Vision-based estimation and robot control

Ezio Malis

HABILITATION À DIRIGER DES RECHERCHES

Présentée à

l'Université de Nice-Sophia Antipolis

par

Ezio Malis

sur le thème :

Méthodologies d'estimation et de commande à partir d'un système de vision

Vision-based estimation and robot control

Soutenue le 13 Mars 2008 devant le jury composé de :

P.	Tarek	Hamel	Président
P.	Alessandro	De Luca	Rapporteur
D.	Bernard	Espiau	Rapporteur
P.	Gregory	Hager	Rapporteur
D.	Francois	Chaumette	Examinateur
P.	Philippe	Martinet	Examinateur
D.	Patrick	Rives	Examinateur

Table des matières

Introduction

The conception of full autonomous robotic systems is an enduring ambition for humankind. Nowadays theoretical and technological advances have allowed engineers to conceive complex systems that can replace humans in many tedious or dangerous applications. However, in order to considerably enlarge the flexibility and the domain of applications of such autonomous systems we still need to face several scientific problems at the crossroad of many domains like for example artificial intelligence, signal processing and non-linear systems control.

Amidst these challenges, the perception of the environment and the interaction of robotic systems with the environment are fundamental problems in the design of such autonomous systems. Indeed, the performance of an autonomous robot not only depends on the accuracy, duration and reliability of its perception but also on the ability to use the perceived information in automatic control loops to interact safely with the environment despite unavoidable modeling and measurement errors. Thus, automatic environment sensing and modeling, and robust sensor-based robot control are central scientific issues in robotics.

Several exteroceptive sensors are commonly used in robotics : contact sensors (e.g. force sensors, tactile sensors), GPS, sonars, laser telemeters, vision sensors, and even olfactory sensors. However, artificial vision is of particular importance and interest, mainly due to its versatility and extended range of applicability. It can be used both for the perception and modeling of the robot's environment and for the control of the robot itself. In this context, the spectrum of research is vast and I will focus on vision-based estimation and control problems. Vision-based estimation refers to the methods and techniques dedicated to the extraction of the information that can be useful not only for the modeling of the environment but also for robot control. In the case of robotic applications, major challenges are to increase the efficiency, the accuracy and the robustness of the estimation from visual data. Vision-based control refers to the methods and techniques dedicated to the use of visual information in automatic control loops. The challenges are to choose the appropriate visual information and to design stable control laws that are robust to modeling and measurement errors.

The objective of this document is to review, analyze and discuss my research work on these topics during the last ten years : two years as a Research Associate at the

University of Cambridge (United Kingdom) and eight years as a Research Scientist at INRIA Sophia-Antipolis (France). In order to acknowledge the contributions from all the colleagues and students that have collaborated with me, I have tried as far as possible to cite all our common publications. The document is divided into four chapters :

- Chapter 1 : **Modeling and Statement of Research Problems**.

  This introductory chapter is dedicated to presenting, as generally as possible, the vision-based estimation and control problems that have been considered by computer vision and robotic researchers. I will start with general models that can be used to classify and discuss the possible problems in parametric estimation and robot control. However, in order to establish the basis for the more specific problems that I will address in the rest of the document, I also briefly describe the specific models and assumptions that are commonly considered in the literature.
- Chapter 2 : **Numerical Analysis**.

  This chapter considers numerical analysis methods that can be applied both to visual parametric estimation and vision-based control. Our research work has focused on methods which suppose that the parametric functions are differentiable. The main contribution has been to propose a new numerical method called the Efficient Second-order approximation Method (ESM). The main advantage of this method is that, when applicable, it has a faster convergence rate and larger convergence domain than standard numerical methods. Thus, it can be used to improve the parametric estimation and control from visual data.
- Chapter 3 : **Vision-based parametric estimation**.

  This chapter details the vision-based parametric estimation problems that I have addressed. Our research work has focused on monocular and stereo vision systems composed by central cameras. The main contribution has been to propose efficient methods to solve the image registration problem (visual tracking) and the problems of localization and/or mapping. These methods have been designed to meet the requirements of efficiency, accuracy and robustness needed in real-time robotic applications.
- Chapter 4 : **Vision-based robot control**.

  This chapter details the vision-based robot control problems that I have addressed. Our research work has focused on vision-based control of robots that can be considered as ideal Cartesian motion devices (such as for example omnidirectional mobile robots or robot manipulators). The main contribution has been the design of robust vision-based control schemes that do not need an exact measure of the intrinsic parameters of the vision system and that do not need any “a priori” knowledge of the structure of the observed scene. A particular emphasis has been placed on the theoretical analysis of the robustness of the control laws with respect to errors on the uncertain parameters.

Chapitre 1

Modeling and problems statement

The research work presented in this document is related to both computer vision and automatic control. Indeed, my main objective has been to design robust methods to control a robot evolving autonomously in natural environments using real-time vision systems. In order to control the robot, we must not only be able to find its location with respect to the environment but we may also need to recover a representation of the environment itself. In this context, another objective of my research work has been devoted to real-time parametric estimation from visual data. The aim of this chapter is to give a general statement of the vision-based estimation and control problems that have been considered by computer vision and robotic researchers. A more detailed description of our contributions will be presented in Chapter 2, Chapter 3 and Chapter 4.

For both parametric estimation and control objectives, it is extremely important to have simple and accurate models of the robot, the environment and of the vision system. Although the ideal objective would be to control any complex system in any unknown environment with any vision system, such a general solution is out of reach. Thus, only limited models are currently used. The objective of this chapter is not only to introduce these models but also to understand what their limits are in terms of describing the reality.

The present chapter is organized as follows. First of all, I briefly introduce the models that are considered in this document : the representation of the pose and of the structure of rigid objects, their kinematics, as well as some robot models. Then, I will describe a general model based on an ideal plenoptic function that is able to describe any possible vision system. This function can be used to classify research work on parametric estimation from visual data and to analyze the related problems. Then, I will give a general overview of the research problems on vision-based robot control that I have considered.

1.1 Rigid bodies and robot models

1.1.1 Rigid body kinematics

Consider a rigid body in the Cartesian space (see Figure 1.1). A point $\mathcal{P}$ of the rigid body can be represented with its Cartesian coordinates $\mathbf{m}_r = (X_r, Y_r, Z_r) \in \mathbb{R}^3$ with respect to an inertial reference frame $\mathcal{F}_r$ equipped with a right-handed Cartesian coordinate system (in such a coordinate system, a rotation from the $\vec{x}$-axis to the $\vec{y}$-axis (about the $\vec{z}$-axis) is positive) which is attached to a point $\mathcal{O}$. Let $\mathcal{F}_c$ be a current frame attached to a point $\mathcal{C}$ and freely moving with respect to the inertial frame $\mathcal{F}_r$.

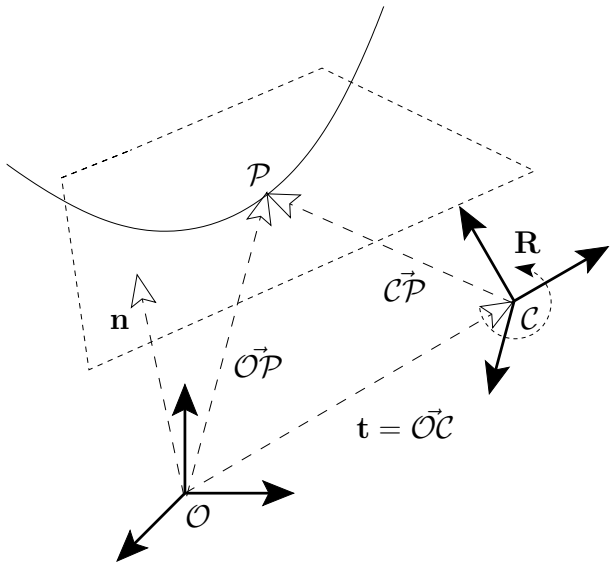


FIG. 1.1 – Configuration of a rigid body.

1.1.1.1 Representing pose and structure of a rigid body

The current pose of the rigid body (position and orientation) with respect to the moving frame can be described by a (4×4) homogeneous transformation matrix ${}^r\mathbf{T}_c$ containing the coordinates of the current frame $\mathcal{F}_c$ in the basis of the fixed frame $\mathcal{F}_r$. The matrix ${}^r\mathbf{T}_c \in \mathbb{SE}(3)$ (see for example [Warner 71, Varadarajan 74, Hall 03]) can be written as :

$$ {}^r\mathbf{T}_c = \begin{bmatrix} {}^r\mathbf{R}_c & {}^r\mathbf{t}_c \\ 0 & 1 \end{bmatrix} \tag{1.1} $$

where ${}^r\mathbf{t}_c = (\vec{\mathcal{O}\mathcal{C}})_r \in \mathbb{R}^3$ is the translation vector and ${}^r\mathbf{R}_c \in \mathbb{SO}(3)$ is the rotation matrix of frame $\mathcal{F}_c$ with respect to frame $\mathcal{F}_r$. The matrix ${}^r\mathbf{T}_c$ tells us where the current frame $\mathcal{F}_c$ is with respect to the reference frame $\mathcal{F}_r$. Let $\mathbf{m}_c = (X_c, Y_c, Z_c) \in \mathbb{R}^3$ be the coordinates of the same point $\mathcal{P}$ expressed in frame $\mathcal{F}_c$. Using the homogeneous coordinates $\underline{\mathbf{m}}_r = (\mathbf{m}_r, 1)$ and $\underline{\mathbf{m}}_c = (\mathbf{m}_c, 1)$, the coordinates of the point $\mathcal{P}$ in the current frame $\mathcal{F}_c$ can be obtained from the coordinates in the reference frame $\mathcal{F}_r$:

$$ \underline{\mathbf{m}}_c = {}^c\mathbf{T}_r \, \underline{\mathbf{m}}_r \tag{1.2} $$

where ${}^c\mathbf{T}_r = {}^r\mathbf{T}_c{}^{-1}$. The change of coordinates in equation (1.2) can also be written for non-homogeneous coordinates as follows :

$$\mathbf{m}_c = \boldsymbol{\tau}({}^c\mathbf{T}_r, \mathbf{m}_r) = {}^c\mathbf{R}_r\,\mathbf{m}_r + {}^c\mathbf{t}_r \tag{1.3}$$

The three points $\mathcal{O}$, $\mathcal{C}$ and $\mathcal{P}$ are coplanar. Thus we can obtain from equation (1.3) :

$$\mathbf{m}_r^\top \left[{}^r\mathbf{t}_c\right]_\times {}^r\mathbf{R}_c\,\mathbf{m}_c = \mathbf{m}_r^\top \left[{}^r\mathbf{E}_c\right]_\times \mathbf{m}_c = 0 \tag{1.4}$$

where the (3×3) matrix ${}^r\mathbf{E}_c = \left[{}^r\mathbf{t}_c\right]_\times {}^r\mathbf{R}_c$ is called the essential matrix.

When the point $\mathcal{P}$ belongs to a planar surface (see Figure 1.1) its coordinates satisfy the following equation :

$$\mathbf{n}_r^\top \mathbf{m}_r = 1 \tag{1.5}$$

where $\mathbf{n}_r$ is a vector normal to the plane such that $\|\mathbf{n}_r\| = 1/d_r$ (d_r being the distance between the plane and the point $\mathcal{O}$). Thus, the point $\mathbf{m}_r$ and $\mathbf{m}_c$ are related by the following equation :

$$\mathbf{m}_c = {}^c\mathbf{H}_r\,\mathbf{m}_r \tag{1.6}$$

where the (3×3) matrix ${}^c\mathbf{H}_r$ is called the homography matrix. This matrix can be written as follows :

$${}^c\mathbf{H}_r = {}^c\mathbf{R}_r + {}^c\mathbf{t}_r\,\mathbf{n}_r \tag{1.7}$$

Note that if $\det(\mathbf{H}) = 0$ the plane passes through the origin of the current frame.

