

POLITECNICO DI TORINO

MASTER IN MECHATRONIC ENGINEERING



MASTER THESIS

**VISUAL SERVOING BASED ON OPTICAL
COHERENCE TOMOGRAPHY C-SCANS**

Supervisors

Prof. Rizzo Alessandro

Prof. Andreff Nicolas

Dr. Dahroug Bassem

Candidate

Iraci Giuseppe

Table of Contents

List of Figures	III
1 Introduction	2
1.1 Biopsy	2
1.2 Optical Coherence Tomography	8
2 State of Art	13
2.1 Basics of Visual Servoing	13
2.2 Direct Image Based Visual Servoing	14
2.2.1 Photometry Visual Servoing	14
2.2.2 Wavelet Based Visual Servoing	16
2.2.3 Visual servoing using 2D wavelets transformation	17
2.2.4 3D Ultrasound Visual Servoing	18
2.2.5 Conclusion	19
3 Visual Servoing from 3D OCT Data	20
3.1 Introduction	20
3.2 Visual Servoing Using 3D Wavelet Transform	21
3.3 Visual Servoing using 3D wavelets transformation	22
3.4 Conclusion	23
4 Experimentation	24
4.1 Parallel Robot	24
4.2 Instrumentation	24
4.2.1 Control Strategy	26
4.2.2 Result	31
4.3 Conclusion	35
5 Conclusion	36
Bibliography	37

List of Figures

1.1	Bone Marrow Biopsy [1]	2
1.2	Endoscopy biopsy [1]	3
1.3	Needle biopsy [1]	3
1.4	Skin Biopsy [1]	4
1.5	Surgical biopsy [1]	4
1.6	Autofluorescence image of a treated tissue [6]	5
1.7	Narrow band imaging [7]	6
1.8	MRI image [9]	6
1.9	Swept source Optical Coherence tomography [11]	9
1.10	Spectral Domain Optical Coherence tomography [11]	9
1.11	OCT instrumentation	10
1.12	SD-OCT scheme [11]	10
1.13	A-Scan [11]	11
1.14	B-Scan [11]	11
1.15	C-SCAN	12
1.16	(a) Da Vinci, (b) Praxiteles, (c) Cyberknife [14][15]	12
2.1	Visual Servoing with eye-in-hand configuration	13
2.2	Diagram of Wavelet Equation apply to 2D image	17
3.1	Diagram of Wavelet Equation apply to 3D image [23]	21
4.1	SpaceFAB SF-300 BS	24
4.2	Instrumentation setup	25
4.3	Control Strategy Flow Chart	25
4.4	Daubechies of quadratic order, function and corresponding filter [10]	30
4.5	OCT volume, in the right data denoise in the left all the data acquired	31
4.6	Volume acquired on Matlab	31
4.7	Plot the translation perform by the robot	32
4.8	Plot of the rotation perform by the robot	33
4.9	Plot of the velocity sent to the robot	34
4.10	Difference of reference frame between the acquired volume (left) and the same volume on Matlab	35

Chapter 1

Introduction

1.1 What is a Biopsy

The diagnosis of the cancer is principal realized with a medical methodology call biopsy. This technique is able to distinguish cancerous cells from healthy cells. The purpose of biopsy is to obtain enough tissue sample of interest, region that will be analyzed in the laboratory in order to accurately choice the treatment and the prognostic decision.

There is a lot of different typology of how take samples from the human body, here's a list of the different types of biopsy procedures:

1. **Bone marrow biopsy:** It is recommended if an abnormality is detected in the blood or the tumor cells are originated or travel from the bone marrow. To perform this type of biopsy a long needle is used in order to take a bone marrow sample from the back of the hip bone(1.1).

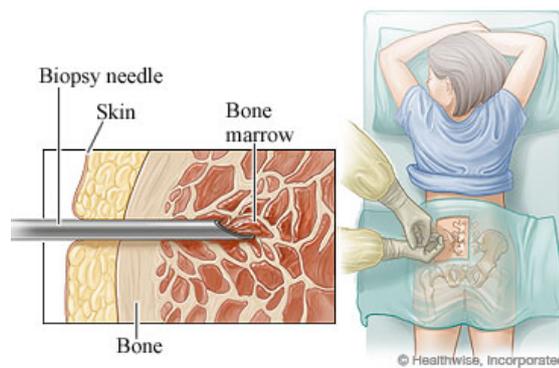


Figure 1.1: Bone Marrow Biopsy [1]

2. **Endoscopy biopsy:** this type of biopsy can be used to collect tissue from inside the bladder, from the lung and from the colon. In this case a flexible tube with a light on the end to see structures inside your body (1.2). Special tools are passed through the tube to take a small sample of tissue to be analyzed.

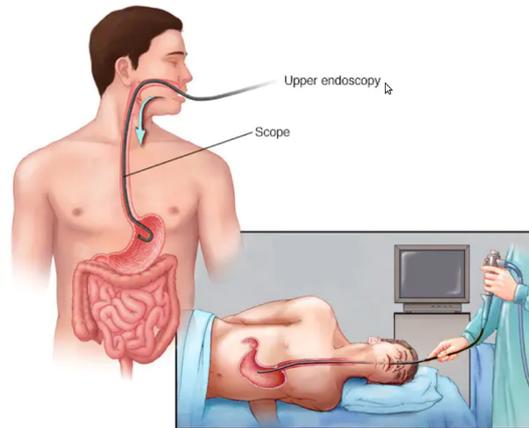


Figure 1.2: Endoscopy biopsy [1]

3. **Needle biopsy:** this method is related to taking samples from the part of the body that can be felt through the skin. The use of imaging guide (as X-ray) allows to take samples from a suspicious area that is placed close to the skin.(1.3). In this case a special needle is used to extract cells. Needle biopsy procedures include:

- Fine-needle aspiration
- Core needle biopsy
- Vacuum-assisted biopsy
- Image-guided biopsy

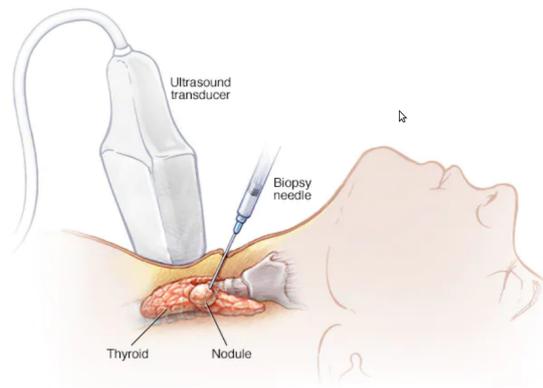


Figure 1.3: Needle biopsy [1]

4. **Skin biopsy:**remove cell from the surface of the body,this is used for diagnosis of melanoma and other skin cancers.1.4

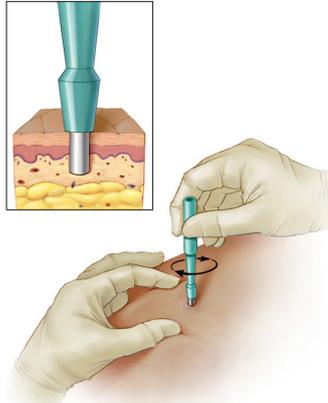


Figure 1.4: Skin Biopsy [1]

5. **Surgical biopsy:** is mostly used where the place to reach is difficult to reach.In this case an incision is made in the skin in other to access the suspicious area in order to take samples.[1](1.5)



Figure 1.5: Surgical biopsy [1]

Physical biopsy gives the ability to reach various parts of the body, but that also involves numerous risks. The most frequent risks are: the possibility of infection, the dispersion of cancer cells during treatment and the limited number of samples that can be taken. [2]

To solve this problem, research advance is made in the field of biomedical technology in order to develop new diagnostic tools to perform conventional biopsies through non invasive sampling of the body tissue. [3]

In the last years a new technology for the analysis of tissue has grown. This new type of methodology takes the name of Optical biopsy. It is a non invasive procedure that offers the possibility to obtain high resolution images of the interest area. This procedure allows to have:

- real time imaging of the area of interest
- more sample of the same region without add cost
- non invasive procedure

The Optic Biopsy is a non invasive method for tissue analysis. Instead of taking samples from the suspect issue this technique uses the optical response of the suspected tissue in order to make a diagnosis. The images take are analyzed in order to understand if there are the presence of tumor cell.

The optical biopsy can be divided into two groups, the methods that recall to highlight the interested party to the study and the part of the equipment used in order to capture the images. These two methods are used together to obtain the most accurate analyses possible. [4]

- **Optical imaging**

There are a number of approach used for optical imaging. The various methods depend upon fluorescence, bio luminescence, absorption or reflectance as source of contrast.

In this case, the tissue is stimulated by a low wavelength light source. This excitation creates a phenomenon called as "autofluorescence" due to the emission of a longer wavelength light by the tissues. The physical properties of this type of emission depend on the excitation light, chemical composition and tissue activity. Depending on the type of wavelength chosen, tissue cells produce different color depending on their state of health.1.6 [5]

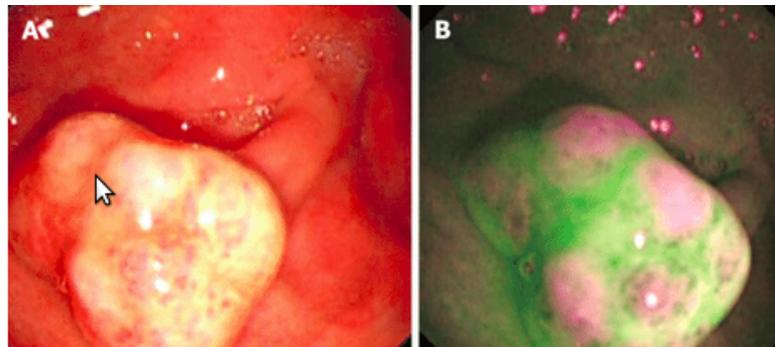


Figure 1.6: Autofluorescence image of a treated tissue [6]

- **Narrow Band imaging:** use three optical interference filters to illuminate the mucosa with light from selected or narrowed wavelength bands of 415 or 540 nm. This technique uses the propriety of diffraction and uptake of white that are absorbed by the hemoglobin present in blood vessels. The wavelengths used to increase the visibility of capillaries and enhancing the borders between structures. [7] [8]

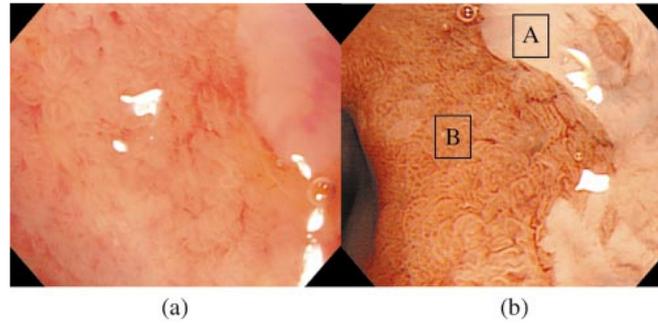


Figure 1.7: Narrow band imaging [7]

- **Magnetic Resonance Imaging:** this method is based on the principles of magnetic resonance. Through the injection of intravenous contrast the low molecular weight agents will pass into the extra-vascular and extracellular space. The imaging follows all the procedure of the injection of the contrast agent till the end. The result obtains (1.8) show how this method is able to tag the cancer cell.[9]

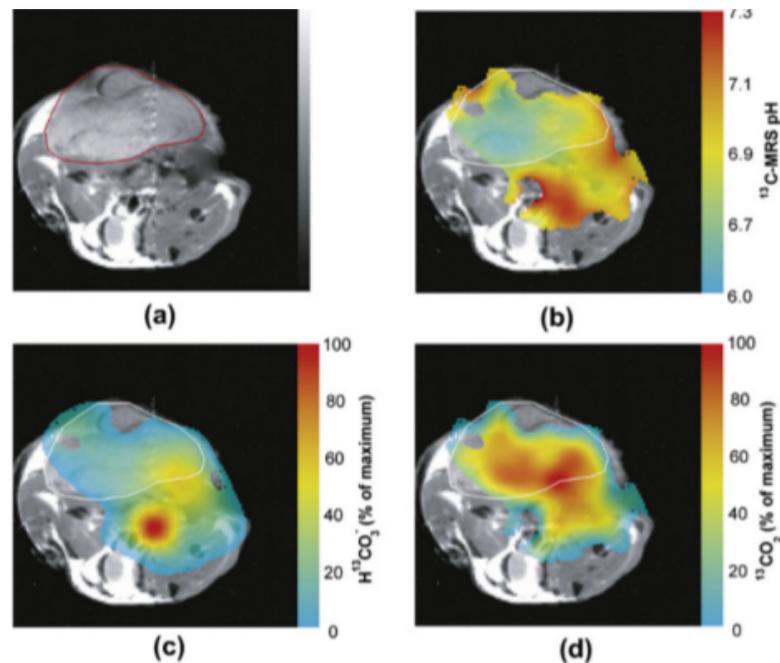


Figure 1.8: MRI image [9]

There are several method that allow to obtain image at high resolution such as: [10]

1. Endoscopy
2. Ecography
3. Confocal microscopy
4. Photoacoustic tomography
5. Optical coherence tomography

The aim of this thesis is to project a controller for a parallel robot that use 3D image acquired by an Optical Coherence tomography as sensor to control the motion of the robot.

