOLO structure . . . 26
2.13 Visual-servoed control architecture for robot-assisted vascular ultrasound imaging . . . 31
2.14 Control Scheme of a PI controller . . . 33

3.1 Actuator for Vascular Phantom Motion Simulation . . . 35
3.2 System Components of the Visual Servoing Control framework for Robot-Assisted vascular Ultrasound Imaging and Reconstruction . . . 37
3.3 Experiment results of visual servoing tracking of vascular anatomies under linear external motion disturbance . . . 41
3.4 Linear Movement . . . 41
3.5 Box Plot of the Linear Trajectory Experiment . . . 42

3.6 Experiment results of visual servoing tracking of vascular anatomies under sinusoidal external motion disturbance 43
3.7 Sinusoidal Movement 44
3.8 Box Plot of the Sinusoidal Trajectory Experiment 44
3.9 Experiment results of visual servoing tracking of vascular anatomies under manual motion 46
3.10 Box Plot of the Manual Movement Experiment 46
3.11 3D Vessel Reconstruction 47

List of Tables

2.1 KUKA LWR IV+ joints position and torque limits 11

3.1 Proportional (K_p) and Integral (K_i) gain values used in the PI controller for each method. 38
3.2 Median error and IQR under linear external motion disturbance 42
3.3 Median error and IQR under sinusoidal external motion disturbance 44
3.4 Median error and IQR under linear manual motion of the Phantom 46

Acknowledgements

Prima di tutto vorrei ringraziare i miei genitori per avermi sempre sostenuto nonstante qualche annetto di troppo per completare il mio percorso di studio.Ringrazio infine i miei amici, tutti i compagni di laboratorio e la professoressa per avermi dato la possibilità di lavorare a questa tesi.