1.1.1.2 Velocity of a rigid body

Consider a (4×4) matrix $\mathbf{A}_c \in \mathfrak{se}(3)$ (see [Warner 71, Varadarajan 74, Hall 03]) This matrix contains the instantaneous rotation and translation velocity vectors $\boldsymbol{\omega}_c$ and $\boldsymbol{\nu}_c$ expressed in the current frame (i.e. the velocity of the frame $\mathcal{F}_c$ expressed in the current frame itself) :

$$\mathbf{A}_c = \begin{bmatrix} [\boldsymbol{\omega}_c]_\times & \boldsymbol{\nu}_c \\ 0 & 0 \end{bmatrix} \tag{1.8}$$

The derivative of ${}^r\mathbf{T}_c$ can be written as :

$${}^r\dot{\mathbf{T}}_c = \begin{bmatrix} {}^r\dot{\mathbf{R}}_c & {}^r\dot{\mathbf{t}}_c \\ 0 & 0 \end{bmatrix} = {}^r\mathbf{T}_c\mathbf{A}_c \tag{1.9}$$

One may want to express the velocity of the current frame not in the coordinate system of the current frame itself but in the coordinate system of the reference frame. Let $\mathbf{A}_r \in \mathfrak{se}(3)$ be the velocity of the current frame expressed in the coordinates of the reference frame :

$$\mathbf{A}_r = \begin{bmatrix} [\boldsymbol{\omega}_r]_\times & \boldsymbol{\nu}_r \\ 0 & 0 \end{bmatrix} \tag{1.10}$$

The derivative of ${}^r\mathbf{T}_c$ can be written as :

$${}^r\dot{\mathbf{T}}_c = \mathbf{A}_r\,{}^r\mathbf{T}_c \tag{1.11}$$

From equations (1.9) and (1.11) we can compute $\mathbf{A}_r$ as a function of $\mathbf{A}_c$:

$$\mathbf{A}_r = {}^r\mathbf{T}_c \, \mathbf{A}_c \, {}^c\mathbf{T}_r \tag{1.12}$$

Then, one can easily compute this very well known change of frame for the velocity :

$$\begin{bmatrix} \boldsymbol{\nu}_r \\ \boldsymbol{\omega}_r \end{bmatrix} = \begin{bmatrix} {}^r\mathbf{R}_c & [{}^r\mathbf{t}_c]_\times \, {}^r\mathbf{R}_c \\ 0 & {}^r\mathbf{R}_c \end{bmatrix} \begin{bmatrix} \boldsymbol{\nu}_c \\ \boldsymbol{\omega}_c \end{bmatrix} \tag{1.13}$$

The velocity of a point in the current frame can easily be calculated by deriving equation (1.3) and using equation (1.9) :

$$\dot{\mathbf{m}}_c = -\boldsymbol{\nu}_c + [\mathbf{m}_c]_\times \, \boldsymbol{\omega}_c \tag{1.14}$$

1.1.2 Robot models

In this document, I will consider robots for which the kinematic model can be written as follows :

$$\dot{\mathbf{x}} = \sum_{i=1}^{m} u_i \mathbf{b}_i(\mathbf{x}) \tag{1.15}$$

where $\mathbf{x} \in \mathbb{R}^n$ is a vector which contains local coordinates of the configurations space, u_i are the control inputs and $\mathbf{b}_i(\mathbf{x})$ are the corresponding vector fields. I will now give two examples of models for holonomic and non-holonomic robots.

1.1.2.1 Holonomic robots

Holonomic robots (like manipulator or omnidirectional mobile robots) can be viewed as ideal Cartesian motion devices. Let me set in this case $\mathbf{x} = (\mathbf{t}, \mathbf{r}) \in \mathbb{R}^6$, where $\mathbf{t} \in \mathbb{R}^3$ is the translation vector and $\mathbf{r} \in \mathbb{R}^3$ is a local representation of the rotation. Thus, the vector $\mathbf{x}$ is a local parametrization of $\mathbb{SE}(3)$). Let me use the angle-axis representation $\mathbf{r} = \theta\mathbf{u}$ due to its links with Lie algebra (see [Warner 71], [Varadarajan 74], [Hall 03]). The matrix ${}^r\mathbf{T}_c$ can be written as a function of $\mathbf{x}$ as follows :

$${}^r\mathbf{T}_c = \mathbf{T}(\mathbf{x}) = \begin{bmatrix} \exp([\mathbf{r}]_\times) & \mathbf{t} \\ 0 & 1 \end{bmatrix} \tag{1.16}$$

The derivative of the matrix ${}^r\mathbf{T}_c$ can be written :

$${}^r\dot{\mathbf{T}}_c = {}^r\mathbf{T}_c \, \mathbf{A}_c \tag{1.17}$$

and thus $\mathbf{A}_c = \mathbf{T}(\mathbf{x})^{-1}\dot{\mathbf{T}}(\mathbf{x})$ (see equation 1.8). Let me set $\mathbf{v} = (\boldsymbol{\nu}_c, \boldsymbol{\omega}_c) \in \mathbb{R}^6$. Rearranging equation (1.17), we can write :

$$\dot{\mathbf{x}} = \mathbf{B}(\mathbf{x}) \, \mathbf{v} \tag{1.18}$$

where $\mathbf{B}(\mathbf{x})$ is a (6×6) matrix. We can identify this equation with equation (1.15) by setting $m = n = 6$, $u_i = v_i$ and $\mathbf{b}_i(\mathbf{x})$ being the i-th column of the matrix $\mathbf{B}(\mathbf{x})$. I will suppose that a low level controller exists such that the control input of the robot is the Cartesian velocity $\mathbf{v}$.

1.1.2.2 Non-holonomic robots

Non-holonomic robots have kinematic constraints on the velocity in the configuration space. This means that $m < n$ in equation (1.15). For example, if we consider a unicycle evolving on a planar surface we can set $m = 2$ and suppose that the vector $\mathbf{x} \in \mathbb{R}^3$ is a local representation of $\mathbb{SE}(2)$. Non-holonomic robots are generally critical non-linear systems. Some tasks that can easily be accomplished by holonomic robots become very complex for these systems. For example, positioning a non-holonomic robot can be extremely difficult even if the robot is controllable [Morin 04].

1.2 Image models

The formation of an image depends on the geometry of the scene and its photometric properties, on the distribution of the illumination sources and their photometric properties as well as on the intrinsic characteristics of the image sensor and its position and orientation with respect to the scene.

An image sensor is a device that captures light and converts it to an electric signal. Let me suppose that the device is composed of a rectangular grid of photo-detectors that may also be able to measure light wavelength (i.e. colors). Thus, an image can be considered as a $(s_u \times s_v \times s_c)$ tensor where s_v is the number of pixel rows, s_u is the number of pixel columns and s_c is the number of color channels. A common assumption is to suppose that the discrete signal measured by the image sensor has been sampled from a smooth continuous-time signal. Let $\mathbf{p}_c = (u, v)$ be the (2×1) vector in $\mathbb{R}^2$ containing the coordinates of a point in the sensor grid. Each point $\mathbf{p}_c$ corresponds to an entry of the tensor :

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\phi}, \mathbf{p}_c) \tag{1.19}$$

where $\boldsymbol{\phi}$ is a vector containing some photometric parameters (see Section 1.2.1 for details). We suppose that the light measured at point $\mathbf{p}_c$ comes from a visible 3D point of the scene $\mathbf{m}_c$. Thus, we suppose that a smooth projection function $\boldsymbol{\pi}$ exists such that :

$$\mathbf{p}_c = \boldsymbol{\pi}(\boldsymbol{\kappa}, \mathbf{m}_c) \tag{1.20}$$

where $\boldsymbol{\kappa}$ are some projection parameters that depend on the camera model (see Section 1.2.2). From equation (1.3) we can write $\mathbf{m}_c$ as a function of the coordinates $\mathbf{m}_r$ of the same point in the inertial reference frame and of the matrix ${}^r\mathbf{T}_c$ which allows the coordinate transformation from the current frame to the inertial reference frame. Plugging equations (1.20) and (1.3) into (1.19) and grouping all the coordinates $\mathbf{m}_r$ of the 3D points in a single vector $\boldsymbol{\gamma} = (\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_p)$ we obtain the following function :

$$\boldsymbol{\psi}(\boldsymbol{\phi}, \boldsymbol{\kappa}, \boldsymbol{\gamma}, \mathbf{T}) = \{\boldsymbol{\mathcal{I}}(\boldsymbol{\phi}, \boldsymbol{\pi}(\boldsymbol{\kappa}, \boldsymbol{\tau}(\mathbf{T}, \mathbf{m}_1))), \ldots, \boldsymbol{\mathcal{I}}(\boldsymbol{\phi}, \boldsymbol{\pi}(\boldsymbol{\kappa}, \boldsymbol{\tau}(\mathbf{T}, \mathbf{m}_p)))\} \tag{1.21}$$

This function can be interpreted as a generalization of the plenoptic function proposed by [Adelson 91]. The general plenoptic mapping in equation (1.21) is an ideal function that describes everything that is visible from any pose in space, for any camera, for any structure and for any illumination (and at any time).

I will give now some examples of standard models that are generally used to describe the plenoptic function. These models will be considered in the rest of the document.

1.2.1 Photometric models

1.2.1.1 Non Lambertian surfaces

According to major illumination models, the luminance at a particular pixel $\mathbf{p}$ is due to diffuse, specular and ambient reflections :

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\phi}, \mathbf{p}_c) = \boldsymbol{\mathcal{I}}_s(\boldsymbol{\phi}_s, \mathbf{p}_c) + \boldsymbol{\mathcal{I}}_d(\boldsymbol{\phi}_d, \mathbf{p}_c) + \boldsymbol{\mathcal{I}}_a \tag{1.22}$$

where $\boldsymbol{\phi} = (\boldsymbol{\phi}_s, \boldsymbol{\phi}_d)$ and $\boldsymbol{\phi}_s$ and $\boldsymbol{\phi}_d$ are parameters that depend on the illumination model (e.g. [Blinn 77], or [Cook 82]). The ambient intensity $\boldsymbol{\mathcal{I}}_a$ is constant. The parameters $\boldsymbol{\phi}_s$, defining the specular reflections, also depend on the position of the image sensor with respect to the observed surface. The diffusion parameters $\boldsymbol{\phi}_d$ depend amongst other parameters (e.g. the albedo) on how the surface faces the light sources.

1.2.1.2 Lambertian surfaces

Lambertian surfaces or "ideal diffuse surfaces" are surfaces for which the image intensity of a pixel depends only on how the surface faces the light sources (i.e. there are no specular reflexions). Equation (1.22) can be written

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\phi}, \mathbf{p}_c) = \boldsymbol{\mathcal{I}}_d(\boldsymbol{\phi}_d, \mathbf{p}_c) + \boldsymbol{\mathcal{I}}_a, \tag{1.23}$$

where $\boldsymbol{\phi} = \boldsymbol{\phi}_d$ contains the diffusion parameters only. Thus, if we observe a static environment and the light sources are fixed, then two corresponding pixels in two different images will have the same intensity. This particular case corresponds to the well known "Brightness Constancy Assumption".