1.2 Optical Coherence Tomography

Optical Coherence Tomography is a noninvasive optical imaging modality that provides real time images of the interested sample with micro-level resolution and millimeters of imaging depth. This method utilizes the coherent properties of a light source to measure optical path length delays in a sample. Light is divided into sample and reference arms of an interferometer setup. Backscattered light, attributed to variations in the index of refraction within a sample, is recalled into the sample arm fiber and then combined with the light that has traveled a fixed optical path along the reference arm. A resulting interferogram is measured through the detection arm of the interferometer. The distances and displacements measured is of the same order of magnitude as the wavelengths used. It is able you to create images of a depth between 3/3.5 mm and a resolution of 5 to 10 μmm .

The first technologies develop is the Time Derivative OCT, where in order to obtain a single dimensional measurement (A-Scan) a micrometric translation of the reference mirror according to the desired depth of the sample. In order to reduce the time of acquisition of the sample another technology called Frequency Domain-OCT is developed. In this case there isn't the necessary to move the mirror of the arms, instead a spectrometer is used in order to make the acquisition of the sample thanks to the separation of the component of the spectrum. [11]

The factor that characterize a FD-OCT are:

- resolution: depend on the measure of the optical path delay and on the dimension of the laser spot.
- sensitivity: which depends on the ability of the detector to eliminate noise from the source of the system.
- acquisition rate: that is the time needed in order to acquire a A-scan.

There are two types of FD-OCT systems, each characterized by its light source and detection scheme:

- Swept Source-OCT
- Spectral Domain-OCT

The two systems are both developed using the same fundamental principle, but use different technologies for the reproduction of the same interferogram.

The main difference is that in the SS-OCT (1.9) the source is constituted by the use of a tunable swept light source and a photodetector. This technology allows to use a longer wavelength, due to the fact it is fast, that allow to have an enhanced light penetration and so a better detection of the signal from the deeper layers. This technology allows to use a longer wavelength, due to the fact it is fast, that allow to have an enhanced light penetration and so a better detection of the signal from the deeper layers.

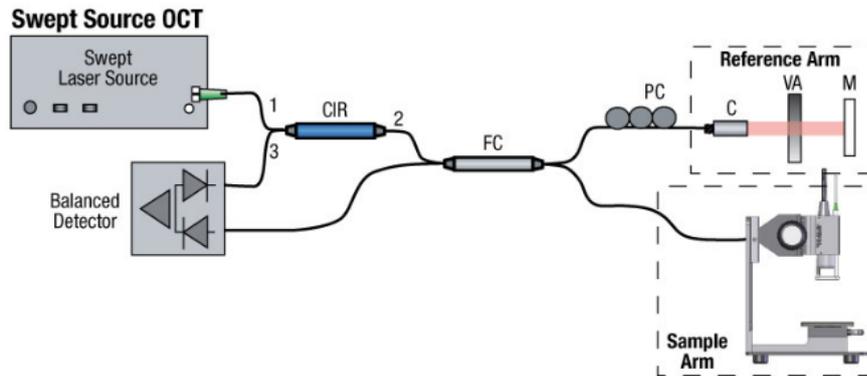


Figure 1.9: Swept source Optical Coherence tomography [11]

The SD-OCT (1.10) use broadband near-infrared superluminescent diode light source and a spectrometer as detector. Gave high mechanical stability and low-phase noise. Availability of a broad range of line cameras has also enabled the development of SD-OCT systems with varying imaging speeds and sensitivities.[11] [12] [13]

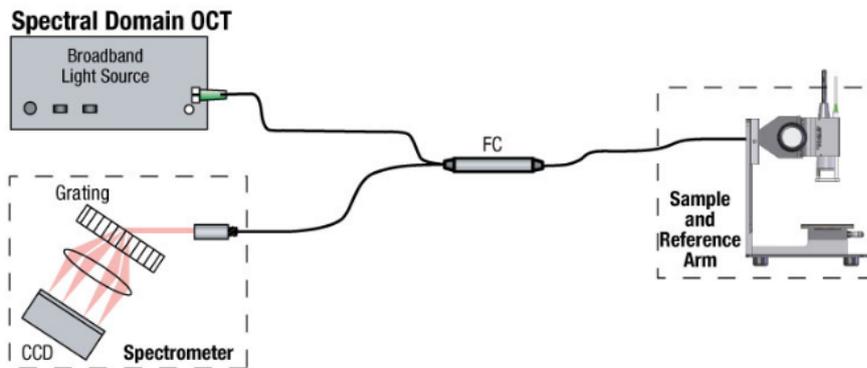


Figure 1.10: Spectral Domain Optical Coherence tomography [11]

In this thesis the system use is the Thorlabs Telesto Spectral-Domain OCT system. This instrument is used to take a non invasive optical biopsy in order to provides high resolution image of human tissue. It is composed of four main parts (1.11): the base unit, an imaging probe and a pc.

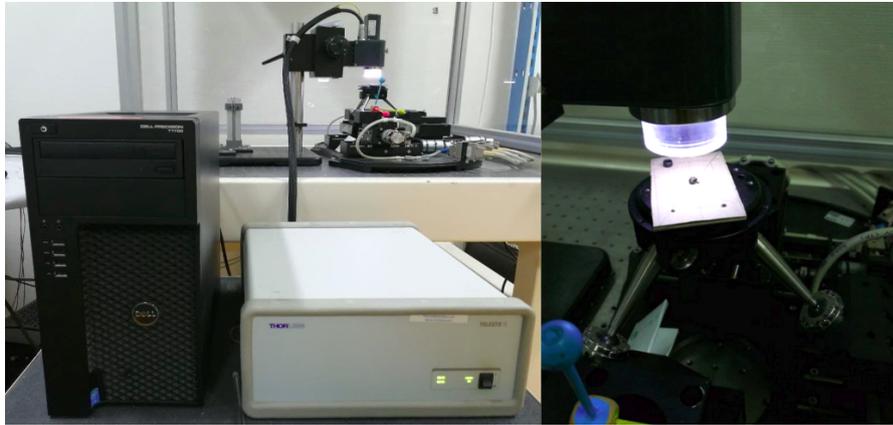


Figure 1.11: OCT instrumentation

The base unit that is the main component of the system, it send light to the application and communicate with the PC through a USB connection, triggers the camera of the spectrometer and delivers the measurement data. The base unit contains two broadband super luminescent diodes (SLD) which are combined by a fiber optic coupler and directed through an optical circulator to the probe port at the backside of the device. The sample beam is routed over two galvanometer actuated mirrors to allow for scanning in two axes. The scan objective, then focuses the beam in sample. Backscattered and back-reflected light is collected by the scan objective again and travels back to the fiber. The light reflected into the reference arm is retro-reflected back into the fiber. (1.12)

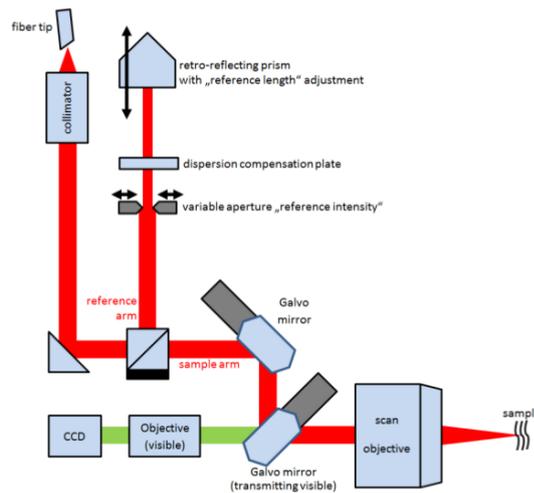


Figure 1.12: SD-OCT scheme [11]

Analyzing the frequency spectrum of the interference between the reflected light and a stationary reference mirror it's possible to determine the optical characteristics of the sample. The information in depth take using this method is called A-Scan. (1.13)

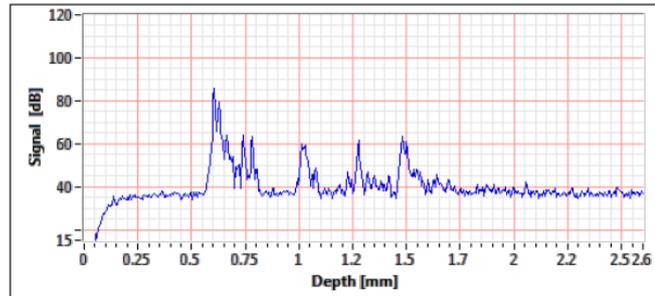


Figure 1.13: A-Scan [11]

The imaging probe is equipped with two galvanometer driven scanner. When scanning one mirror is moving while collecting multiple A-Scans a 2-Dimensional image is created. This is referred to as a B-Scan. (1.14)

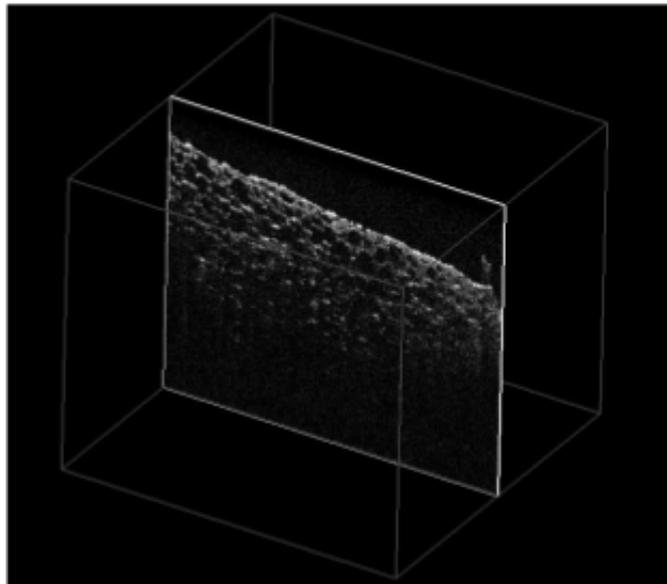


Figure 1.14: B-Scan [11]

and at the end when scan both galvanometer mirror are moving a volume can be acquired. (1.15)

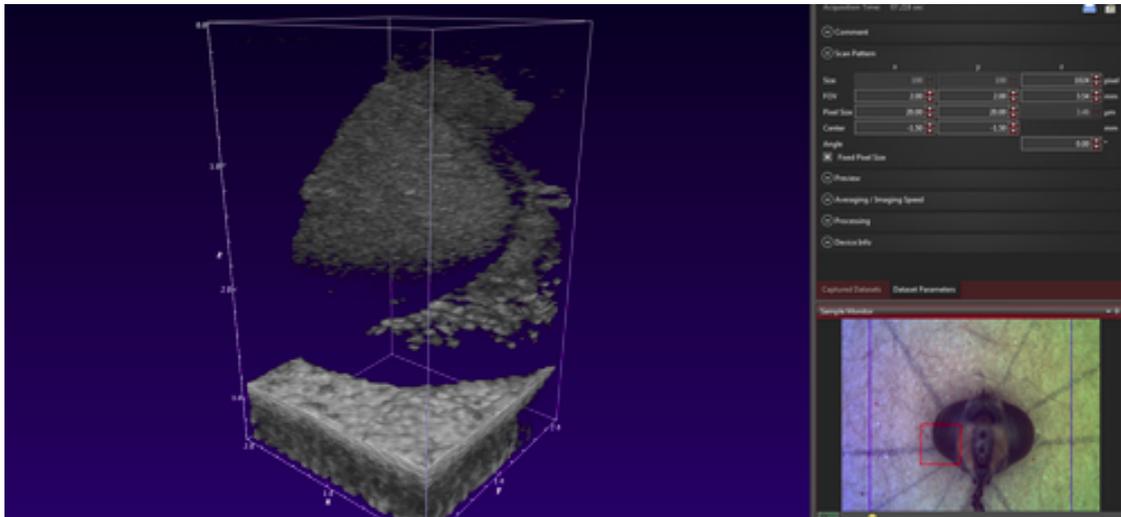


Figure 1.15: C-SCAN

The medical robotics are a constant growth field, the application of the robotics solution in the field of medical field reaches a lot of fields. They are used in order to perform autonomous or semiautonomous surgeries (Da Vinci) in orthopedics is accurate and precise bone resection (Praxiteles) or to guide special tool inside the human body (CyberKnife).(1.16) These types of systems allow operators and medical staff to have at their disposal fast and stable precision instruments that help them in the medical operation.[14]

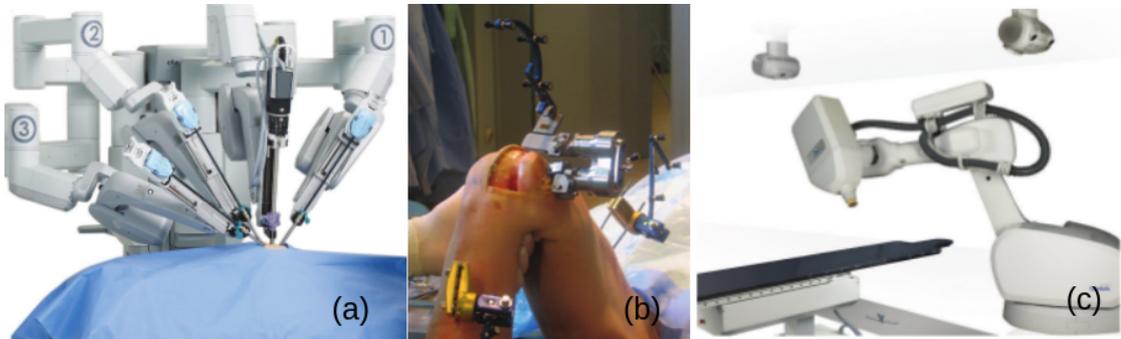


Figure 1.16: (a) Da Vinci, (b) Praxiteles, (c) Cyberknife [14][15]

This project concerns about the development of a visual control law in which the robot sensor is an OCT imaging system. In an OCT optical biopsy the quality of the image is sensitive to the movement of the tissue that is common in-vivo dynamic imaging. The use of a robot to control the position of the probe would make possible to compensate the tissue movement reducing the noise and improve the quality of the image. In addition, the ability to automatically control the movement of the robot gives the possibility to reposition ourselves on a tissue that has already been controlled by the instrument at an early stage. In this way it's possible to understand how the condition of the tissue develops over the time.

Chapter 2

State of Art

2.1 Basics of Visual servoing

The Visual Servoing is a control technique that consists of using computer vision information to control the movement of a robot. [16]. there are different types of visual servoing, they are divided according to the type of information that is extracted from the camera. They are classified into:

1. Image Based (based on the characteristics of the image)
2. Position Base (that extrapolate the position of the object within the image)
3. Hybrid Approach (a hybrid of the two techniques described above)

There are different types of configurations with respect to camera position. It can be placed directly on the end-effector of the robot, eye-in-hand, or can be fixed in the workspace, eye-to-hand. Once the camera position has been set, it is necessary to understand what kind of control strategy should be used. There are different types of control architecture: sequential, cinematic or dynamic. In the case under analysis a cinematic control has been chosen because even if it is less reactive than a dynamic control, it is more stable at low frequencies of image acquisition. With this information the control loop can be set (Figure.2.1).[16, 17]

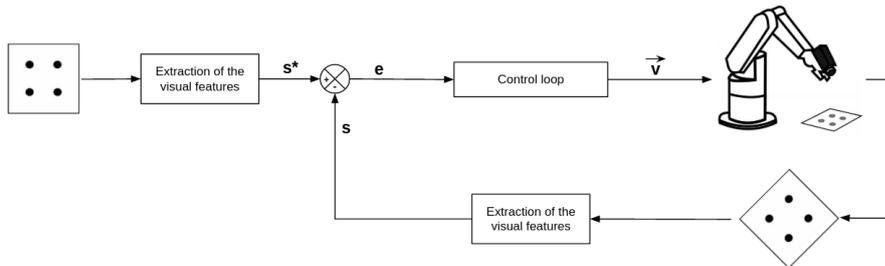


Figure 2.1: Visual Servoing with eye-in-hand configuration

The control loop strategy put in relation the information extract from the scene to the position of the robot. The aim of the control schemes is to minimize the error between the current position of the robot and the desire position. [18]

$$\mathbf{e}(\mathbf{t}) = \mathbf{s}(\mathbf{m}(\mathbf{t}), \mathbf{a}) - \mathbf{s}^* \quad (2.1)$$

The vector of the visual features \mathbf{s} depends on \mathbf{m} a vector that contain a set of information taken from image (coordinates of interest point etc) and from \mathbf{a} that contain other relevant information about the system (3-D models of the object or intrinsic parameter of the camera).[16] The definition of the vector of visual features can be a lot general and it depend on the type of visual servoing approach use. The vector in fact can contain information directly available from the image, (Image based Visual Servoing)or it consists of three-dimensional parameters that are estimated from the image (Position Based Visual Servoing). In the IBVS the command is based directly on the image plane, the information extract from the image are geometric shapes, geometrical moment of edge or can be global information (as in the photometry approach the luminous intensity). This type of approach is less effect to the error given by the camera calibration and from error of the model estimation.

To implement the control scheme, we choose the main direct strategy design a command in velocity. This type of controller put in relation the velocity of the camera to the variation of the 'vector of visual parameter' (\mathbf{s})[16].

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v}_c \quad (2.2)$$

\mathbf{L}_s is the interaction matrix related to \mathbf{s} . Considering the spatial velocity of the camera composed by the instantaneous linear velocity and the angular velocity $\vec{v}_c = (\mathbf{v}_c, \omega_c)$ the dimension of the interaction matrix will be $\mathbf{L}_s \in \mathbb{R}^{k \times 6}$ where k is the number of features present inside the vector of the primitive visual.

Since the control strategy is design to minimize the error between the current position of the robot and the desire position, and suppose that the \mathbf{s}^* (the vector that contain the set of parameter of the desire position) is constant in time we can write the (2.2) as:

$$\dot{\mathbf{e}} = \mathbf{L}_e \mathbf{v}_c \quad (2.3)$$

If we choose an exponential decoupled decrease of the error:

$$\dot{\mathbf{e}} = -\lambda \mathbf{e} \quad (2.4)$$

So the velocity of the camera can be write as:

$$\mathbf{v}_c = -\lambda \mathbf{L}_e^+ \mathbf{e} \quad (2.5)$$

Where the \mathbf{L}_e^+ is chose as the More-Penrose Pseudo inverse that is calculated as $\mathbf{L}_e^+ = (\mathbf{L}_e^T \mathbf{L}_e)^{-1} \mathbf{L}_e^T$

2.2 Direct Image Based Visual Servoing

In our case of study we use a Direct Image Based Visual Servoing approach, this means that it takes the global information from the image and this parameter are used in order to calculate the interaction matrix. T

2.2.1 Photometry Visual Servoing

There are many approach that used different type of image information in order to define the interaction matrix, among these what will be study in this thesis is the photometry approach that that use the luminous intensity of the pixel to define the set of the vector of visual parameter (\mathbf{s}).[19]

$$\mathbf{s}_{photo_i}(t) = [\mathbf{I}(\mathbf{p}_1(t)) \dots \mathbf{I}(\mathbf{p}_{M \times N}(t))] = \mathbf{vec}(I) \quad (2.6)$$

where the $\mathbf{vec}(I)$ is a the operator that reshape the image of dimension $\mathbf{M} \times \mathbf{N}$ in a vector of dimension $\mathbf{MN} \times \mathbf{1}$.

$$s_{photo_i}(t) = \mathbf{I}(\mathbf{p}_i)(\mathbf{t}) \quad (2.7)$$

The vector s_{photo_i} is the image intensity in $\mathbf{p}_i(\mathbf{t})$. Where the parameter \mathbf{p}_i are the coordinates of a pixel in reference frame of the camera at certain time.

$$\mathbf{p}_i(t) = (\mathbf{x}_i, \mathbf{y}_i)^\top \quad (2.8)$$

and it is the perspective projection of the point:

$$\mathbf{p}_i(t) = (\mathbf{X}_i, \mathbf{Y}_i, \mathbf{Z}_i)^\top \quad (2.9)$$

to implement the control law the vector of visual parameter must be derived in time. let us compute e

$$\frac{ds_{photo_i}(t)}{dt} = \mathbf{L}_{s_{photo_i}} \mathbf{v}_c(t) \quad (2.10)$$

let us compute each of its components:

$$\forall \mathbf{i} \frac{ds_{photo_i}(t)}{dt} = \mathbf{L}_{s_{photo_i}} \mathbf{v}_c(t) \quad (2.11)$$

the derivative of the visual parameter can be write as:

$$\frac{ds_{photo_i}(t)}{dt} = \frac{d\mathbf{I}(\mathbf{p}_i(\mathbf{x}, \mathbf{y}, \mathbf{t}))}{dt} = \frac{\partial \mathbf{I}(\mathbf{p}_i(\mathbf{x}, \mathbf{y}))}{\partial \mathbf{x}} \mathbf{v}_x + \frac{\partial \mathbf{I}(\mathbf{p}_i(\mathbf{x}, \mathbf{y}))}{\partial \mathbf{y}} \mathbf{v}_y + \frac{\partial \mathbf{I}(\mathbf{p}_i(\mathbf{x}, \mathbf{y}))}{\partial \mathbf{t}} \quad (2.12)$$

considering the hypothesis that in a short interval of time the intensity of the pixel do not change (optical flow assumption) the derivative of the intensity of the light can be put equal to zero.[20] So the 2.12 became:

$$\frac{ds_{photo_i}(t)}{dt} = \frac{\partial \mathbf{I}(\mathbf{p}_i(\mathbf{x}, \mathbf{y}))}{\partial \mathbf{x}} \mathbf{v}_x + \frac{\partial \mathbf{I}(\mathbf{p}_i(\mathbf{x}, \mathbf{y}))}{\partial \mathbf{y}} \mathbf{v}_y \quad (2.13)$$

$$\frac{ds_{photo_i}(t)}{dt} = [\nabla \mathbf{I}_x(\mathbf{p}_i(\mathbf{t})) \quad \nabla \mathbf{I}_y(\mathbf{p}_i(\mathbf{t}))] \dot{\mathbf{p}}_i(\mathbf{t})^\top \quad (2.14)$$

From (2.13) the velocity of the point in the 2D image plan can be relate to the velocity of the camera. The coordinate of the point in the image plan are calculated using the geometry relation between the 3D point in the Cartesian space $\mathbf{x} = (\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ in the image 2D plane with normalized coordinates $\mathbf{x} = (x, y)$:

$$\begin{cases} x = \frac{X}{Z} \\ y = \frac{Y}{Z} \end{cases} \quad (2.15)$$

in this way is possible to link the relation between the velocity of the point in the bidimensional image and the velocity of the camera:

$$\dot{\mathbf{p}}_i = L_{2D} * v_c \quad (2.16)$$

Where \mathbf{L}_{2D} is the interaction matrix calculated using geometrical constrain [16]

$$\mathbf{L}_{2D} = \begin{bmatrix} \frac{-1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & \frac{-1}{Z} & \frac{y}{Z} & (1+y^2) & -xy & -x \end{bmatrix} \quad (2.17)$$

the (2.14) using the (2.16) can be write a:

$$\frac{ds_{photo_i}(t)}{dt} = [\nabla \mathbf{I}(\mathbf{p}_i(\mathbf{t}))] \mathbf{L}_{2D_i} \mathbf{v}_c^\top \quad (2.18)$$

knowing the 2.2 the vector of the visual feature can be written as:

$$\dot{\mathbf{s}}_i(\mathbf{t}) = [\nabla \mathbf{I}(\mathbf{p}_i(\mathbf{t}))] \mathbf{L}_{2D_i} \mathbf{v}_c^\top \quad (2.19)$$

Final the interaction matrix can be isolate:[18][19]

$$\mathbf{L}_s = - \begin{bmatrix} (\nabla \mathbf{I}(\mathbf{p}_i(\mathbf{t})))^\top \mathbf{L}_{2D_i} \\ \vdots \\ (\nabla \mathbf{I}(\mathbf{p}_{W \times H}(\mathbf{t})))^\top \mathbf{L}_{2D_i} \end{bmatrix} \quad (2.20)$$

2.2.2 Wavelet Based Visual Servoing

This type of solution where the interaction matrix contain the gradient of the luminous intensity is computation heavy and time consuming. To overcome this problem [17] propose to use a method based on the multi resolution wavelet.