1.2.2 Projection models

As already mentioned we suppose that a parametric smooth function $\boldsymbol{\pi}$ exists such that :

$$\mathbf{p}_c = \boldsymbol{\pi}(\boldsymbol{\kappa}, \mathbf{m}_c) = \boldsymbol{\pi}(\boldsymbol{\kappa}, \boldsymbol{\tau}({}^c\mathbf{T}_r, \mathbf{m}_r))$$

The parameters contained in the vector $\boldsymbol{\kappa}$ are called the camera intrinsic parameters and depend on the considered projection model. There are two main groups of cameras :

- Central cameras : for a central camera all the light rays forming the image meet at the same point called the center of projection.
- Non-central cameras : for a non central camera the light rays forming the image meet in more that one center of projection. The set of centers of projection is called a caustic and it determines the characteristics of the camera.

In this document, I will consider central cameras only. I will now describe the models that are generally used for this type of camera.

1.2.2.1 Central catadioptric cameras

Using the unified model for central catadioptric cameras proposed by [Baker 99], [Geyer 00], and [Barreto 02], the image point $\mathbf{p}_c$ is obtained by a projection of $\mathbf{m}_c$ onto a unit sphere, followed by a map projection onto a virtual plane, followed by a coordinate distortion and finally followed by an affine coordinate transformation. The point $\mathbf{m}_c$ can be projected on the unit sphere $\mathcal{S}$ in a 3D point having coordinates $\mathbf{s}_c = (X_s,\ Y_s,\ Z_s)$:

$$\mathbf{s}_c = \frac{\mathbf{m}_c}{\|\mathbf{m}_c\|} \tag{1.24}$$

Then, the projection from a point $\mathbf{s}_c$ to a 2D image point $\mathbf{p}_c$ can be described by a projective mapping $\mathbf{c} : \mathbb{S}^2 \mapsto \mathbb{R}^2$:

$$\mathbf{p}_c = \mathbf{c}(\boldsymbol{\kappa}, \mathbf{s}_c)$$

where $\boldsymbol{\kappa}$ is a vector which contains the camera intrinsic parameters. I will suppose that the map $\mathbf{c}$ is invertible and $\mathbf{s}_c = \mathbf{c}^{-1}(\boldsymbol{\kappa}, \mathbf{p}_c)$. For example, the function $\mathbf{c}$ for a generic camera can be written as follows. The map projection of the point $\mathbf{s}_c$ on the sphere (from a projection point $\mathcal{C}_p$) to a point $\mathbf{q}_c = (x,\ y)$ on a virtual plane can be written :

$$\mathbf{q}_c = \mathbf{h}(\xi, \mathbf{s}_c) = \left(\frac{X_s}{Z_s - \xi}, \frac{Y_s}{Z_s - \xi} \right) \tag{1.25}$$

where ξ is a parameter which defines the characteristic of the camera : The point $\mathbf{q}_c$ may be distorted to the point $\mathbf{q}_d = (x_d, y_d)$:

$$\mathbf{q}_d = \mathbf{d}(\boldsymbol{\delta}, \mathbf{q}_c)$$

where $\boldsymbol{\delta}$ is a vector containing the parameter of the distortion model. Typical distortion models are given in [Weng 92]. Finally, the virtual point is projected on the image plane into the point $\mathbf{p}_c = (u, v)$:

$$\mathbf{p}_c = \mathbf{k}(\boldsymbol{\rho}, \mathbf{q}_d) = (\rho_1\, x_d + \rho_2\, y_d + \rho_3, \rho_4\, y_d + \rho_5) \tag{1.26}$$

It has been shown by [Courbon 07] that this model can also approximate well several projection models for central dioptric cameras (fish-eye cameras).

1.2.2.2 Pinhole cameras

In the particular case when there is no mirror $\xi = 0$ and no image distortion $\boldsymbol{\delta} = 0$ the camera is a pinhole. Using homogeneous coordinates, the equation (1.26) can be written as :

$$\underline{\mathbf{p}}_c = \mathbf{K}\, \underline{\mathbf{q}}_c \tag{1.27}$$

where $\mathbf{K}$ is a (3×3) triangular matrix containing the camera intrinsic parameters :

$$\mathbf{K} = \begin{bmatrix} f & f\,s & u_0 \\ 0 & f\,r & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1.28}$$

where (u_0, v_0) are the coordinates of the principal point (in pixels), f is the focal length measured in pixels, s is the skew and r is the aspect ratio. Note that equation 1.27 can also be written :

$$\underline{\mathbf{p}} \propto \begin{bmatrix} \mathbf{K} & \mathbf{0} \end{bmatrix} \underline{\mathbf{m}}$$

This projection model has been widely used since it is a very good approximation of many commercial cameras with good quality lenses.

1.3 Problems statement

The principal objective of my research work has been to design robust methods for parametric visual estimation and for controlling a robot in large scale environments using a vision system. The information acquired by the vision system can be described by a non-linear system of equations (see Section 1.1). The evolution of the state of the robot can be described by a system of differential equations (see Section 1.2). Thus, in general the overall non-linear dynamic system can be represented by the following state space equations :

$$\dot{\mathbf{x}}(t) = \boldsymbol{g}(\mathbf{x}(t), \mathbf{u}(t), t) \tag{1.29}$$
$$\mathbf{y}(t) = \boldsymbol{h}(\boldsymbol{\eta}(t), \mathbf{x}(t)) \tag{1.30}$$

where $\mathbf{x}$ is the state of the system, $\mathbf{u}$ is the control input, $\boldsymbol{g}$ is a smooth vector function describing the evolution of the state of the system and $\mathbf{y}$ is the output vector represented by a smooth vector function $\boldsymbol{h}$. The function $\boldsymbol{h}$ depends on the pose of the vision system contained in $\mathbf{x}$ but also on the camera intrinsic parameters and on the structure of the environment contained in the vector $\boldsymbol{\eta}$. The problems related to the estimation of this information from visual data are described in Section 1.3.1. Then, in Section 1.3.2 I will discuss the problems related to the stabilization of the non-linear dynamic system using visual information.

1.3.1 Estimation from visual data

The function $\boldsymbol{h}$ that describes the behavior of the output $\mathbf{y}$ of the vision system can be represented using the general plenoptic function described in Section 1.2. Let me consider the most general case where the vision system is composed of n_c synchronized cameras with varying intrinsic parameters (e.g. zooming cameras) that observe a non-rigid scene. In this case the output $\mathbf{y}(t)$ of the vision system contains a collection of images acquired at the same time by all the cameras of the vision system. The output $\mathbf{y}(t)$ can be obtained by rearranging all the n_c plenoptic functions in a single vector :

$$\mathbf{y}(t) = \{\boldsymbol{\psi}(\boldsymbol{\phi}_1(t), \boldsymbol{\kappa}_1(t), \boldsymbol{\gamma}(t), \mathbf{T}_2(t)), \ldots, \boldsymbol{\psi}(\boldsymbol{\phi}_{n_c}(t), \boldsymbol{\kappa}_{n_c}(t), \boldsymbol{\gamma}(t), \mathbf{T}_{n_c}(t))\} \tag{1.31}$$

Suppose we have a set of visual data $\mathbf{y}(t_i)$ acquired at times t_i, $\forall i \in \{1, 2, n_i\}$, (i.e. a collection of images or a video sequence). Suppose also that we have a perfect model of the plenoptic function (see Section 1.2). Thus, some parameters $\overline{\boldsymbol{\phi}}_k(t_i)$, $\overline{\boldsymbol{\kappa}}_k(t_i)$, $\overline{\mathbf{T}}_k(t_i)$

and $\overline{\mathbf{m}}_j(t_i)$ exist such that $\boldsymbol{\mathcal{I}}(\overline{\boldsymbol{\phi}}_k(t_i), \boldsymbol{\pi}(\overline{\boldsymbol{\kappa}}_k(t_i), \boldsymbol{\tau}(\overline{\mathbf{T}}_k(t_i), \overline{\mathbf{m}}_j(t_i)))) = \overline{\boldsymbol{\mathcal{I}}}_{ijk}$, $\forall i, j, k$. Thus, the problem of parameters estimation from the observed visual data is equivalent to the solution of the following system of non-linear equations $\forall i, j, k$:

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\phi}_k(t_i), \boldsymbol{\pi}(\boldsymbol{\kappa}_k(t_i), \boldsymbol{\tau}(\mathbf{T}_k(t_i), \mathbf{m}_j(t_i)))) = \overline{\boldsymbol{\mathcal{I}}}_{ijk} \tag{1.32}$$

An exhaustive description of all possible methods and algorithms that have been proposed to solve this non-linear problem is beyond the scope of this document. The reader may refer to numerous well-known textbooks like [Faugeras 93], [Hartley 00], [Faugeras 01] [Ma 03]. Despite the impressive achievements of computer vision scientists, there still exist three challenging sets of problems related to the solution of this non-linear system of equations.

The first set of problems concerns the modeling of the plenoptic function. The parametric model of the plenoptic function is supposed to be perfect. Obviously, this assumption is not true in general. Thus, the problem is how to find simple, accurate, and general models of the plenoptic function. Simple means that only a few parameters are needed to represent the model. Accurate means that the model can fit the measures accurately. General means that the same model can be applied to most of the existing vision systems.

The second set of problems concerns data association. The photodetectors of the imaging device are supposed to instantaneously measure the light coming from an ideal 3D point. These assumptions are not true in general since during the exposure time the photodetectors integrate the light coming from a portion of a 3D surface. When the shutter speed is not fast enough we can have blur on moving objects. Moreover, as the pixel resolution of the imaging device is fixed, the intensity of the same portion of a 3D surface will not always be measured by a single pixel. Even assuming that the shutter is almost instantaneous and the light is coming from an ideal 3D point, how do we obtain a precise correspondence of the 3D coordinates $\mathbf{m}_j(t_i)$ and the pixel intensity $\boldsymbol{\mathcal{I}}_{ijk}$? Similarly, how we can obtain a precise correspondence of the intensity $\boldsymbol{\mathcal{I}}_{ijk}$ of one image point and the intensity of the corresponding points in the other images ? This last problem can be simplified assuming the visual data come from a video sequence acquired with a sufficiently fast frame rate. This allows us to suppose that two corresponding pixels are adjacent in two consecutive images.

The third set of problems concerns the solution of the non-linear system. The plenoptic function is supposed to be smooth and noiseless. This assumption is generally not true but it allows us to use numerical methods that compute the derivatives of the function. Even supposing that the problem is well posed, how can we robustly and efficiently solve the system of non-linear equations for real-time applications ? Robustness with respect to aberrant measures (i.e. outliers) is necessary since it is impossible to obtain a perfect representation of the plenoptic function. Efficiency is extremely important for fast real-time application (like vision-based control) and it can be obtained in two ways. One possibility is to simplify the system of equations. For example we can eliminate some of the unknowns to obtain a smaller nonlinear system (e.g. subspace methods , invariance ...). The second possibility is to use efficient numerical methods for the solution of the nonlinear system. Whatever the method used to solve the nonlinear system we need to know if the problem is well posed. Computer vision and robotic scientists soon

realized that in this very general form the problem may not have a unique solution (not surprisingly since inverse problems are generally not well posed). Typically, there are more unknown parameters than equations. Thus, some additional constraints are added to the original nonlinear system in order to obtain a well posed problem. The exhaustive description of all possible assumptions is out of the scope of this document. However some of them are very common for many problems that are of interest for real-time applications. For example, a very usual assumption is to suppose that the camera observes a rigid Lambertian surface that is static with respect to the sources of illumination. This assumption allows us to considerably reduce the number of unknown parameters since we can set $\boldsymbol{\phi}_k(t_i) = \boldsymbol{\phi}$ and $\mathbf{m}_j(t_i) = \mathbf{m}_j$. Another technique to reduce the number of unknown parameters is regularization. For example, we can suppose that the observed scene is piecewise planar or that it is composed of smooth surfaces. Temporal regularization is also a common assumption (i.e. assuming that the parameters vary slowly in time or that the collection of images comes from a video sequence). The estimation problem can also become well posed if we assume that the true values of some parameters are already known "a priori". Even if this may be a very strong assumption, this solution has been widely used. In the following sections I will detail three standard classes of problems that have been considered by computer vision and robotic scientists.