The multi-resolution transformation use two different type of function, a scale function and a the wavelet function 2.21. The wavelet transformation of a bi-dimensional signal (in this case an image), gives a four sub image that have a specific characteristics.

$$\psi_{(\mathbf{j}, \mathbf{m})} = 2^{(-\frac{\mathbf{j}}{2})} \psi(2^{(-\mathbf{j})} \mathbf{t} - \mathbf{m}) \quad (2.21)$$

$$\phi_{(\mathbf{j}, \mathbf{m})} = 2^{(-\frac{\mathbf{j}}{2})} \phi(2^{(-\mathbf{j})} \mathbf{t} - \mathbf{m})$$

As the entering data are a 2D image the combination of the two function applied in all the two dimension give four different new function that must be applied in order to obtain the decomposition of the original image.

$$\begin{cases} \psi^0(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})\phi(\mathbf{y}) \\ \psi^1(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})\psi(\mathbf{y}) \\ \psi^2(\mathbf{x}, \mathbf{y}) = \psi(\mathbf{x})\phi(\mathbf{y}) \\ \psi^3(\mathbf{x}, \mathbf{y}) = \psi(\mathbf{x})\psi(\mathbf{y}) \end{cases} \quad (2.22)$$

The application of each function in the equation give us a different type of subsection of the original image as is shown in the figure above.

This procedure allow to encode the same information as the original image in half the space. The control law in this case is developed using the method propose by [21] in which the new equation of the optical flow (called multi resolution optical flow) is calculate as a scalar product between the family of wavelet choose and the equation of the optical flow. In this case the vector of the visual parameter contain the coefficient of the wavelet instead of the information of the original image. .

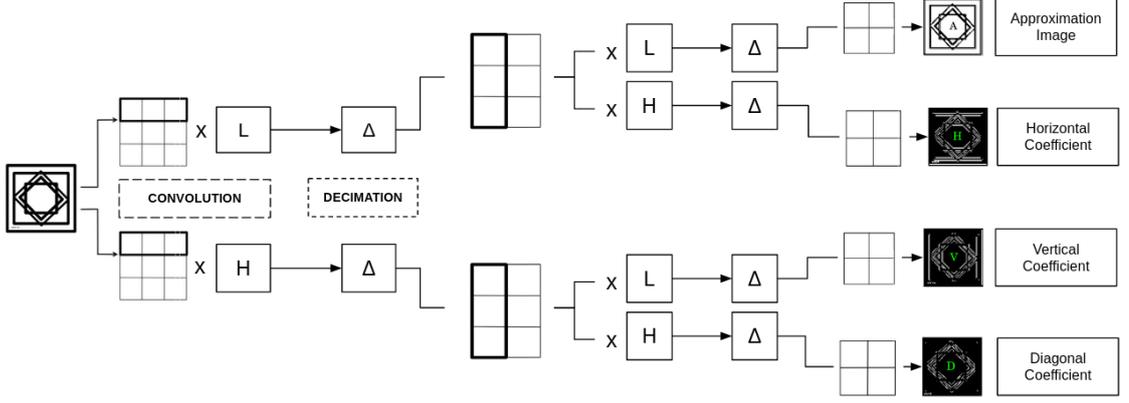


Figure 2.2: Diagram of Wavelet Equation apply to 2D image

2.2.3 Visual servoing using 2D wavelets transformation

in the development of [21] the equation of the optical flow can be write as a functional minimization problem. Using a set of the multi-resolution wavelet the inner product between the equation (2.13) and the wavelet function are computed [21]:

$$\frac{ds_i(t)}{dt} = \frac{d}{dt} \langle \mathbf{I}(\mathbf{p}_i), \psi \rangle = \left\langle \frac{d\mathbf{I}}{dt}(\mathbf{p}_i), \psi \right\rangle = \int \int \left(\frac{\partial \mathbf{I}}{\partial \mathbf{x}} \mathbf{v}_x(\mathbf{p}_i) + \frac{\partial \mathbf{I}}{\partial \mathbf{y}} \mathbf{v}_y(\mathbf{p}_i) + \frac{\partial \mathbf{I}}{\partial \mathbf{t}} \right) \psi_{j,m}^{(1)}(\mathbf{p}_i) dx dy = 0 \quad (2.23)$$

The notation is simplify (2.24) the term \mathbf{I} is referring at the luminous intensity of each pixel $\mathbf{p}_i(\mathbf{x}, \mathbf{y})$.

$$I = I(p_i) \quad (2.24)$$

knowing that:

$$\langle f, g \rangle = \int \int \mathbf{f}(\mathbf{x}) \mathbf{g}(\mathbf{x}) dx dy \quad (2.25)$$

the (??) can be write as:

$$\left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{x}} \mathbf{v}_x(\mathbf{p}_i), \psi_{j,m}^{(1)} \right\rangle + \left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{y}} \mathbf{v}_y(\mathbf{p}_i), \psi_{j,m}^{(1)} \right\rangle + \left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{t}}, \psi_{j,m}^{(1)} \right\rangle = 0 \quad (2.26)$$

[17] made the hypothesis that the velocity of the point in the coordinate of the starting image is the same on the new multi-resolution coordinate.[17]

$$\begin{cases} \mathbf{v}_x(p_i) = \mathbf{v}_x(2^{-j}\mathbf{k}) & \forall \mathbf{x} \in \text{support } \psi_{j,m}^{(1)} \\ \mathbf{v}_y(p_i) = \mathbf{v}_y(2^{-j}\mathbf{k}) & \forall \mathbf{y} \in \text{support } \psi_{j,m}^{(1)} \end{cases} \quad (2.27)$$

so the velocity can be put outside the inner product:

$$\left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{x}}, \psi_{j,m}^{(1)} \right\rangle \mathbf{v}_x(p_i) + \left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{y}}, \psi_{j,m}^{(1)} \right\rangle \mathbf{v}_y(p_i) + \frac{\partial}{\partial \mathbf{t}} \langle \mathbf{I}, \psi_{j,m}^{(1)} \rangle = 0 \quad (2.28)$$

and after a integration by part [22] the equation can be write as:

$$\left\langle \mathbf{I}, \frac{\partial \psi_{\mathbf{j}, \mathbf{m}}^{(1)}}{\partial \mathbf{x}} \right\rangle \mathbf{v}_x(p_i) + \left\langle \mathbf{I}, \frac{\partial \psi_{\mathbf{j}, \mathbf{m}}^{(1)}}{\partial \mathbf{y}} \right\rangle \mathbf{v}_y(p_i) + \left\langle \mathbf{I}, \frac{\partial \psi_{\mathbf{j}, \mathbf{m}}^{(1)}}{\partial \mathbf{z}} \right\rangle \mathbf{v}_z(p_i) + \frac{\partial}{\partial \mathbf{t}} \langle \mathbf{I}, \psi_{\mathbf{j}, \mathbf{m}}^{(1)} \rangle = 0 \quad (2.29)$$

the development of ?? is :

$$\dot{\mathbf{s}}_{W2D_i} = - \begin{bmatrix} \mathbf{g}_{(2^{(j+1)})}^1(\mathbf{p}_i) & \mathbf{g}_{(2^{(j+1)})}^2(\mathbf{p}_i) & \mathbf{g}_{(2^{(j+1)})}^3(\mathbf{p}_i) \end{bmatrix} \begin{bmatrix} \mathbf{v}_x(\mathbf{p}_i) \\ \mathbf{v}_y(\mathbf{p}_i) \end{bmatrix} \quad (2.30)$$

knowing the relation between the velocity of the pixel and the velocity of the camera 2.16:

$$\begin{bmatrix} \mathbf{v}_x(\mathbf{p}_i) \\ \mathbf{v}_y(\mathbf{p}_i) \end{bmatrix} = 2^j \mathbf{L}_{2D} \mathbf{v}_c \quad (2.31)$$

so the development of the equation became:

$$\dot{\mathbf{s}}_{W2D_i} = -2^j \begin{bmatrix} \mathbf{g}_{(2^{(j+1)})}^1(\mathbf{p}_i) & \mathbf{g}_{(2^{(j+1)})}^2(\mathbf{p}_i) \end{bmatrix} \mathbf{L}_{2D} \mathbf{v}_c \quad (2.32)$$

the equation of the interaction matrix can be write for all the element that compose the vector of visual features:

$$\mathbf{L}_{\mathbf{s}_{W2D}} = \begin{bmatrix} -2^j \begin{bmatrix} \mathbf{g}_{(2^{(j+1)})}^1(\mathbf{p}_i(1,1)) & \mathbf{g}_{(2^{(j+1)})}^2(\mathbf{p}_i(1,1)) \end{bmatrix} \mathbf{L}_{2D} \\ \mathbf{L}_{\mathbf{s}_{W2D}_{2^{(j+1)}}}(1,2) \\ \vdots \\ \mathbf{L}_{\mathbf{s}_{W2D}_{2^{(j+1)}}}(M, N) \end{bmatrix} \quad (2.33)$$

2.2.4 3D Ultrasound Visual Servoing

In the method used by [17], the tests were done by extracting information from bi-dimensiona image in the depth of the image. This type of approach gave good results in realizing a controller 3DOF, where the movement of the robot lie on the plan of the image. In the development of a control of 6DOF the same approach did not give the expected results, so it was assumed by [17] to use orthogonal B-scan of the same volume in order to extract the information on the depth of the image.

Following the development of [20], [19] propose a new intensity-based approach that use 3DUS (three-dimensional ultrasound) volume. In this case the equation are:

$$\mathbf{L}_{3DUS}(P_i) = [\nabla_x \mathbf{I}(P_i) \quad \nabla_y \mathbf{I}(P_i) \quad \nabla_z \mathbf{I}(P_i)] \mathbf{L}_{3D}(P_i) \quad (2.34)$$

Where the \mathbf{p}_i is the i-th voxel in the vector-reshaped 3DUS volume $\mathbf{I}(\mathbf{p}_i)$ is the US intensity in that voxel. The relation between the velocity of the time variation of the intensity of the 3D point and the velocity of the robot are calculate using the Varignon's formula for a velocity composition in a solid. [19]

$$\mathbf{L}_{3D} = \begin{bmatrix} I_3 & [\vec{P}_i]_{\times} \end{bmatrix} \quad (2.35)$$

This interaction matrix is computed with the skew-symmetric matrix of the vector cross product. The interaction matrix doesn't need to be estimated using the coordinate of the camera reference frame, it is estimate only using the information of the coordinate associate to the 3D image.

2.2.5 Conclusion

The aim of this thesis is elaborate a controller that is able to guide a robot in the objective of optical biopsies.

The mathematical approach In this thesis we investigate image-based techniques (using three dimensional image) in order to be able to command the end-effector of a robot into a patient to take measurement for living mucosa.

The theoretical part of the thesis start from the work that [17] do using B-scan, that we extend relying on the development of the equation of the optical flow made by [21] for a three dimension image.

Chapter 3

Visual Servoing from 3D OCT Data

3.1 Introduction

As we have seen previously, the work done by [17] uses two-dimensional images that allow the realization of a robust controller that moves the robot in the plane of the image. To be able to develop a 6DOF control we thought to use all the information related to the volume of the image in order to be able to extract the information regarding the depth. Using the development of the optical flow equation for three dimensional signals developed by [21] we are able to develop the control law for the parallel robot. The last missing step was to be able to calculate an interaction matrix that relates the velocity of the point in the image volume to the velocity of the camera. We chose to use the approach developed by [19] for the construction of a 6DOF controller using 3D ultrasound volumes. This type of solution uses the Varignon's formula to calculate the interaction matrix. In this thesis we try to merge the two approaches [17, 19] in order to be able to develop a 6DOF using C-scans.