1.3.1.1 Calibration of the vision system

One of the first problems addressed by robotic vision scientists has been the estimation of the intrinsic parameters of a vision system. The reason why these parameters deserve particular attention is that in many applications they can be considered constant so that they can be estimated once and for all. Suppose that we observe several images of a known rigid object (a calibration grid for example) and that the intrinsic parameters of the cameras are constant $\boldsymbol{\kappa}_k(t_i) = \boldsymbol{\kappa}_k$ (i.e. the cameras do not zoom). Since the object is known the coordinates $\mathbf{m}_j(t_i) = \overline{\mathbf{m}}_j$ of all the 3D points of the object are known in a reference frame. Finally, suppose that we know that the intensity $\overline{\boldsymbol{\mathcal{I}}}_{ijk}$ measured by each image sensor corresponds to the point $\overline{\mathbf{m}}_j$. Thus, we can solve the following system of non linear equations :

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\phi}_k(t_i), \boldsymbol{\pi}(\boldsymbol{\kappa}_k, \boldsymbol{\tau}(\mathbf{T}_k(t_i), \overline{\mathbf{m}}_j))) = \overline{\boldsymbol{\mathcal{I}}}_{ijk} \tag{1.33}$$

This problem is called the "camera calibration problem" and several solutions have been proposed in the literature. Even if only one image of a known grid can be sufficient to solve the problem, several images are generally considered in order to obtain a precise estimation of the parameters.

1.3.1.2 Localization and/or mapping

Suppose now that the camera intrinsic parameters have been recovered once and for all $\boldsymbol{\kappa}_k = \overline{\boldsymbol{\kappa}}_k$ and that we observed a rigid scene (i.e. $\mathbf{m}_j(t_i) = \mathbf{m}_j$). Then one may be interested to solve the simultaneous localization and mapping problem :

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\phi}_k(t_i), \boldsymbol{\pi}(\overline{\boldsymbol{\kappa}}_k, \boldsymbol{\tau}(\mathbf{T}_k(t_i), \mathbf{m}_j))) = \overline{\boldsymbol{\mathcal{I}}}_{ijk} \tag{1.34}$$

where the structure and the translations of the camera can be estimated only up to a scale factor if we do not have any additional constraint.

Another problem of interest in robotic vision is the reconstruction of the structure from known motion. Indeed, if a calibrated vision system is mounted on a well calibrated robot we are able to measure the displacement $\overline{\mathbf{T}}_k(t_i)$ for each image. Thus the problem becomes :

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\phi}_k(t_i), \boldsymbol{\pi}(\overline{\boldsymbol{\kappa}}_k, \boldsymbol{\tau}(\overline{\mathbf{T}}_k(t_i), \mathbf{m}_j))) = \overline{\boldsymbol{\mathcal{I}}}_{ijk} \tag{1.35}$$

Another problem of interest is the localization of the vision system when the model of the object is known. The localization can be obtained by solving the following system :

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\phi}_k(t_i), \boldsymbol{\pi}(\overline{\boldsymbol{\kappa}}_k, \boldsymbol{\tau}(\mathbf{T}_k(t_i), \overline{\mathbf{m}}_j))) = \overline{\boldsymbol{\mathcal{I}}}_{ijk} \tag{1.36}$$

This problem is closely related to the camera calibration problem but here only the camera displacement and the photometric parameters are unknown. The localization of the vision system can also be obtained if the structure $\mathbf{m}_j$ is unknown but the vision system has multiple calibrated cameras (e.g. a calibrated stereo pair). In this case the poses of the cameras with respect to each other are known.

1.3.1.3 Self-calibration of the vision system

If we observe an unknown rigid object and the cameras' intrinsic parameters are not constant (e.g. if we consider zooming cameras) then the problem of estimating all possible parameters is called the "camera self-calibration problem" :

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\phi}_k(t_i), \boldsymbol{\pi}(\boldsymbol{\kappa}_k(t_i), \boldsymbol{\tau}(\mathbf{T}_k(t_i), \mathbf{m}_j))) = \overline{\boldsymbol{\mathcal{I}}}_{ijk} \tag{1.37}$$

This problem is generally not well posed and it cannot always be solved. For example, consider the very simple case of a pinhole camera with constant camera parameters $\boldsymbol{\kappa}_k(t_i) = \boldsymbol{\kappa}_k$ observing a motionless Lambertian surface. The camera self-calibration problem cannot be solved if the camera does not rotates. If all the camera intrinsic parameters vary with time then the problem cannot be solved even if the camera rotate. Note also that if we do not have any additional metric knowledge, the structure and the translations of the cameras can be estimated only up to a scale factor

1.3.2 Control from visual data

The first step to controlling the robot using the visual information is to define the task to be accomplished by the robot. There are several possible tasks that can be defined, for example to reach a reference position or a reference velocity. The definition of the task also depends on the configuration of the vision system and on the type of robot. For example, the vision system can be mounted on the end-effector of the robot (this configuration is called "*eye-in-hand*") or not (this configuration is called *eye-to-hand*). An exhaustive description of all possible tasks and configurations is beyond the scope of this document. The reader may refer to numerous books, tutorials and surveys that have been published on the subject, like [Hashimoto 93, Hutchinson 96, Malis 02b, Chaumette 06,

Chaumette 07]. Research in visual servoing initially focused on the problem of controlling the pose of a camera, assuming that the camera can (locally) move freely in all directions. This is the case, for instance, when the camera is mounted on an omnidirectional mobile robot, or on the end-effector of a classical manipulator endowed with (at least) 6 degrees of freedom. This is equivalent to viewing the robot as an ideal Cartesian motion device. The control part of the problem is then simplified since standard control techniques, like pure state feedback linearization, can be applied. The case of robotic vision-carriers subjected to either nonholonomic constraints (like car-like vehicles) or underactuation (like most aerial vehicles) raises a new set of difficulties.

Consider now the case when the robot can be considered as an ideal Cartesian motion device. In order to simplify the discussion a specific task is considered here : positioning a holonomic robot with respect to a motionless object using the information acquired by a *eye-in-hand* vision system. Thus, the state of the system $\mathbf{x}$ is locally homeomorphic to $\mathbb{SE}(3)$ and the function $\boldsymbol{g}$ that describes the behavior of the state $\mathbf{x}$ can be modeled as in Section 1.1. The function $\boldsymbol{h}$ that describes the behavior of the output $\mathbf{y}$ can be modeled as in Section 1.2. Thus, the system of equations (1.29) can be rewritten as follows :

$$\dot{\mathbf{x}}(t) = \mathbf{B}(\mathbf{x}(t))\,\mathbf{v}(t) \tag{1.38}$$

$$\mathbf{y}(t) = \boldsymbol{h}(\boldsymbol{\eta}(t), \mathbf{x}(t)) \tag{1.39}$$

where the velocity of the camera $\mathbf{v}(t)$ is the control input.

1.3.2.1 Design of vision-based control schemes

The design of a vision-based control scheme depends on how the task has been defined. For example, the positioning task can be defined directly in the Cartesian space as the regulation of the vector $\mathbf{x}(t)$. Without loss of generality, we can suppose that our objective is to regulate the vector $\mathbf{x}(t)$ to zero. In this case one must be able to estimate the current camera pose from image data. Using an approximation of the parameters $\widehat{\boldsymbol{\eta}}(t)$, one must design a non-linear state observer $\boldsymbol{\psi}$ such that :

$$\widehat{\mathbf{x}}(t) = \boldsymbol{\psi}(\widehat{\boldsymbol{\eta}}(t), \boldsymbol{h}(\boldsymbol{\eta}(t), \mathbf{x}(t))) \tag{1.40}$$

Obviously, in the absence of modeling errors we have $\widehat{\boldsymbol{\eta}}(t) = \boldsymbol{\eta}(t)$ and thus the state can be perfectly estimated $\widehat{\boldsymbol{x}}(t) = \boldsymbol{\psi}(\boldsymbol{\eta}(t), \boldsymbol{h}(\boldsymbol{\eta}(t), \mathbf{x}(t))) = \boldsymbol{x}(t)$. The problem of reconstructing the pose of the camera with respect to the object has been widely studied in the robotic vision community (see Chapter 3) and several solutions have been proposed. For example, if we have a calibrated monocular vision system we need an accurate model of the target. If we have a calibrated stereo system, the model of the target can be estimated on-line by triangulation. The main limitation of this approach is that any modeling error may produce an error in the final pose.

In order to design vision-based control schemes that are more robust to modeling errors, researchers have proposed using a "teaching-by-showing" approach where the reference pose is not given explicitly. Instead, a reference output $\overline{\mathbf{y}}$ is acquired at the

reference pose (again we can suppose without loss of generality that the reference pose corresponds to $\mathbf{x} = 0$) :

$$\overline{\mathbf{y}} = \mathbf{h}(\overline{\boldsymbol{\eta}}, 0) \tag{1.41}$$

where $\overline{\boldsymbol{\eta}}$ contains the parameters at the time of the acquisition. Thus, we can build an error function (not necessarily having the same size as the vector $\mathbf{x}$) from image data only :

$$\boldsymbol{\epsilon}(t) = \boldsymbol{\delta}(\boldsymbol{h}(\boldsymbol{\eta}(t), \mathbf{x}(t)), \boldsymbol{h}(\overline{\boldsymbol{\eta}}, 0)) \tag{1.42}$$

such that if (and only if) $\boldsymbol{\epsilon} = 0$ then $\mathbf{x} = 0$. The design of such an error function is not easy, especially if the parameters $\boldsymbol{\eta}(t)$ vary. Thus, a common assumption is to suppose that imaging conditions do not change $\boldsymbol{\eta}(t) = \overline{\boldsymbol{\eta}}$. Even with such an assumption, the design of the error function is not easy and its choice may greatly influence the behavior of the visual servoing. In particular, the influence of measurement noise can be very different depending on the choice of the error function.

In the teaching-by-showing approach, the choice of the error function cannot be generally decoupled from the design of the control law. Indeed, the same control law can have an unstable behavior with a given control error and a stable behavior with a different control error. Moreover, even if it is possible to compute the control error directly from image data, the design of a stable control law often needs an estimate of the parameters $\overline{\boldsymbol{\eta}}$.