3.2 Visual Servoing Using 3D Wavelet Transform

In this section the wavelet function are extend to the decomposition of the volume. As the entering data are a 3D volume the combination of the two function applied in all the three dimension give eight different function that can be applied to the equation.

$$\begin{cases} \psi^0(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \phi(\mathbf{x})\phi(\mathbf{y})\phi(\mathbf{z}) \\ \psi^1(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \phi(\mathbf{x})\phi(\mathbf{x})\psi(\mathbf{z}) \\ \psi^2(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \phi(\mathbf{x})\psi(\mathbf{y})\phi(\mathbf{z}) \\ \psi^3(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \phi(\mathbf{x})\psi(\mathbf{y})\psi(\mathbf{z}) \\ \psi^4(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \psi(\mathbf{x})\phi(\mathbf{y})\phi(\mathbf{z}) \\ \psi^5(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \psi(\mathbf{x})\phi(\mathbf{y})\psi(\mathbf{z}) \\ \psi^6(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \psi(\mathbf{x})\psi(\mathbf{y})\phi(\mathbf{z}) \\ \psi^7(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \psi(\mathbf{x})\psi(\mathbf{y})\psi(\mathbf{z}) \end{cases} \quad (3.1)$$

The application of each function in the equation give us a decomposition of the original volume as is shown in the figure above.

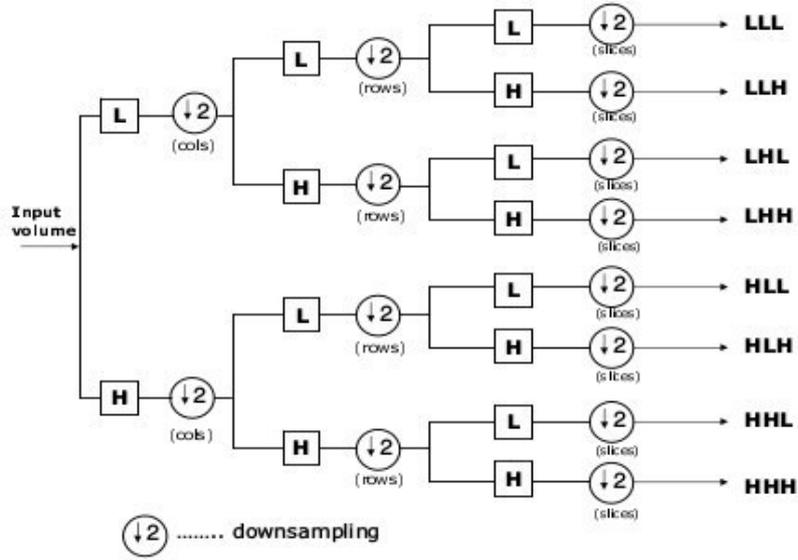


Figure 3.1: Diagram of Wavelet Equation apply to 3D image [23]

3.3 Visual Servoing using 3D wavelets transformation

For the command law using the multi-resolution wavelet transformation we have to define the interaction matrix. In this part the equation propose by [21] are extend to a 3D entering signal and what we obtain is :

$$\int \int \left(\frac{\partial \mathbf{I}}{\partial \mathbf{x}} \mathbf{v}_x(\mathbf{p}_i) + \frac{\partial \mathbf{I}}{\partial \mathbf{y}} \mathbf{v}_y(\mathbf{p}_i) + \frac{\partial \mathbf{I}}{\partial \mathbf{z}} \mathbf{v}_z(\mathbf{p}_i) + \frac{\partial \mathbf{I}}{\partial \mathbf{t}} \right) \psi_{\mathbf{j},\mathbf{m}}^{(1)}(\mathbf{p}_i(\mathbf{x}, \mathbf{y}, \mathbf{z})) \, dx dy dz = 0 \quad (3.2)$$

The notation is simplify (3.3) the term \mathbf{I} is referring at the luminous intensity of each pixel $\mathbf{p}_i(\mathbf{x}, \mathbf{y}, \mathbf{z})$.

$$I = I(p_i(x, y, z)) \quad (3.3)$$

knowing that:

$$\langle f, g \rangle = \int \int \mathbf{f}(\mathbf{x}) \mathbf{g}(\mathbf{x}) \, dx dy \quad (3.4)$$

we can write the (3.2) as:

$$\left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{x}} \mathbf{v}_x(\mathbf{p}_i), \psi_{\mathbf{j},\mathbf{m}}^{(1)} \right\rangle + \left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{y}} \mathbf{v}_y(\mathbf{p}_i), \psi_{\mathbf{j},\mathbf{m}}^{(1)} \right\rangle + \left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{z}} \mathbf{v}_z(\mathbf{p}_i), \psi_{\mathbf{j},\mathbf{m}}^{(1)} \right\rangle + \left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{t}}, \psi_{\mathbf{j},\mathbf{m}}^{(1)} \right\rangle = 0 \quad (3.5)$$

as done before the velocity of the point in the coordinate of the starting image is assumed to be unchanged in the new multi-resolution coordinate.

$$\begin{cases} \mathbf{v}_x(p_i) = \mathbf{v}_x(2^{-j}\mathbf{k}) & \forall \mathbf{x} \in \text{support } \psi_{\mathbf{j},\mathbf{m}}^{(1)} \\ \mathbf{v}_y(p_i) = \mathbf{v}_y(2^{-j}\mathbf{k}) & \forall \mathbf{y} \in \text{support } \psi_{\mathbf{j},\mathbf{m}}^{(1)} \\ \mathbf{v}_z(p_i) = \mathbf{v}_z(2^{-j}\mathbf{k}) & \forall \mathbf{z} \in \text{support } \psi_{\mathbf{j},\mathbf{m}}^{(1)} \end{cases} \quad (3.6)$$

so the velocity can be put outside the inner product:

$$\left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{x}}, \psi_{\mathbf{j},\mathbf{m}}^{(1)} \right\rangle \mathbf{v}_x(p_i) + \left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{y}}, \psi_{\mathbf{j},\mathbf{m}}^{(1)} \right\rangle \mathbf{v}_y(p_i) + \left\langle \frac{\partial \mathbf{I}}{\partial \mathbf{z}}, \psi_{\mathbf{j},\mathbf{m}}^{(1)} \right\rangle \mathbf{v}_z(p_i) + \frac{\partial}{\partial \mathbf{t}} \langle \mathbf{I}, \psi_{\mathbf{j},\mathbf{m}}^{(1)} \rangle = 0 \quad (3.7)$$

and after a integration by part the equation can be write as:

$$\left\langle \mathbf{I}, \frac{\partial \psi_{\mathbf{j},\mathbf{m}}^{(1)}}{\partial \mathbf{x}} \right\rangle \mathbf{v}_x(p_i) + \left\langle \mathbf{I}, \frac{\partial \psi_{\mathbf{j},\mathbf{m}}^{(1)}}{\partial \mathbf{y}} \right\rangle \mathbf{v}_y(p_i) + \left\langle \mathbf{I}, \frac{\partial \psi_{\mathbf{j},\mathbf{m}}^{(1)}}{\partial \mathbf{z}} \right\rangle \mathbf{v}_z(p_i) + \frac{\partial}{\partial \mathbf{t}} \langle \mathbf{I}, \psi_{\mathbf{j},\mathbf{m}}^{(1)} \rangle = 0 \quad (3.8)$$

using the development of [22] we can write that:

$$\dot{\mathbf{I}}_{(2^{j+1})} = - \begin{bmatrix} \mathbf{g}_{(2^{j+1})}^1(\mathbf{p}_i(\mathbf{x}, \mathbf{y}, \mathbf{z})) & \mathbf{g}_{(2^{j+1})}^2(\mathbf{p}_i(\mathbf{x}, \mathbf{y}, \mathbf{z})) & \mathbf{g}_{(2^{j+1})}^3(\mathbf{p}_i(\mathbf{x}, \mathbf{y}, \mathbf{z})) \end{bmatrix} \begin{bmatrix} \mathbf{v}_x(\mathbf{p}_i) \\ \mathbf{v}_y(\mathbf{p}_i) \\ \mathbf{v}_z(\mathbf{p}_i) \end{bmatrix} \quad (3.9)$$

We use the 3D point interaction matrix use by [19] to put in relation the velocity of the 3D point and the velocity of the camera:

$$\begin{bmatrix} \mathbf{v}_x(\mathbf{p}_i) \\ \mathbf{v}_y(\mathbf{p}_i) \\ \mathbf{v}_z(\mathbf{p}_i) \end{bmatrix} = 2^j \mathbf{L}_{3D} \mathbf{v}_c \quad (3.10)$$

The 2^j is used to take into account the effect of the down sampling. The development of the equation became:

$$\dot{\mathbf{I}}_{(2^{j+1})} = -2^j \begin{bmatrix} \mathbf{g}_{(2^{j+1})}^1(\mathbf{p}_i(\mathbf{x}, \mathbf{y})) & \mathbf{g}_{(2^{j+1})}^2(\mathbf{p}_i(\mathbf{x}, \mathbf{y})) & \mathbf{g}_{(2^{j+1})}^3(\mathbf{p}_i(\mathbf{x}, \mathbf{y})) \end{bmatrix} \mathbf{L}_{3D} \mathbf{v}_c \quad (3.11)$$

The vector of visual features is compose by value of the approximation image:

$$\mathbf{s}_{2^{j+1}} = (\mathbf{I}_{2^{j+1}(1,1)}, \dots, \mathbf{I}_{2^{j+1}(1,2)}, \mathbf{I}_{2^{j+1}(M,N)}) \quad (3.12)$$

the final interaction matrix is compute stacking in a single matrix the result of the 3.11 calculate for all the voxel of the volume.

$$\mathbf{L}_{s(2^{j+1})} = \begin{bmatrix} -2^j \begin{bmatrix} \mathbf{g}_{(2^{j+1})}^1(\mathbf{p}_i(\mathbf{1}, \mathbf{1})) & \mathbf{g}_{(2^{j+1})}^2(\mathbf{p}_i(\mathbf{1}, \mathbf{1})) & \mathbf{g}_{(2^{j+1})}^3(\mathbf{p}_i(\mathbf{1}, \mathbf{1})) \end{bmatrix} \mathbf{L}_{3D} \\ \mathbf{L}_{s(2^{j+1})(1,2)} \\ \vdots \\ \mathbf{L}_{s(2^{j+1})(M,N)} \end{bmatrix} \quad (3.13)$$

The equation used are taken following the idea of the 2D Wavelet Visual Servoing [17] and will be explained in the experimentation chapter.

3.4 Conclusion

In this chapter we have derive the equation of a 3D visual servoing starting from the approach use by [17]. This was possible because in calculating the 3D interaction matrix we relied on the method used by nadaue. The entire mathematical basis on which our algorithm will then be based has been presented. In the next chapter we will show how these mathematical formulas have been implemented in order to build a 6DOF controller using a 3D volume (C-Scan).

Chapter 4

Experimentation

The experiment set-up are composed by:

1. parallel robot and relative controller
2. optical coherence tomography

4.1 Parallel Robot

The robot use is the SpaceFAB SF-300 BS is a 3-PPSR parallel robot which consist in three serial chains, each with two actuated prismatic joints, one spherical joint and one revolution at the end. Thus, the resulting robot has 6 Dof at the moving platform enable higher positioning processes and a load capacity up to 3 kg center mounted.4.1

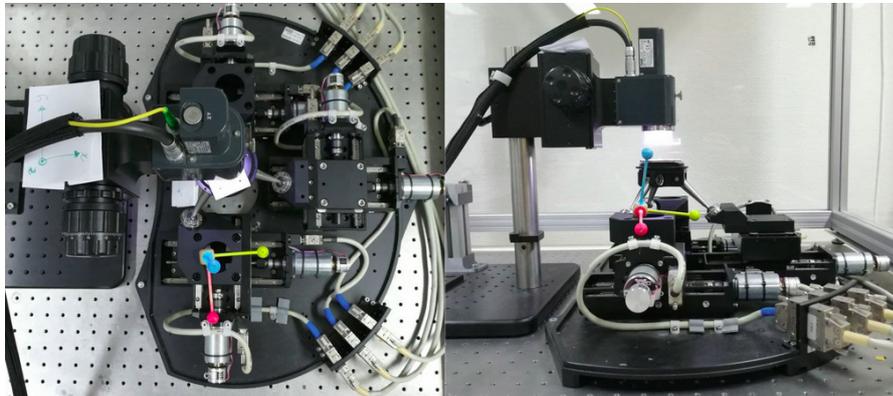


Figure 4.1: SpaceFAB SF-300 BS

4.2 Instrumentation

In the project we work with different PC, one where we have set the controller of the OCT and one where we have the controller of the parallel robot. 4.2

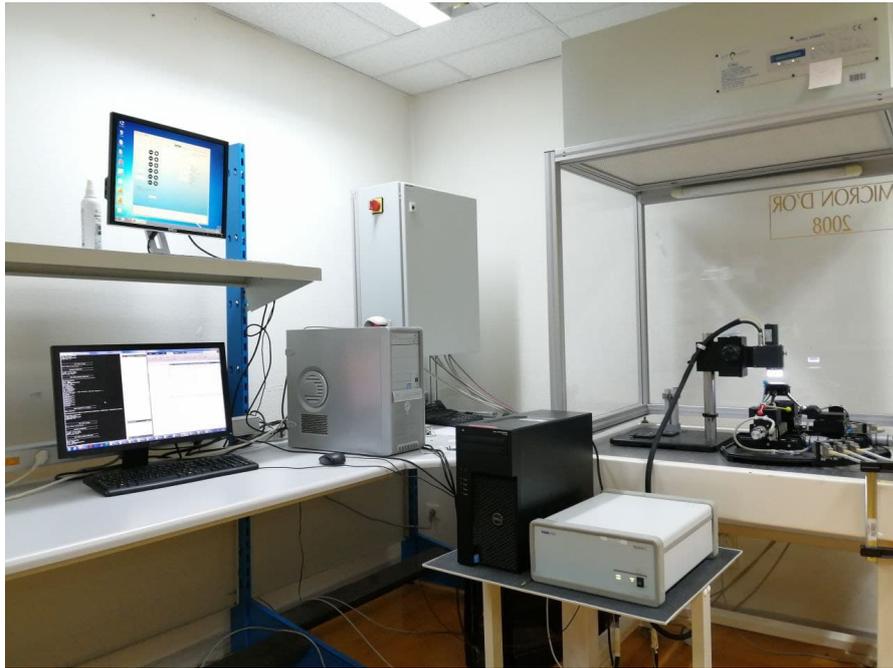


Figure 4.2: Instrumentation setup

Also the algorithm is split in two different environment: one that is Qt and one is Matlab. To put all this device and environment work properly together we use IP connection. This allows to send and receive data for all the devices involved in this project. 4.3

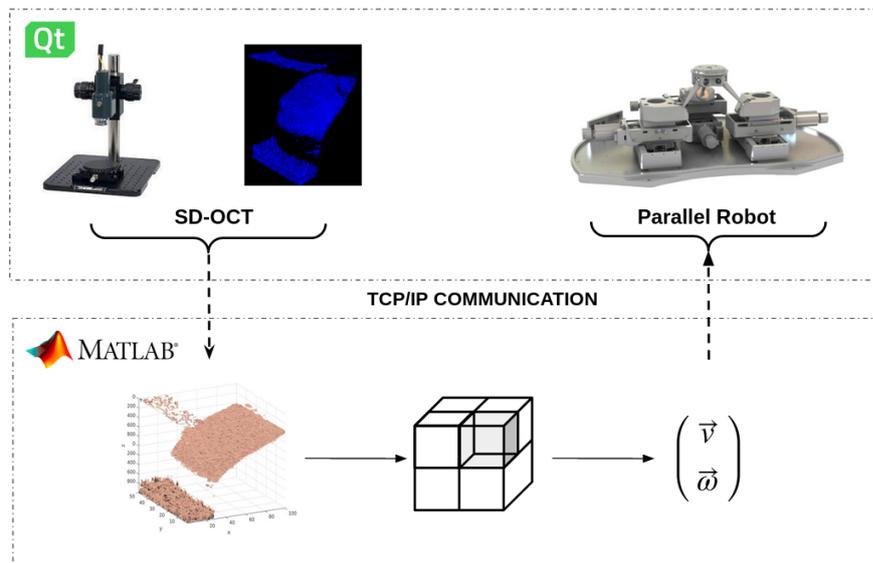


Figure 4.3: Control Strategy Flow Chart

4.2.1 Control Strategy

The control strategy develops is moving and wait, this mean that we send a velocity to the robot and set the amount of time that it has to maintain this velocity. The control strategy as said before is both develop on a C++ program using Qt software and on Matlab. To be more clear the control of the movement of the parallel robot are developed on Qt while the computation of control strategy are develop on Matlab. This is cause on Qt there isn't Wavelet library that satisfies our request while we can easily made all the computation on Matlab.

The controller are composed by the following step(4.3):

Algorithm 1 ControlFunction. In this function the control strategy are develop.

```
1: procedure CONTROLFUNCTION( $\alpha, \beta_1, \beta_2, f, \theta_0$ )
2:    $\triangleright$  In the first stage we set the desired position of the robot
3:   GotoPosition(RobotDesirePose)
4:   TakeCScan( $C - ScanParameter$ )
5:   SendVolumeToMatlab(Datasize)
6:    $\triangleright$  The robot is sent to the starting position
7:    $\triangleright$  Once the desired C-Scan are send to Matlab and the robot is in the starting position
   the control strategy finally begins
8:   GotoPosition(RobotStartPose)
9:   while (error is not converged) do
10:     $\triangleright$  The relate C-Scan of the current position are take
11:    TakeCScan( $C - ScanParameter$ )
12:     $\triangleright$  The Volume acquired are send to Matlab
13:    SendVolumeToMatlab(Datasize)
14:     $\triangleright$  The Interaction matrix are calculate
15:    InteractionMatrixInteractionMatrix(HighPass, LowPass, VolumeAcquire)
16:     $\triangleright$  Once the velocity of the robot are calculate it's send to the robot that perform this
17:     $\triangleright$  velocity for a fixed time slot
18:    SendVelocity(RobotPose, time)
19:  end while
20: end procedure
```

All the function that compose the control strategy will be explained in the following steps. The first function is the **GotoPosition**, here we fix a deride position of the robot. Set the vector contained the value in mm of the position where we ant to move the parallel robot according to his reference frame.

Algorithm 2 GoToPosition. This function is used to move the parallel robot to a fix position. Through a vector is possible to defined the wish position of the robot in the space.

```

procedure GOTOPosition( $\alpha, \beta_1, \beta_2, f, \theta_0$ )
2:   ▷ Initialization parallel robot
   ▷ Connection to the parallel robot
4:    $RobotPose \leftarrow Value(x, y, z, \alpha, \beta, \gamma)$    ▷ Include the three translation and three rotation
   ▷ Send Pose to the Robot
6:   ▷ Check Reach Pose
end procedure

```

Once the robot is in the desired position we acquired the corresponding C-scan. In the function **TakeCScan** gave the possibility to set all the parameter of the C-scan that we have to take. Starting from the dimension of the value up to the range of value that we want to take from the sample.

Algorithm 3 TakeCScan. In this function all the parameter relate to the characteristics of the desired volume are set and a C-scan is acquired in the current pose of the robot.

```

1: procedure TAKECSAN(OCT-Parameters)
2:   ▷ Initialization SD-OCT
3:   ▷ Set the variable of the that have to contain the parameter and the data that has to be
   taken
4:   ▷ Setting OCT parameter
5:    $Range_x \leftarrow Value(x)$    ▷ Range along x -axis in mm
6:    $Range_y \leftarrow Value(x)$    ▷ Range along y -axis in mm
7:    $PixelSize \leftarrow Value$    ▷ Pixel size in mm
8:    $Min_{dB} \leftarrow Value$    ▷ Minimum dynamic range of the scanned data
9:    $Man_{dB} \leftarrow Value$    ▷ Maximum dynamic range of the scanned data
10:  ▷ Start acquisition data procedure
11:  ▷ End acquisition
12: end procedure

```

The volume acquires are then send to Matlab in order to calculate the error of the acquired C-scan and the desired once.

Algorithm 4 SendVolumeToMatlab. Here the volume takes by the OCT are split into vector and send through socket packet.

```

1: procedure SENDVOLUMETOMATLAB( $Data_{size}, OCT - DataPointer$ )
2:    $\triangleright$  Initialization
3:    $VolSize_0 \leftarrow 0$   $\triangleright$  Value Assignment
4:    $VolSize_0 \leftarrow Data_{size}$   $\triangleright$  the  $Data_{size}$  contain the size of the acquired volume
5:    $\triangleright$  The total number of points contain in the volume are calculate
6:    $N_{point} = Data_{size}(x) * Data_{size}(y) * Data_{size}(z)$ 
7:    $\triangleright$  The maximum packet size (64k) are set in order to calculate the number of packet that
8:    $\triangleright$  have to be send
9:    $MaxN_{point} = sizeof(double) * 65536$ 
10:   $SizeN_{point} = sizeof(double) * N_{point}$ 
11:   $N_{packet} = SizeN_{point} / MaxN_{point}$ 
12:   $Pointer_{C-scan} \leftarrow OCT - DataPointer$ 
13:   $\triangleright$  The pointer are initialize to the pointer of the C-Scan
14:   $\triangleright$  Execution
15:  while  $i < N_{packet}$  do
16:    while  $j < Vec_{size}$  do
17:       $Vec_{send}(j) = Pointer_{C-scan}$ 
18:       $Pointer_{C-scan} ++$ 
19:       $j ++$   $\triangleright$  Update parameters
20:    end while
21:     $i ++$   $\triangleright$  Update parameters
22:     $\triangleright$  The vector are send to Matlab.
23:  end while
24: end procedure

```

After the C-scan that correspond to the desired position are sent to Matlab, the next step of the algorithm is where that the control strategy finally begins. The Robot is sent to the starting position and the corresponding volume is acquired. The volume is sent to Matlab were the algorithm **InteractionMatrix** calculate the interaction matrix of the corresponding volume. The volume is decomposed following the idiots of the 3. The decomposition are made using the equation of [22], where the wavelet decomposition are made using filter (low pass and high pass) the decomposed three-dimensional matrix are obtained through the product of convolution between the original matrix and the filter.

Algorithm 5 InteractionMatrix, In this algorithm the volume send by the OCT is decompose and the interaction matrix is calculate in order to perform the control strategy.

```

1: procedure INTERACTIONMATRIX( $\alpha, \beta_1, \beta_2, f, \theta_0$ )
2:    $\triangleright$  Once the volume are acquired by Matlab the function to calculate the decomposition of
   the volume are set.
3:    $\triangleright$  Identification of the derivation of the wavelet Daubechies (dB4).
4:   HighPass[0.23 - 0.330.150.05 - 0.03 - 0.010.0050.002]
5:   LowPass[-0.00, -0.02, 0.08, -0.02, -0.34, 0.29, 0.96, 0.46]
6:    $\triangleright$  Initialization of the 3D matrix
7:    $L_s(W_{3D}) = [(Vol_{Row} * Vol_{Cols} * Vol_{Prof}), 6]$ 
8:    $\triangleright$  Definition of the correlation matrix
9:    $\alpha = 2.5/50, \beta = 3.5/50, \gamma = 3.4/1024$   $\triangleright$  The coefficient are calculate to convert the value
   from pixel to mm
10:   $L_3D = [\alpha 00, 0\beta 0, 00\gamma]$ 
11:  while Rowisnotend do
12:    while Colsisnotend do
13:      while Profisnotend do
14:         $Vec = [\alpha * (i - (Row/2)), \beta * (j - (Cols/2)), \gamma * (i - (Prof/2))]$ 
15:         $Skew = [0 - Vec(3) - Vec(2), Vec(3) 0 - Vec(1), -Vec(2) Vec(1) 0]$ 
16:         $L_3D = [eye(3) - Skew]$ 
17:         $L_s(W_{3D})(c,:) = 2 * [G_x(i,j,k) G_y(i,j,k) G_k(i,j,k)] * Conv * L_3D$ 
18:         $c = c + 1$ 
19:      end while
20:    end while
21:  end while
22:   $\triangleright$  Calculation of the final interaction matrix
23:   $L_s(W_{3D})^* = pinv(L_s(W_{3D})^\top * L_s(W_{3D})) * L_s(W_{3D})^\top$ 
24:   $L_s(W_{3D}) = \lambda * inv(\mu * eye(6) + (L_s(W_{3D}))^\top * L_s(W_{3D})) * L_s(W_{3D})^*$ 
25:   $\triangleright$  Conversion acquire volume in vector
26:   $s = vec(Vol)$ 
27:   $\triangleright$  Conversion of the wish volume
28:   $s^* = vec(Vol_{wish})$ 
29:   $Velocity_{robot} = L_s(W_{3D}) * (s - s^*)$ 
30:   $\triangleright$  Velocity are send to parallel robot
31: end procedure