1.3.2.2 Design of vision-based control laws

The design of control laws is a fairly standard problem in robotics when the pose of the camera can be estimated explicitly. However, it is worth noting that for critical systems the design of the control law is more difficult than for holonomic robots. Let me consider here the design of control laws for the teaching-by-showing approach supposing that imaging conditions do not change $\boldsymbol{\eta}(t) = \overline{\boldsymbol{\eta}}$. After taking the time derivative of the control error in equation (1.42) we obtain the following state equations :

$$\begin{aligned} \dot{\boldsymbol{\epsilon}}(t) &= \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x}(t))\, \mathbf{v}(t) \\ \mathbf{y}(t) &= \boldsymbol{h}(\overline{\boldsymbol{\eta}}, \mathbf{x}(t)) \end{aligned} \tag{1.43}$$

where $\mathbf{L}$ is an interaction matrix that generally depends on the parameters $\overline{\boldsymbol{\eta}}$. The problem is to find an appropriate function $\mathbf{k}$ in order to ensure a stable control input :

$$\mathbf{v}(t) = \mathbf{k}(\widehat{\boldsymbol{\eta}}, \mathbf{y}(t), \overline{\mathbf{y}}) = \mathbf{k}(\widehat{\boldsymbol{\eta}}, \boldsymbol{h}(\overline{\boldsymbol{\eta}}, \mathbf{x}(t)), \boldsymbol{h}(\overline{\boldsymbol{\eta}}, 0))$$

such that $\boldsymbol{\epsilon}(t) \to 0$ when $t \to \infty$ starting from an initial pose $\mathbf{x}_0 = \mathbf{x}(0)$. The challenge is to design stable control laws that depend as little as possible on the unknown parameters $\overline{\boldsymbol{\eta}}$. If an estimation $\widehat{\boldsymbol{\eta}}$ is needed, another challenge is to design stable control laws that are robust to errors on the estimation of these parameters.

1.3.2.2.1 Stability and robustness

In ideal conditions (i.e. assuming no modeling and measurement errors) the control law should be at least locally stable. This is the minimal requirement in the design of vision-based control laws. However, it is also important to determine the "size" of the stability domain. A very difficult problem is to design control laws with a large stability domain. The best would be to design globally stable control laws (i.e. the control error convergence to zero whatever the starting position of the robot).

Another problem is the design of robust control laws. A control law can be called "robust" if it is able to perform the assigned stabilization task despite modeling and measurement errors. Determining the "size" of "admissible" errors is important in practice. However, carrying out this type of analysis is usually technically quite difficult. Robustness is needed to ensure that the controlled system will behave as expected. It is an absolute requirement for most applications, not only to guarantee the correct execution of the assigned tasks but also for safety reasons, especially when these tasks involve direct interactions with humans (robotic aided surgery, automatic driving,...).

1.3.2.2.2 Visibility and Continuity

The aim of vision-based control techniques is to control a robot with the feedback coming from visual information. If the visual information gets out of the camera's field of view feedback is no longer possible and the visual servoing must then be stopped. A very important problem is to take into account this constraint in the design of the control law. This visibility problem is amplified when considering critical systems, like non-holonomic robots, due to the restriction on the possible instantaneous motions of the robot.

Another problem related to the visibility appears when the target is partially occluded. In this case, if we suppose that sufficient information is still visible in order to achieve the task, the problem is that the occlusion may perturb the computation of the control error and/or the control law. Depending on how the control scheme has been defined some discontinuities may appear. These discontinuities perturb and decrease the performances of the control law.

Chapitre 2

Numerical analysis

Many parametric estimation problems can be solved by finding the solution of a system of non-linear equations. When the system has more equations than unknowns it is usual to rewrite the problem as a nonlinear least squares optimization which is again solved by finding the solution of another system of non-linear equations. In general, these nonlinear problems do not have an analytical closed-form solution. Thus, an important research subject concerns the study of efficient iterative numerical methods. The objective of this chapter is to describe my contributions in this field.

Numerical methods have been widely studied in the literature. An exhaustive description of them is beyond the scope of this document. In this chapter, the focus is on numerical methods that make use of derivatives of the nonlinear functions involved in the problems. Indeed, such methods have a strong link with automatic control methods. The reader may refer to well-known textbooks like [Isaacson 66], [Dennis 83], [Quarteroni 00] which describe in detail most of the standard numerical methods. Besides the standard methods, I will consider methods that use additional information on the derivatives at the solution. In several robotic vision applications, such as for example parametric identification or vision-based control, it is possible to measure or approximate this additional knowledge. In this context, one important contribution has been to propose the Efficient Second-order approximation Method (ESM) which has several advantages with respect to standard methods. The ESM has been successfully applied to vision-based estimation (see Chapter 3) and to vision-based robot control (see Chapter 4).

The chapter is organized as follows. First the problem of finding the solution of one nonlinear equation in one unknown is considered. The reason for studying the one-variable problem separately is that it allows us to more easily understand the principles of the different methods that have been proposed in the literature. For example, geometric interpretations of the methods can be plotted [Isaacson 66]. The extension to the multidimensional case (i.e. the solution of systems of nonlinear equations) is almost straightforward. Then, the generalization of the iterative methods to nonlinear systems defined on Lie groups is introduced. Finally, I will consider the optimization of a nonlinear least squares problem and its modification to increase robustness with respect to aberrant measurements.

2.1 Iterative solution of non-linear equations

In this section, I consider the problem of iteratively finding a root of the equation :

$$f(x) = 0 \tag{2.1}$$

given the function $f : \mathcal{D} = (a, b) \subseteq \mathbb{R} \mapsto \mathbb{R}$ and starting from an initial approximation $\widehat{x}_0$ of a root $\overline{x}$. The problem of how to find the initial approximation $\widehat{x}_0$ is not easy to solve but is very important since we will see that most of the iterative method will not work if $\widehat{x}_0$ is chosen too far from $\overline{x}$. For simplicity, let me suppose that the function $f(x) \in C^\infty$ so that it can be expanded using a Taylor series about $\widehat{x}$:

$$f(\widehat{x} + x) = f(\widehat{x}) + g(\widehat{x})\, x + \frac{1}{2} h(\widehat{x})\, x^2 + \frac{1}{6} q(x^*)\, x^3 \tag{2.2}$$

where the last term is a third-order Lagrange remainder and $x^* \in (\widehat{x}, x)$ and the smooth functions $g(x)$, $h(x)$, and $q(x)$ are defined as follows :

$$g(x) = \frac{df(x)}{dx} \tag{2.3}$$

$$h(x) = \frac{d^2 f(x)}{dx^2} \tag{2.4}$$

$$q(x) = \frac{d^3 f(x)}{dx^3} \tag{2.5}$$

Suppose that $\overline{x}$ is a simple root of the equation (2.1), therefore :

$$f(\overline{x}) = 0 \tag{2.6}$$

$$g(\overline{x}) \neq 0 \tag{2.7}$$

Starting from the initial approximation $\widehat{x}_0$, the root-finding problem can be solved by iteratively finding an increment $\widetilde{x}$ in order to generate a sequence of values

$$\widehat{x}_{k+1} = \widehat{x}_k + \widetilde{x}_k \tag{2.8}$$

such that :

$$\lim_{k \to \infty} \widehat{x}_k = \overline{x} \tag{2.9}$$

Two important questions should be answered : firstly, how far from the true root $\overline{x}$ can we chose the initial approximation $\widehat{x}_0$ (convergence domain) ? secondly, how fast the sequence of values $\{\widehat{x}_k\} = \{\widehat{x}_0, \widehat{x}_1, \widehat{x}_2, \ldots\}$ does converge to the true root (convergence rate). In order to answer these two questions let me rewrite the root finding problem in a more general form as proposed by [Isaacson 66].

2.1.1 Functional iteration methods for root finding

If we define :

$$\boldsymbol{\varphi}(x) = x - k(x)\, f(x) \tag{2.10}$$

where $0 < |k(x)| < \infty, \quad \forall x \in \mathcal{D}$, then any equation $f(x) = 0$ can be rewritten in the following form :

$$x = \boldsymbol{\varphi}(x) \tag{2.11}$$

As pointed out by [Isaacson 66], most of the iterative methods can be written in the form of a functional iteration method (also known as the Picard iteration method) :

$$\widehat{x}_{k+1} = \boldsymbol{\varphi}(\widehat{x}_k) \tag{2.12}$$

Starting from an initial approximation $\widehat{x}_0$ of $\overline{x}$ the convergence of such an iteration process depends on the behavior of the function $\boldsymbol{\varphi}$. In this section, I will show under which conditions there exists a unique fixed point $\overline{x}$ satisfying :

$$\overline{x} = \boldsymbol{\varphi}(\overline{x}) \tag{2.13}$$

which obviously implies that $f(\overline{x}) = 0$ since $k(\overline{x}) \neq 0$. First of all, let me give the definitions of the domain of convergence and the order of convergence of a functional iteration method.

2.1.1.1 Definition of the convergence domain

There are several possible definitions of the convergence domain of a functional iteration method. The more general definition is the following :

Definition 1 (Domain of Convergence) *The convergence domain $\mathcal{C}$ of the functional iteration in equation (2.12) is defined as the set of the starting points $\{\widehat{x}_0\} \in \mathcal{C} \subseteq \mathcal{D}$ (obviously containing the root itself) for which the iteration process converges to the root.*

The problem is that finding such a domain is extremely difficult in general since it highly depends to a large extent on the shape of the function $\boldsymbol{\varphi}$. In order to obtain more generic results one must restrict the definition of the convergence domain :

Definition 2 (Monotone Domain of Convergence) *The monotone domain of convergence $\mathcal{C}$ of the functional iteration in equation (2.12) is the set of the starting points $\{\widehat{x}_0\} \in \mathcal{C} \subseteq \mathcal{D}$ such that the error is always decreasing $|\widehat{x}_{k+1} - \overline{x}| < |\widehat{x}_k - \overline{x}|$.*

This second definition is more restrictive since starting points may exist for which the error initially increases and then converges. However, such a monotone convergence domain is often easier to obtain. For example, Theorem 1 gives simple sufficient conditions on the function $\boldsymbol{\varphi}(x)$ that assure the convergence of a functional iteration method to a unique root in a monotone convergence domain. The corollary to Theorem 1 provides sufficient conditions on the first derivative $\boldsymbol{\varphi}^1(x)$ which are sometimes easier to compute. However, the true monotone convergence domain is often larger than the one defined by the conditions of the corollary.

Theorem 1 *If $x = \boldsymbol{\varphi}(x)$ has a root at $x = \overline{x}$ and*

$$|\boldsymbol{\varphi}(x) - \boldsymbol{\varphi}(\overline{x})| \leq \lambda |x - \overline{x}| \tag{2.14}$$

for all x in the domain of convergence defined by

$$\mathcal{C} = |x - \overline{x}| < \delta \tag{2.15}$$

Then, for any initial estimate $\widehat{x}_0 \in \mathcal{C}$:

i) all the iterates x_k defined by equation (2.12) lie within the domain of convergence $\mathcal{C}$:

$$\overline{x} - \delta < \widehat{x}_k < \overline{x} + \delta$$

ii) the iterates converge to the fixed point $\overline{x}$

$$\lim_{k \to \infty} \widehat{x}_k = \overline{x}$$

iii) $\overline{x}$ is the only root in the domain $\mathcal{C}$.

Proof: Let me start by proving *i)*. From equation (2.13) we have $\boldsymbol{\varphi}(\overline{x}) = \overline{x}$. Setting $x = \widehat{x}_k$, using equation (2.13) the condition in equation (2.14) can be written

$$|x_{k+1} - \overline{x}| \leq \lambda |x_k - \overline{x}| \tag{2.16}$$

which means that the error is decreasing monotonically $|\widetilde{x}_{k+1}| \leq |\widetilde{x}_k|$ since $\lambda < 1$. To prove *ii)* it is sufficient to show that $\{\widehat{x}\}$ is a Cauchy sequence. From equation (2.16) we get :

$$|x_{k+1} - \overline{x}| \leq \lambda |x_k - \overline{x}| \leq \lambda^2 |x_{k-1} - \overline{x}| \leq \cdots \leq \lambda^{k+1} |x_0 - \overline{x}| \tag{2.17}$$

By letting $k \to \infty$ since $\lambda < 1$ then

$$\lim_{k \to \infty} \widetilde{x}_k = 0$$

and :

$$\lim_{k \to \infty} \widehat{x}_k = \overline{x}$$

Finally, to prove *iii)* let me suppose that there exists $\overline{x}' \in \mathcal{D}$ such that $\overline{x}' \neq \overline{x}$ and $\boldsymbol{\varphi}(\overline{x}') = \overline{x}'$. Then,

$$|\overline{x}' - \overline{x}| = |\boldsymbol{\varphi}(\overline{x}') - \boldsymbol{\varphi}(\overline{x})| \leq \lambda |\overline{x}' - \overline{x}| < |\overline{x}' - \overline{x}| \tag{2.18}$$

This contradiction implies that $\overline{x}' = \overline{x}$ which is impossible by assumption.