```

The **3DConvolution** generate the decomposition volume. The filter chosen are the Daubechies of quadratic order. This filter has been chosen cause they give the best result in terms of stability and precision.4.4

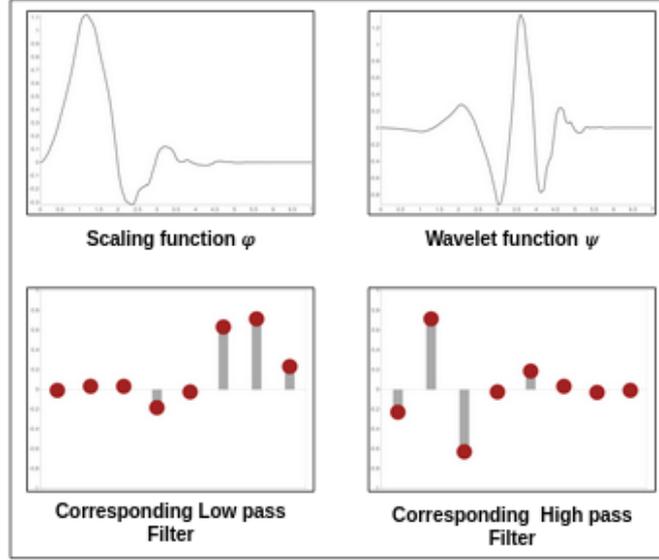


Figure 4.4: Daubechies of quadratic order, function and corresponding filter [10]

Algorithm 6 3DConvolution, Here is the develop the function in order to perform the convolution of the acquire volume. The filter use are the Daubechies(dB4)

```

1: procedure 3DCONVOLUTION( $\alpha, \beta_1, \beta_2, f, \theta_0$ )
2:    $\triangleright$  A new volume are initialize of the same size of the 3D image acquire
3:    $Volume_{Dec} \leftarrow Volume_{DataSize}$ 
4:    $\triangleright$  Convolution along Rows
5:   while  $k < Volume_{Prof}$  do
6:     while  $i < Volume_{Rows}$  do
7:       while  $j < Volume_{Cols}$  do
8:          $Volume_{Dec} = filter_{Rows} * Volume_{Acquire}(i, j, k)$ 
9:       end while
10:    end while
11:  end while
12:   $Volume_{Dec} = Volume_{Dec}(:, 1 : 2 : 2end, :)$   $\triangleright$  The column are decimate
13:   $\triangleright$  Convolution along Columns
14:  while  $k < Volume_{Prof}$  do
15:    ...
16:     $Volume_{Dec} = filter_{Column} * Volume_{Acquire}(i, j, k)$ 
17:  end while
18:   $\triangleright$  The same procedure are applied alongthe depth. The function give the possibility to set
    the filter that as to be use along each axes.
19: end procedure

```

4.2.2 Result

The image that we obtain from the OCT are shown in figure 4.5:

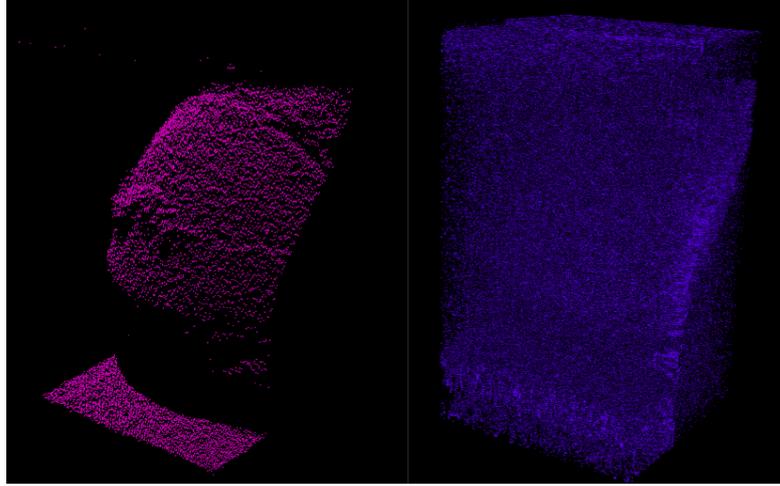


Figure 4.5: OCT volume, in the right data denoise in the left all the data acquired

As is visible the not all the data collect from the OCT are used, but we set an index in order to eliminate the noise. The data collected are then send to Matlab.

To be sure of the correct computation of the interaction matrix and this way of the velocity the volume received by Matlab are shown in the figure 4.6 is compared to the ones that was acquired.

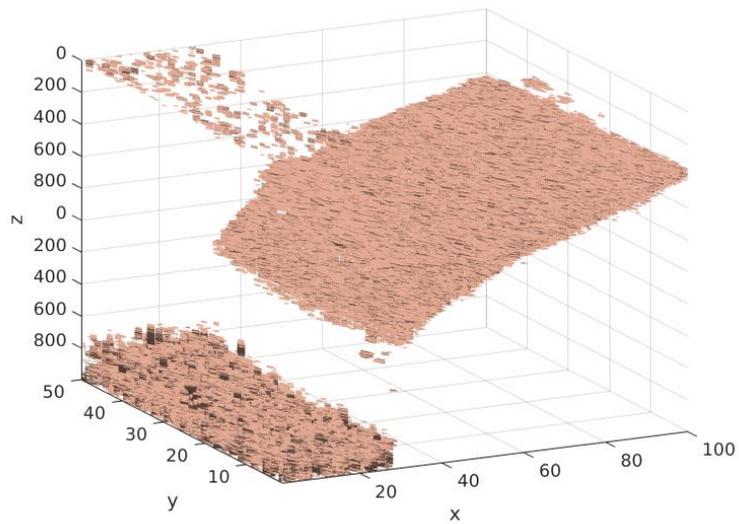


Figure 4.6: Volume acquired on Matlab

The test was structured in this way : the robot was moved to the End-Position, here the OCT is triggered in order to obtain the volume of the corresponding sample in this specific position.(This volume will be the reference for the control loop)

This C-Scan of the sample is sent to Matlab while the robot return in a Start position. At this moment the control loop starts: the OCT collect volume that is compared to the reference one in order to obtain at the end of the computation explained in the3 the velocity of the robot.This velocity is sent to the robot that return to the algorithm the new position and the loop restart. It will end only when the error will be very small.

The End-Position are chooses in advance and for the first test only simple translation is performed. In the first tests a simple translation on the Y-axis of the robot is set.But what is visible from the plot of the position reach by the robot 4.8 is that we have a movement in translation along the X and Z axis while the position in the Y axes remain constant for all the tests.

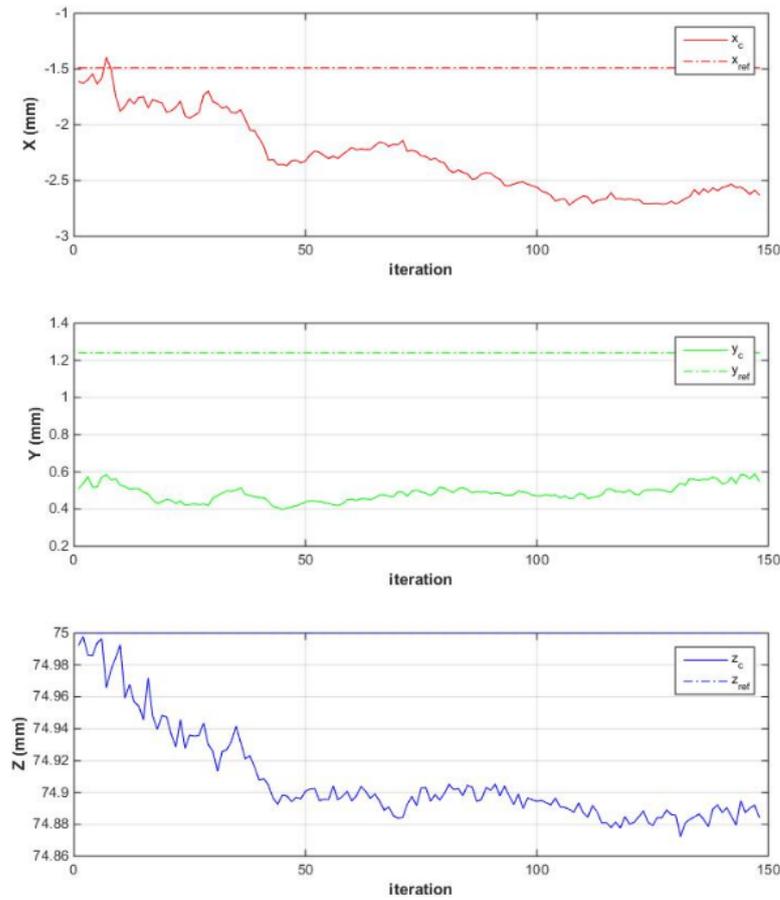


Figure 4.7: Plot the translation perform by the robot

Meanwhile, looking to the rotation angle we have a movement towards Y-axis while the movement around X and Z are overall zero.

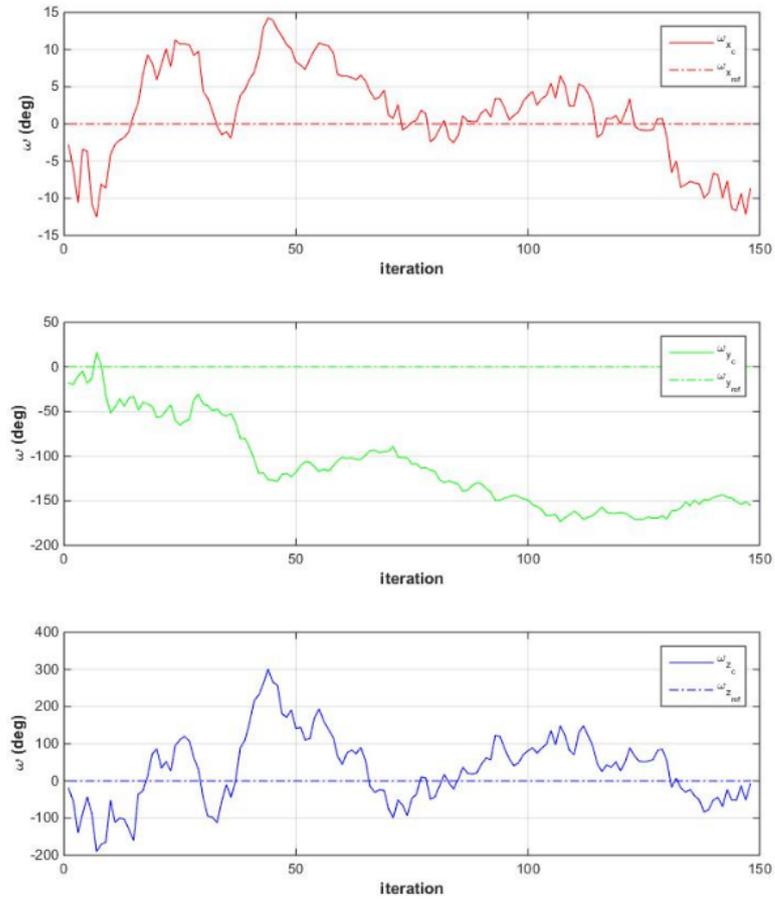


Figure 4.8: Plot of the rotation perform by the robot

Looking for the plot of the velocity 4.9 , the velocity in X and Z reach the most biggest value, while in Y it's been near zero. The opposite is in the rotation wherein Y has reach the biggest value. The average value of the robot is zero that means that there is no preferential direction in which the robot moves.

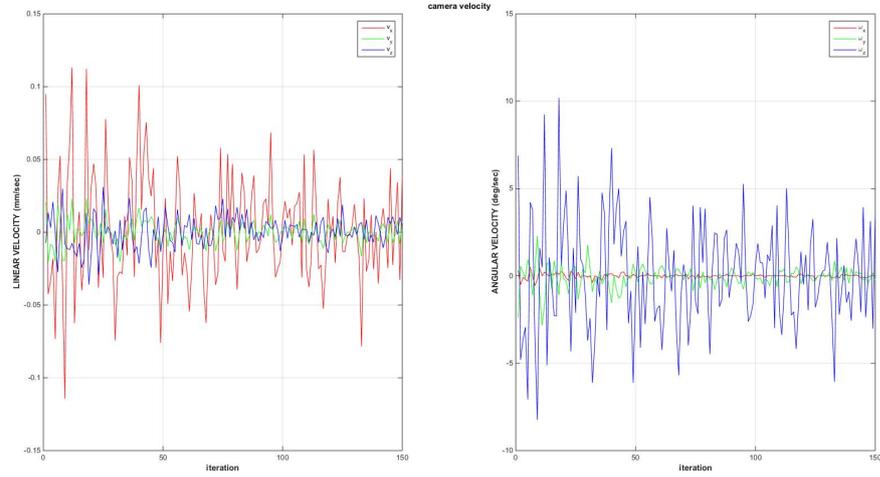


Figure 4.9: Plot of the velocity sent to the robot

This type of error can be reasonably addressed to a mismatch between the reference frame of the robot and of the 3D image. As is shown in figure 4.10 in the first part of the project the data sent to Matlab have a different reference frame respect to the reference frame of the OCT. This problem was solved calculating the rotation matrix that had to be multiplied to the interaction matrix in order to obtain from the control loop velocity that are referred to the reference frame of the parallel robot

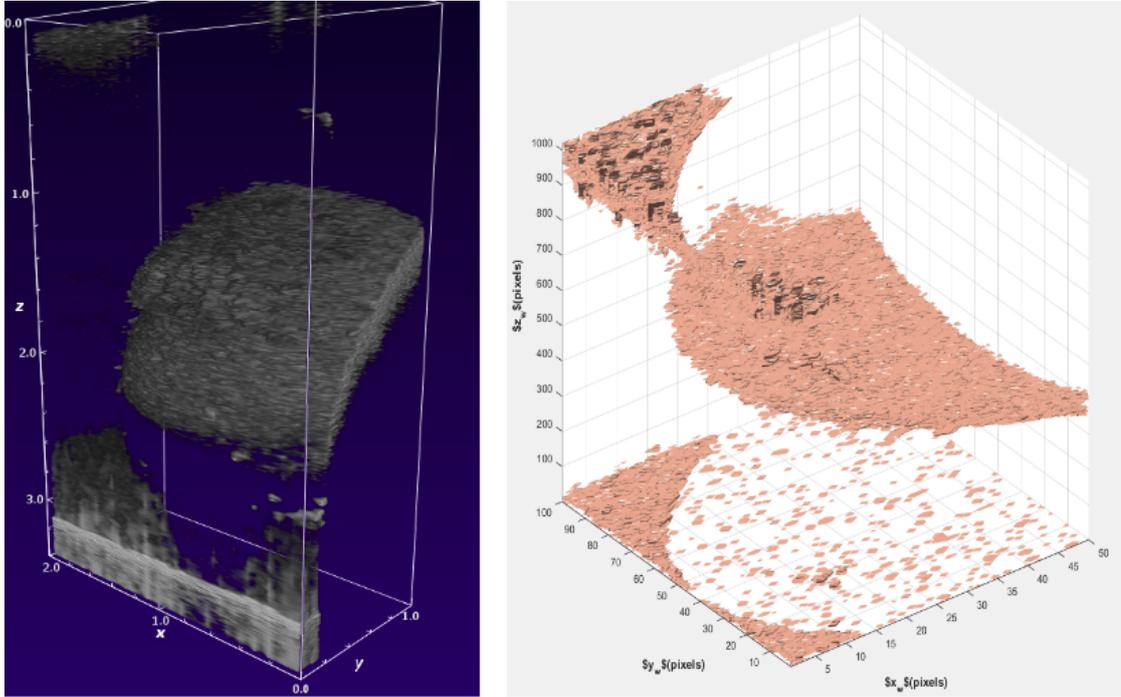


Figure 4.10: Difference of reference frame between the acquired volume (left) and the same volume on Matlab

4.3 Conclusion

In the last week of the work a lot of tests are made, and different rotational matrix is tried in order to figure out what is not working with the controller. The problem was addressed how the data taken from the OCT are sent and processed in the first part of the algorithm (**SendVolumeToMatlab**) and then from the Matlab (**InteractionMatrix**).

Chapter 5

Conclusion

The goal that we set ourselves was to succeed in creating a 6-degree controller of freedom that was able to achieve displacements along all directions of his Cartesian space. The theoretical development is based on an earlier development of a controller that using two-dimensional images was able to move the robot along two axes. Starting from that set of equations we used Bernard's developments to rewrite the three-dimensional equations and Nadeau's developments to be able to calculate an interaction matrix that was able to relate the reference system of the OCT with that of the parallel robot. The experimentation required a lot of computation power for the calculation of the interaction matrix because the images were of very important dimensions and the use of TCP/IP communication slowed the performance of the controller. In order to achieve the desired result, all the data passages that occur in the different phases of the program should be re-analyzed in such a way as to understand what kind of rotations occur in the data passage to build a matrix of rotations that puts in relations between the two reference systems (Acquisition and parallel robots) uniquely. Once you succeed in validating the controller's strategy you could switch to the realization of the controller completely in the Qt domain, in this you would increase exponentially the performance of the controller.

Bibliography

- [1] Elizabeth A. Stone. «Biopsy: Principles, Technical Considerations, and Pitfalls». In: *Veterinary Clinics of North America: Small Animal Practice* 25.1 (1995), pp. 33–45. ISSN: 0195-5616. DOI: [https://doi.org/10.1016/S0195-5616\(95\)50003-8](https://doi.org/10.1016/S0195-5616(95)50003-8). URL: <https://www.sciencedirect.com/science/article/pii/S0195561695500038> (cit. on pp. 2–4).
- [2] Ramanathan Vaidyanathan, Ren Hao Soon, Pan Zhang, Kuan Jiang, and Chwee Teck Lim. «Cancer diagnosis: from tumor to liquid biopsy and beyond». In: *Lab Chip* 19 (1 2019), pp. 11–34. DOI: 10.1039/C8LC00684A. URL: <http://dx.doi.org/10.1039/C8LC00684A> (cit. on p. 4).
- [3] Mayo Clinic Staff. «Biopsy: Types of biopsy procedures used to diagnose cancer». In: (Jan. 2020). URL: <https://www.mayoclinic.org/diseases-conditions/cancer/in-depth/biopsy/art-20043922> (cit. on p. 4).
- [4] OpticsPhotonics. «Toward an All-Optical Biopsy». In: (Jan. 2019). URL: https://www.optica-opn.org/home/articles/volume_30/april_2019/features/toward_an_all-optical_biopsy/ (cit. on p. 5).
- [5] MA Kara and JJGHM Bergman. «Autofluorescence imaging and narrow-band imaging for the detection of early neoplasia in patients with Barrett’s esophagus». In: *Endoscopy* 38.06 (2006), pp. 627–631 (cit. on p. 5).
- [6] Monalisa Filip, Sevastita Iordache, Adrian Saftoiu, and Tudorel Ciurea. «Autofluorescence imaging and magnification endoscopy». In: *World journal of gastroenterology : WJG* 17 (Jan. 2011), pp. 9–14. DOI: 10.3748/wjg.v17.i1.9 (cit. on p. 5).
- [7] Yasuo Sasagawa, Takuya Akai, Satoko Nakada, Hiroshi Minato, Osamu Tachibana, Takayuki Nojima, and Hideaki Iizuka. «Narrow band imaging-guided endoscopic biopsy for intraventricular and paraventricular brain tumors: clinical experience with 14 cases». In: *Acta Neurochirurgica* 156 (2014), pp. 681–687 (cit. on p. 6).
- [8] Kazuhiro Gono, Takashi Obi, Masahiro Yamaguchi, Nagaaki Oyama, Hirohisa Machida, Yasushi Sano, Shigeaki Yoshida, Yasuo Hamamoto, and Takao Endo. «Appearance of enhanced tissue features in narrow-band endoscopic imaging». In: *Journal of Biomedical Optics* 9.3 (2004), pp. 568–577. DOI: 10.1117/1.1695563. URL: <https://doi.org/10.1117/1.1695563> (cit. on p. 6).
- [9] F.A. Gallagher. «An introduction to functional and molecular imaging with MRI». In: *Clinical Radiology* 65.7 (2010). Molecular Imaging, pp. 557–566. ISSN: 0009-9260. DOI: <https://doi.org/10.1016/j.crad.2010.04.006>. URL: <https://www.sciencedirect.com/science/article/pii/S0009926010001832> (cit. on p. 6).

- [10] Ralph Weissleder and Umar Mahmood. «Molecular Imaging». In: *Radiology* 219.2 (2001). PMID: 11323453, pp. 316–333. DOI: 10.1148/radiology.219.2.r01ma19316. eprint: <https://doi.org/10.1148/radiology.219.2.r01ma19316>. URL: <https://doi.org/10.1148/radiology.219.2.r01ma19316> (cit. on pp. 7, 30).
- [11] Thorlabs. «OCT Systems Tutorial». In: (Jan. 2020). URL: https://www.thorlabs.com/newgrouppage9.cfm?objectgroup_id=10763 (cit. on pp. 8–11).
- [12] Andrew R. Miller et al. «Comparison Between Spectral-Domain and Swept-Source Optical Coherence Tomography Angiographic Imaging of Choroidal Neovascularization». In: *Investigative Ophthalmology and Visual Science* 58.3 (Apr. 2017), pp. 1499–1505 (cit. on p. 9).
- [13] Fischer J Aumann S Donner S. «Optical Coherence Tomography (OCT): Principle and Technical Realization». In: (Aug. 2019). URL: https://www.ncbi.nlm.nih.gov/books/NBK554044/%20doi:%2010.1007/978-3-030-16638-0_3 (cit. on p. 9).
- [14] Jocelyne Troccaz. *Robotique medicale. Traité IC2*. Hermès - Lavoisier, Feb. 2012, p. 447. URL: <https://hal.archives-ouvertes.fr/hal-00636339> (cit. on p. 12).
- [15] C Plaskos, P Cinquin, S Lavallée, and A J Hodgson. «Praxiteles: a miniature bone-mounted robot for minimal access total knee arthroplasty». In: *The International Journal of Medical Robotics and Computer Assisted Surgery* 1.4 (2005), pp. 67–79. DOI: <https://doi.org/10.1002/rcs.59>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/rcs.59>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/rcs.59> (cit. on p. 12).
- [16] François Chaumette, Seth Hutchinson, and Peter Corke. «Visual Servoing». In: *Handbook of Robotics, 2nd edition*. Ed. by B. Siciliano and O. Khatib. Springer, 2016, pp. 841–866. URL: <https://hal.inria.fr/hal-01355384> (cit. on pp. 13, 14, 16).
- [17] Mouloud Ourak. «Asservissement visuel direct fondé sur les ondelettes pour le positionnement automatique d’une sonde de tomographie par cohérence optique». PhD thesis. 2016 (cit. on pp. 13, 16–20, 23).
- [18] Lesley-Ann DufLOT. «Asservissement visuel direct utilisant les décompositions en shearlets et en ondelettes de l’image». PhD thesis. 2018 (cit. on pp. 13, 16).
- [19] Caroline Nadeau and Alexandre Krupa. «Intensity-based direct visual servoing of an ultrasound probe». In: *2011 IEEE International Conference on Robotics and Automation*. IEEE. 2011, pp. 5677–5682 (cit. on pp. 14, 16, 18, 20, 23).
- [20] Christophe Collewet and Eric Marchand. «Photometric visual servoing». In: *IEEE Transactions on Robotics* 27.4 (2011), pp. 828–834. URL: <https://hal.inria.fr/inria-00629834> (cit. on pp. 15, 18).
- [21] Bernard Espiau, François Chaumette, and Patrick Rives. «A new approach to visual servoing in robotics». In: *IEEE Transactions on Robotics and Automation* 8.3 (1992), pp. 313–326 (cit. on pp. 16, 17, 19, 20, 22).
- [22] Stephane G Mallat. «A theory for multiresolution signal decomposition: the wavelet representation». In: *IEEE transactions on pattern analysis and machine intelligence* 11.7 (1989), pp. 674–693 (cit. on pp. 18, 22, 28).
- [23] Aleš Procházka, Lucie Grajciarova, and Oldřich Vyšata. «Three-Dimensional Wavelet Transform in Multi-Dimensional Biomedical Volume Processing». In: (July 2011) (cit. on p. 21).
- [24] S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok. «A Novel Ultrathin Elevated Channel Low-temperature Poly-Si TFT». In: 20 (Nov. 1999), pp. 569–571.