Corollary 1 *If we replace the assumption (2.14) with*

$$|\boldsymbol{\varphi}^1(x)| \leq \lambda < 1 \tag{2.19}$$

for $x \in \mathcal{C}$, then the Theorem 1 still holds.

Proof: From the mean value theorem we get :

$$\boldsymbol{\varphi}(x) - \boldsymbol{\varphi}(\overline{x}) = \boldsymbol{\varphi}^1(x^*)(x - \overline{x})$$

for some $x^* \in (\overline{x}, x)$. Thus, from equation (2.19) we obtain the assumption (2.14)

$$|\boldsymbol{\varphi}(x) - \boldsymbol{\varphi}(\overline{x})| \leq |\boldsymbol{\varphi}^1(x^*)||x - \overline{x}| \leq \lambda |x - \overline{x}|$$

where $\lambda < 1$.

2.1.1.2 Definition of the order of convergence

In numerical analysis, the speed at which a convergent sequence approaches its limit is called the order of convergence and it is defined as follows :

Definition 3 (Order of Convergence) *Assume that the sequence* $\{\widehat{x}_k\} = \{\widehat{x}_0, \widehat{x}_1, \cdots\}$ *converges to* $\overline{x}$*, and set* $\widetilde{x}_k = \widehat{x}_k - \overline{x}$*. If two positive constants* r *and* s *exist, and*

$$\lim_{k\to\infty} \frac{|\widehat{x}_k - \overline{x}|}{|\widehat{x}_{k-1} - \overline{x}|^r} = s$$

then the sequence is said to converge to $\overline{x}$ *with order of convergence* r*. The number* s *is called the asymptotic error constant. If* $r = 1$*, the convergence of* $\{\widehat{x}_k\}$ *is called linear. If* $r = 2$*, the convergence of* $\{\widehat{x}_k\}$ *is called quadratic. If* $r = 3$*, the convergence of* $\{\widehat{x}_k\}$ *is called cubic, and so on.*

The concept of order of convergence is important since it is related to the number of iterations that an iterative method needs to converge. A higher order of convergence is obviously preferable. However, in order to compare the convergence rate of two different methods one must also take into account the computational complexity for each iteration. The following theorem gives simple conditions to determine the order of convergence of a functional iteration method.

Theorem 2 *If in addition to the assumptions of Theorem 1 :*

$$\boldsymbol{\varphi}^1(\overline{x}) = \boldsymbol{\varphi}^2(\overline{x}) = \ldots = \boldsymbol{\varphi}^{r-1}(\overline{x}) = 0 \tag{2.20}$$

$$\boldsymbol{\varphi}^r(\overline{x}) \neq 0 \tag{2.21}$$

and if $\forall x \in \mathcal{C}$ *:*

$$|\boldsymbol{\varphi}^r(x)| \leq r!\, M \tag{2.22}$$

Then the sequence $\{\widetilde{x}_k\}$ *converges to* $\overline{x}$ *with order of convergence* r*.*

Proof: The convergence of the iterates to the fixed point $\overline{x}$ is assured by the assumptions of Theorem 1. Consider the Taylor series of $\boldsymbol{\varphi}(x)$ about $\overline{x}$ is :

$$\boldsymbol{\varphi}(x) = \boldsymbol{\varphi}(\overline{x}) + \boldsymbol{\varphi}^1(\overline{x}) + \frac{1}{2!}\boldsymbol{\varphi}^2(\overline{x}) + \ldots + \frac{1}{r!}\boldsymbol{\varphi}^r(x^*)(x - \overline{x})^r$$

where $x^* \in (x, \overline{x})$. Using equation (2.20) the Taylor series can also be written as follows :

$$\boldsymbol{\varphi}(x) = \boldsymbol{\varphi}(\overline{x}) + \frac{1}{r!}\boldsymbol{\varphi}^r(x^*)(x - \overline{x})^r$$

Using equation (2.13) and equation (2.22) the error at iteration k can be bounded as follows :

$$|\widehat{x}_k - \overline{x}| = |\boldsymbol{\varphi}(\widehat{x}_{k-1}) - \boldsymbol{\varphi}(\overline{x})| = \frac{1}{r!}|\boldsymbol{\varphi}^r(x^*_{k-1})||\widehat{x}_{k-1} - \overline{x}|^r \leq M|\widehat{x}_{k-1} - \overline{x}|^r \tag{2.23}$$

Then, there exists a scalar $s \leq M$ such that

$$\lim_{k\to\infty} \frac{|\widehat{x}_k - \overline{x}|}{|\widehat{x}_{k-1} - \overline{x}|^r} = s$$

2.1.2 Standard iterative methods

Several root finding methods have been proposed in the literature (see for example [Quarteroni 00]). In this section, I will describe in detail only the Newton-Raphson and Halley methods. Despite the fact that the Newton-Raphson method is the most well known and used method its description is unavoidable since it can be considered as a reference method to which new ones can be compared. On the other hand, the Halley method is less well known but its description is useful to better understand the compromise between order of convergence and efficiency. The Halley method is also closely related to the ESM method that will be propose in the next section.

2.1.2.1 The Newton-Raphson method

The Newton-Raphson method, also simply called the Newton method, is a root-finding algorithm that was independently discovered by both sir Isaac Newton and Joseph Raphson. The Newton-Raphson method uses the first-order terms of the Taylor series of the function $f(x)$ computed at $\widehat{x} \approx \overline{x}$. Indeed, keeping terms of equation (2.2) only to first-order we obtain a first-order approximation of the function :

$$f(\widehat{x} + x) \approx f(\widehat{x}) + g(\widehat{x})\, x \tag{2.24}$$

Evaluating $f(\widehat{x} + x)$ at $\widetilde{x} = \overline{x} - \widehat{x}$ we obtain :

$$f(\overline{x}) = f(\widehat{x} + \widetilde{x}) \approx f(\widehat{x}) + g(\widehat{x})\, \widetilde{x} = 0 \tag{2.25}$$

Supposing that $g(\widehat{x}) \neq 0$ we can solve equation (2.24) and compute the Newton-Raphson increment $\widetilde{x}$:

$$\widetilde{x} = -\frac{f(\widehat{x})}{g(\widehat{x})} \tag{2.26}$$

and the estimation is updated as follows : $\widehat{x}_{k+1} = \widehat{x}_k + \widetilde{x}_k$. For the Newton method, $k(x) = 1/g(x)$ and the function $\boldsymbol{\varphi}(x)$ is defined as :

$$\boldsymbol{\varphi}(x) = x - \frac{f(x)}{g(x)} \tag{2.27}$$

2.1.2.1.1 Convergence domain of the Newton-Raphson method

Theoretically, the Newton-Raphson method cannot be computed when $g(x) = 0$. Since we have supposed that $g(\overline{x}) \neq 0$ and that $g(x)$ is smooth, the convergence domain of the Newton-Raphson method is a connected domain which includes the root $\overline{x}$.

$$\mathcal{C} \subset (x_{min}, x_{max}) \tag{2.28}$$

such that :

$$g(x_{min}) = g(x_{max}) = 0$$

The convergence domain is included in these bounds but is generally much smaller. From Theorem 1 the monotone domain of convergence is defined by :

$$\left| x - \overline{x} - \frac{f(x)}{g(x)} \right| \leq |x - \overline{x}|$$

From corollary 1 a more restricted monotone domain of convergence is defined by the inequality :

$$|f(x)h(x)| < g(x)^2$$

2.1.2.1.2 Order of convergence of the Newton-Raphson method

If we apply Theorem 2 to the function defined in the equation (2.27) it is easy to verify that the Newton-Raphson method has at least quadratic order of convergence. Indeed, the first derivative of the function is :

$$\boldsymbol{\varphi}^1(x) = \frac{h(x)}{g(x)^2} f(x) \tag{2.29}$$

When computed at $\overline{x}$ we obtain :

$$\boldsymbol{\varphi}^1(\overline{x}) = 0 \tag{2.30}$$

2.1.2.2 The Halley method

The Halley method was proposed by the mathematician and astronomer Edmond Halley. Keeping terms of equation (2.2) only to second-order we obtain a second-order approximation of the function :

$$f(\widehat{x} + x) \approx f(\widehat{x}) + g(\widehat{x})\, x + \frac{1}{2} h(\widehat{x})\, x^2 \tag{2.31}$$

As for the Newton-Raphson method, setting $x = \widetilde{x}$ then $f(\widehat{x} + \widetilde{x}) = 0$ and solving the following quadratic equation we obtain the *irrational* Halley method :

$$f(\widehat{x}) + g(\widehat{x})\, \widetilde{x} + \frac{1}{2} h(\widehat{x})\, \widetilde{x}^2 = 0 \tag{2.32}$$

However, the solution of the quadratic equation involves a square root evaluation. Moreover, its extension to the multidimensional case involves the solution of a system of n quadratic equations in n unknowns which in general has not a closed-form solution and it may not have a solution at all. For this reason, the only Halley method that I will consider in this document is the *rational* Halley method that can be obtained by rewriting the equation :

$$f(\widehat{x}) + (g(\widehat{x}) + \frac{1}{2} h(\widehat{x})\, \widetilde{x})\widetilde{x} = 0 \tag{2.33}$$

Plugging the Newton-Raphson iteration

$$\widetilde{x} = -\frac{f(\widehat{x})}{g(\widehat{x})}$$

only into the expression in the brackets one obtains :

$$f(\widehat{x}) + \left(g(\widehat{x}) - \frac{1}{2} h(\widehat{x}) \, \frac{f(\widehat{x})}{g(\widehat{x})} \right) \widetilde{x} = 0 \tag{2.34}$$

and solving this equation one obtains the *rational* Halley increment :

$$\widetilde{x} = -\frac{2 g(\widehat{x}) f(\widehat{x})}{2\, g(\widehat{x})^2 - f(\widehat{x}) h(\widehat{x})} \tag{2.35}$$

for the Halley method $k(x) = 2g(x)/(2\, g(x)^2 - f(x)h(x))$ and the function $\boldsymbol{\varphi}(x)$ is defined as :

$$\varphi(x) = x - \frac{2 g(x) f(x)}{2\, g(x)^2 - f(x) h(x)} \tag{2.36}$$

2.1.2.2.1 Convergence domain of the Halley method

The convergence domain of the Halley method is at least as big as the convergence domain of the Newton method. However, it could be smaller since the Halley method cannot be computed either when $2\, g(x)^2 = f(x)h(x)$.

2.1.2.2.2 Order of convergence of the Halley method

If we apply Theorem 2 to the function defined in the equation (2.36) it is easy to verify that the Halley method has at least cubic order of convergence. Indeed, the derivatives of the function are :

$$\begin{aligned} \boldsymbol{\varphi}^1(x) &= \alpha(x) f(x)^2 & (2.37) \\ \boldsymbol{\varphi}^2(x) &= (\alpha^1(x) f(x) + 2\alpha(x)) f(x) & (2.38) \end{aligned}$$

where

$$\alpha(x) = \frac{3h(x)^2 - 2g(x)q(x)}{(2g(x)^2 - f(x)h(x))^2}$$

When computed at $\overline{x}$ we obtain :

$$\begin{aligned} \boldsymbol{\varphi}^1(\overline{x}) &= 0 & (2.39) \\ \boldsymbol{\varphi}^2(\overline{x}) &= 0 & (2.40) \end{aligned}$$

The Halley method has cubic order of convergence. So why is it not the preferred root finding method ? This is because a high order of convergence alone is not enough. It should be weighted by the computational complexity per iteration. For example, if the computation of the Halley iteration costs twice the computation of the Newton iteration then we are able to compute two Newton iterations at a time. Thus, we obtain a method that has quartic order of convergence.

2.1.3 Iterative methods with known gradient at the solution

Suppose that we can measure the gradient $g(x)$ of the function $f(x)$ at the solution (i.e. $g(\overline{x})$ can be measured). Can we use this additional information to improve the efficiency of root finding algorithms ? For example, using $g(\overline{x})$ instead of $g(x)$ in the standard Newton method leads to an algorithm which has the same order of convergence while being more efficient. For this reason I will call it the Efficient Newton method. The Efficient Newton method has been applied in vision-based robot control by [Espiau 92] and for image registration by [Baker 04]. I will show that we can use the information on $g(\overline{x})$ differently and we can design an algorithm called ESM (see [Malis 04a], [Benhimane 04]) which has the same order of convergence as the Halley method while being more efficient since it has the same computational complexity per iteration of the standard Newton method.

2.1.3.1 The efficient Newton method

The efficient Newton method can be obtained by considering the Taylor series of $f(x)$ about $\overline{x}$:

$$f(\overline{x}+x) = f(\overline{x}) + g(\overline{x})x + \frac{1}{2}h(x^*)x^2 \tag{2.41}$$

Since $f(\overline{x}) = 0$, keeping terms up to first order we obtain :

$$f(\overline{x}+x) \approx g(\overline{x})x \tag{2.42}$$

Evaluating the equation at $x = -\widetilde{x} = \widehat{x} - \overline{x}$ gives :

$$f(\widehat{x}) \approx -g(\overline{x})\widetilde{x}$$

We obtain an efficient Newton method by computing :

$$\widetilde{x} = -\frac{f(\widehat{x})}{g(\overline{x})}$$

It is efficient since the inverse of the gradient is computed once and for all. For the efficient Newton method $k(x) = 1/g(\overline{x})$ and the function $\boldsymbol{\varphi}(x)$ is defined as :

$$\boldsymbol{\varphi}(x) = x - \frac{f(x)}{g(\overline{x})} \tag{2.43}$$

2.1.3.1.1 Convergence domain of the Efficient Newton method

The convergence domain of the Efficient Newton method could be bigger than the convergence domain of the Newton and Halley methods. Indeed, since I supposed $g(\overline{x}) \neq 0$ the Efficient Newton iteration can always be computed.

Theorem 3 *If $0 < g(x) \leq g(\overline{x})$ then the monotone convergence domain of the Newton-Raphson method is included in the monotone convergence domain of the Efficient Newton method. This means that if the Newton-Raphson method converges then the Efficient Newton method also converges.*

Proof: The monotone convergence domain of the Newton-Raphson method is defined by :

$$|\varphi_{NRM}(x) - \overline{x}| < \lambda |x - \overline{x}| \tag{2.44}$$

where :

$$\varphi_{NRM}(x) = x + \widetilde{x}_{NRM} = x - \frac{f(x)}{g(x)}$$

while the convergence domain of the Efficient Newton method is defined by :

$$|\varphi_{ENM}(x) - \overline{x}| < \lambda |x - \overline{x}| \tag{2.45}$$

where :

$$\varphi_{ENM}(x) = x + \widetilde{x}_{ENM} = x - \frac{f(x)}{g(\overline{x})}$$

The Efficient Newton increment can be rewritten as a function of the Newton increment :

$$\widetilde{x}_{ENM} = -\frac{g(x)}{g(\overline{x})} \frac{f(x)}{g(x)} = \gamma(x)\, \widetilde{x}_{NRM}$$

where :

$$\gamma(x) = \frac{g(x)}{g(\overline{x})}$$

If $0 < g(x) \leq g(\overline{x})$ then $0 < \gamma(x) \leq 1$ and

$$|\varphi_{ENM}(x) - \overline{x}| \leq |\varphi_{NRM}(x) - \overline{x}| < \lambda |x - \overline{x}| \tag{2.46}$$

2.1.3.1.2 Order of convergence of the Efficient Newton method

The following theorem shows that the efficient Newton method has the same order of convergence as the standard Newton-Raphson method.

Theorem 4 *The efficient Newton method has at least quadratic order of convergence.*

Proof: Simply apply Theorem 2 to the function $\boldsymbol{\varphi}(x)$ defined in equation (2.43). The derivative of the function is :

$$\boldsymbol{\varphi}^1(x) = 1 - \frac{g(x)}{g(\overline{x})} \tag{2.47}$$

When computed at $\overline{x}$ we obtain :

$$\boldsymbol{\varphi}^1(\overline{x}) = 0 \tag{2.48}$$

which proves that the efficient Newton method has at least quadratic order of convergence.

2.1.3.2 The Efficient Second-order approximation Method

Like the Efficient Newton method, the Efficient Second-order approximation Method (ESM) assumes that $g(\overline{x})$ can be measured. Like the Newton-Raphson method, it is a root finding method. However, like the Halley method, it uses the first two terms of the Taylor series.

The ESM uses the Taylor series of $g(x)$ about $\widehat{x}$ which can be written :

$$g(\widehat{x}+x) = g(\widehat{x}) + h(\widehat{x})\,x + \frac{1}{2}q(x^*)\,x^2 \tag{2.49}$$

where the last term is a second-order Lagrange remainder. Then, we can compute :

$$h(\widehat{x})\,x = g(\widehat{x}+x) - g(\widehat{x}) - \frac{1}{2}q(x^*)\,x^2 \tag{2.50}$$

Plugging equation (2.50) into equation (2.2) we obtain an expression of the function $f(x)$ without the second-order terms :

$$f(\widehat{x}+x) = f(\widehat{x}) + \frac{1}{2}\left(g(\widehat{x}) + g(\widehat{x}+x)\right)\,x - \frac{1}{12}q(x^*)\,x^3 \tag{2.51}$$

Keeping the terms of this equation only to second-order we obtain an efficient second-order approximation of the function :

$$f(\widehat{x}+x) \approx f(\widehat{x}) + \frac{1}{2}\left(g(\widehat{x}) + g(\widehat{x}+x)\right)\,x \tag{2.52}$$

When compared to the second-order approximation (2.31) used in the Halley method, it is evident that the second-order approximation (2.52) is more efficient since it is obtained without computing the second derivatives of the function. Thus, the ESM can also be viewed as an efficient version of the Halley method where instead of plugging the Newton-Raphson iteration into the second order approximation (2.31) we plug equation (2.50). Setting $x = \widetilde{x}$ and $f(\widehat{x}+\widetilde{x}) = 0$ one can solve the following linear equation :

$$f(\widehat{x}) + \frac{1}{2}\left(g(\widehat{x}) + g(\overline{x})\right)\,\widetilde{x} = 0 \tag{2.53}$$

Supposing that $g(\widehat{x}) + g(\overline{x}) \neq 0$ we find the ESM iteration is :

$$\widetilde{x} = -\,\frac{2\,f(\widehat{x})}{g(\widehat{x}) + g(\overline{x})} \tag{2.54}$$

For the ESM $k(x) = 2/(g(x) + g(\overline{x}))$ and the function $\boldsymbol{\varphi}(x)$ is defined as :

$$\varphi(x) = x - \frac{2\,f(x)}{g(x) + g(\overline{x})} \tag{2.55}$$

2.1.3.2.1 Convergence domain of the ESM

The following theorem shows that the bounds on the convergence domain of the ESM are wider than the bounds on the convergence domains of the Newton-Raphson and Halley methods.

Theorem 5 *The bounds of the convergence domains of the Newton-Raphson and Halley methods are included in the bounds of the convergence domain of the ESM.*

Proof: The bounds for the Newton-Raphson and Halley methods are defined by $g(x) = 0$. On the other hand, the ESM method cannot be computed when $g(x) = -g(\overline{x})$ which determines the bounds of the convergence domain. This means that the sign of $g(x)$ must be opposite to the sign of $g(\overline{x})$. Since $g(x)$ is smooth, it must become null before changing its sign. This means that if $g(x_1) = 0$ and $g(x_2) = -g(\overline{x})$ then $|x_1 - \overline{x}| < |x_2 - \overline{x}|$ which proves that the bounds of the convergence domain of the ESM are wider.

Theorem 6 *If $0 < g(x) \leq g(\overline{x})$ then the monotone convergence domain of the Newton-Raphson method is included in the monotone convergence domain of the ESM. This means that if the Newton-Raphson method converges then the ESM also converges.*

Proof: The monotone convergence domain of the Newton-Raphson method is defined by :

$$|\varphi_{NRM}(x) - \overline{x}| < \lambda |x - \overline{x}| \tag{2.56}$$

where :

$$\varphi_{NRM}(x) = x + \widetilde{x}_{NRM} = x - \frac{f(x)}{g(x)}$$

while the convergence domain of the ESM is defined by :

$$|\varphi_{ESM}(x) - \overline{x}| < \lambda |x - \overline{x}| \tag{2.57}$$

where :

$$\varphi_{ESM}(x) = x + \widetilde{x}_{ESM} = x - \frac{2f(x)}{(g(x) + g(\overline{x}))}$$

The ESM increment can be rewritten as a function of the Newton increment :

$$\widetilde{x}_{ESM} = -\frac{2g(x)}{(g(x) + g(\overline{x}))} \frac{f(x)}{g(x)} = \gamma(x)\, \widetilde{x}_{NRM}$$

where :

$$\gamma(x) = \frac{2g(x)}{(g(x) + g(\overline{x}))}$$

If $0 < g(x) \leq g(\overline{x})$ then $0 < \gamma(x) \leq 1$ and

$$|\varphi_{ESM}(x) - \overline{x}| \leq |\varphi_{NRM}(x) - \overline{x}| < \lambda |x - \overline{x}| \tag{2.58}$$

2.1.3.2.2 Order of convergence of the ESM

The following theorem shows that the ESM has the same order of convergence as the Halley method which is also higher than the Newton method.

Theorem 7 *The Efficient Second-order approximation Method has at least cubic order of convergence.*

Proof: Simply apply Theorem 2 to the function $\boldsymbol{\varphi}(x)$ defined in equation (2.36). The derivatives of the function are :

$$\boldsymbol{\varphi}^1(x) = \frac{g(\overline{x}) - g(x)}{g(x) + g(\overline{x})} + \frac{2\,h(x)}{(g(x) + g(\overline{x}))^2} f(x) \tag{2.59}$$

$$\boldsymbol{\varphi}^2(x) = 2\,h(x)\frac{g(x) - g(\overline{x})}{(g(x) + g(\overline{x}))^2} + \frac{2q(x)(g(x) + g(\overline{x})) - 4h(x)^2}{(g(x) + g(\overline{x}))^3} f(x) \tag{2.60}$$

When computed at $\overline{x}$ we obtain :

$$\boldsymbol{\varphi}^1(\overline{x}) = 0 \tag{2.61}$$

$$\boldsymbol{\varphi}^2(\overline{x}) = 0 \tag{2.62}$$

This theorem is important since the computation cost of the ESM is almost the same as the computation cost of the Newton method and much less than the computation cost of the Halley method. This will be even more true for multidimensional systems where the inverse of the gradient become the inverse of a matrix.

2.2 Iterative solution of nonlinear systems

The extension of the methods presented in the previous section to the iterative solution of systems of nonlinear equations (i.e. n nonlinear equations with n unknowns) is almost straightforward and it is mainly a matter of notations The theorems proved in the previous section can also be extended to the multidimensional case as in [Isaacson 66].

2.2.1 Extension to systems of nonlinear equations

Consider the following system of m equations in n unknowns :

$$\mathbf{f}(\mathbf{x}) = 0 \tag{2.63}$$

The Taylor series of the vector function $\mathbf{f}(\widehat{\mathbf{x}} + \mathbf{x})$ about $\widehat{\mathbf{x}}$ can be written :

$$\mathbf{f}(\widehat{\mathbf{x}} + \mathbf{x}) = \mathbf{f}(\widehat{\mathbf{x}}) + \mathbf{J}(\widehat{\mathbf{x}})\,\mathbf{x} + \frac{1}{2!}\mathbf{H}(\widehat{\mathbf{x}}, \mathbf{x})\,\mathbf{x} + \frac{1}{3!}\mathbf{Q}(\mathbf{x}^*, \mathbf{x}^2)\,\mathbf{x} \tag{2.64}$$

where the last term is a third-order Lagrange remainder and $\mathbf{x}^* \in (\widehat{\mathbf{x}}, \mathbf{x})$ and the matrices $\mathbf{J}(\mathbf{x})$, $\mathbf{H}(\mathbf{x})$, and $\mathbf{Q}(\mathbf{x})$ are defined as follows :

$$\begin{aligned} \mathbf{J}(\mathbf{x}) &= \nabla \mathbf{f}(\mathbf{x}) && (2.65) \\ \mathbf{H}(\widehat{\mathbf{x}}, \mathbf{x}) &= \nabla \mathbf{J}(\mathbf{x})\widehat{\mathbf{x}} && (2.66) \\ \mathbf{Q}(\widehat{\mathbf{x}}, \mathbf{x}) &= \nabla \mathbf{H}(\widehat{\mathbf{x}}, \mathbf{x})\widehat{\mathbf{x}} && (2.67) \end{aligned}$$

Suppose that $\overline{\mathbf{x}}$ is a simple solution of the system (2.63), therefore :

$$\begin{aligned} \mathbf{f}(\overline{\mathbf{x}}) &= 0 && (2.68) \\ \det(\mathbf{J}(\overline{\mathbf{x}})) &\neq 0 && (2.69) \end{aligned}$$

Starting from the initial approximation $\widehat{\mathbf{x}}_0$, the problem can be solved by iteratively finding an increment $\widetilde{\mathbf{x}}$ in order to generate a sequence of values

$$\widehat{\mathbf{x}}_{k+1} = \widehat{\mathbf{x}}_k + \widetilde{\mathbf{x}}_k \tag{2.70}$$

such that :

$$\lim_{k \to \infty} \widehat{\mathbf{x}}_k = \overline{\mathbf{x}} \tag{2.71}$$

Similarly to the uni-dimensional case we can define a vector function

$$\boldsymbol{\varphi}(\mathbf{x}) = \mathbf{x} - \mathbf{K}(\mathbf{x})\mathbf{f}(\mathbf{x})$$

where $\mathbf{K}(\mathbf{x})$ is a full rank (n×n) matrix (i.e. $\det(\mathbf{K}) \neq 0$. We obtain the multidimensional equivalent of a functional iteration by solving the problem :

$$\mathbf{x} = \boldsymbol{\varphi}(\mathbf{x}) \tag{2.72}$$

Again, if $\overline{\mathbf{x}}$ is a solution of

$$\overline{\mathbf{x}} = \boldsymbol{\varphi}(\overline{\mathbf{x}})$$

then $\mathbf{f}(\overline{\mathbf{x}}) = 0$ since $\mathbf{K}(\overline{\mathbf{x}})$ is a full rank matrix.

2.2.1.1 The multidimensional Newton method

Keeping terms of equation (2.64) only to first order we obtain :

$$\mathbf{f}(\widehat{\mathbf{x}} + \mathbf{x}) \approx \mathbf{f}(\widehat{\mathbf{x}}) + \mathbf{J}(\widehat{\mathbf{x}})\,\mathbf{x} \tag{2.73}$$

This expression can be used to estimate the amount of offset $\widetilde{\mathbf{x}}$ needed to land closer to the root starting from an initial guess $\widehat{x}$. Setting $\mathbf{x} = \widetilde{\mathbf{x}} = \overline{\mathbf{x}} - \widehat{\mathbf{x}}$ we have $\mathbf{f}(\widehat{\mathbf{x}} + \widetilde{\mathbf{x}}) = \mathbf{f}(\overline{\mathbf{x}}) = 0$ and :

$$\mathbf{f}(\widehat{\mathbf{x}} + \widetilde{\mathbf{x}}) \approx \mathbf{f}(\widehat{\mathbf{x}}) + \mathbf{J}(\widehat{\mathbf{x}})\,\widetilde{\mathbf{x}} = 0 \tag{2.74}$$

Supposing $\det(\mathbf{J}(\widehat{\mathbf{x}})) \neq 0$ we can solve equation (2.74) and obtain the multidimensional Newton-Raphson increment :

$$\widetilde{\mathbf{x}} = -\mathbf{J}(\widehat{\mathbf{x}})^{-1}\mathbf{f}(\widehat{\mathbf{x}}) \tag{2.75}$$

The multidimensional Newton-Raphson also has at least a quadratic order of convergence.

2.2.1.2 The multidimensional Halley method

Keeping terms of equation (2.64) only to second order we obtain :

$$\mathbf{f}(\widehat{\mathbf{x}}+\mathbf{x}) \approx \mathbf{f}(\widehat{\mathbf{x}}) + \mathbf{J}(\widehat{\mathbf{x}})\,\mathbf{x} + \frac{1}{2!}\mathbf{H}(\widehat{\mathbf{x}},\mathbf{x})\,\mathbf{x} \tag{2.76}$$

Setting $\mathbf{x} = \widetilde{\mathbf{x}} = \overline{\mathbf{x}} - \widehat{\mathbf{x}}$ we have $\mathbf{f}(\widehat{\mathbf{x}}+\widetilde{\mathbf{x}}) = \mathbf{f}(\overline{\mathbf{x}}) = 0$ and :

$$\mathbf{f}(\widehat{\mathbf{x}}+\widetilde{\mathbf{x}}) \approx \mathbf{f}(\widehat{\mathbf{x}}) + \mathbf{J}(\widehat{\mathbf{x}})\,\widetilde{\mathbf{x}} + \frac{1}{2!}\mathbf{H}(\widehat{\mathbf{x}},\widetilde{\mathbf{x}})\,\widetilde{\mathbf{x}} = 0 \tag{2.77}$$

Plugging the Newton iterate :

$$\widetilde{\mathbf{x}} = -\mathbf{J}(\widehat{\mathbf{x}})^{-1}\mathbf{f}(\widehat{\mathbf{x}})$$

into $\mathbf{H}(\widehat{\mathbf{x}},\widetilde{\mathbf{x}})$ of equation (2.77) we obtain the following linear equation :

$$\mathbf{f}(\widehat{\mathbf{x}}+\mathbf{x}) \approx \mathbf{f}(\widehat{\mathbf{x}}) + \frac{1}{2}(2\mathbf{J}(\widehat{\mathbf{x}}) + \mathbf{H}(\widehat{\mathbf{x}}, -\mathbf{J}(\widehat{\mathbf{x}})^{-1}\mathbf{f}(\widehat{\mathbf{x}})))\widetilde{\mathbf{x}}$$

which can be solved to obtain the multidimensional Halley increment :

$$\widetilde{\mathbf{x}} = -2(2\mathbf{J}(\widehat{\mathbf{x}}) + \mathbf{H}(\widehat{\mathbf{x}}, -\mathbf{J}(\widehat{\mathbf{x}})^{-1}\mathbf{f}(\widehat{\mathbf{x}})))^{-1}\mathbf{f}(\widehat{\mathbf{x}})$$

The multidimensional Halley method is extremely costly since it involves the computation of the Hessian of the functions and two matrix inversions.

2.2.1.3 The multidimensional Efficient Newton Method

The efficient Newton method can be obtained by considering the Taylor series of $\mathbf{f}(\mathbf{x})$ about $\overline{\mathbf{x}}$:

$$\mathbf{f}(\overline{\mathbf{x}}+\mathbf{x}) = \mathbf{f}(\overline{\mathbf{x}}) + \mathbf{J}(\overline{\mathbf{x}})\mathbf{x} + \frac{1}{2}\mathbf{H}(\mathbf{x}^*,\mathbf{x})\mathbf{x} \tag{2.78}$$

Since $\mathbf{f}(\overline{\mathbf{x}}) = 0$, keeping terms up to first order we obtain :

$$\mathbf{f}(\overline{\mathbf{x}}+\mathbf{x}) \approx \mathbf{J}(\overline{\mathbf{x}})\mathbf{x} \tag{2.79}$$

Evaluating the equation at $\mathbf{x} = -\widetilde{\mathbf{x}} = \widehat{\mathbf{x}} - \overline{\mathbf{x}}$ gives :

$$\mathbf{f}(\widehat{\mathbf{x}}) \approx -\mathbf{J}(\overline{\mathbf{x}})\widetilde{\mathbf{x}}$$

We obtain an efficient Newton method by computing :

$$\widetilde{\mathbf{x}} = -\mathbf{J}(\overline{\mathbf{x}})^{-1}\mathbf{f}(\widehat{\mathbf{x}})$$

It is efficient since the inverse of the Jacobian is computed once and for all.

2.2.1.4 The multidimensional ESM

Using the first order series of the matrix $\mathbf{J}(\mathbf{x})$ about $\widehat{\mathbf{x}}$, we obtain :

$$\mathbf{J}(\widehat{\mathbf{x}}+\mathbf{x}) = \mathbf{J}(\widehat{\mathbf{x}}) + \mathbf{H}(\widehat{\mathbf{x}},\mathbf{x}) + \frac{1}{2!}\mathbf{Q}(\mathbf{x}^*,\mathbf{x}^2) \tag{2.80}$$

Plugging equation (2.80) in equation (2.64), we obtain :

$$\mathbf{f}(\widehat{\mathbf{x}}+\mathbf{x}) = \mathbf{f}(\widehat{\mathbf{x}}) + \frac{1}{2}(\mathbf{J}(\widehat{\mathbf{x}}+\mathbf{x}) + \mathbf{J}(\widehat{\mathbf{x}}))\,\mathbf{x} - \frac{1}{12}\mathbf{Q}(\mathbf{x}^*,\mathbf{x}^2)\,\mathbf{x} \tag{2.81}$$

where the second-order terms in $\mathbf{x}$ have disappeared. Without computing the Hessian, we obtain an efficient second-order approximation of $\mathbf{f}$ (i.e. only using first order derivatives) by setting $\mathbf{f}(\widehat{\mathbf{x}}+\widetilde{\mathbf{x}}) = \mathbf{f}(\overline{\mathbf{x}}) = 0$:

$$\mathbf{f}(\widehat{\mathbf{x}}+\widetilde{\mathbf{x}}) \approx \mathbf{f}(\widehat{\mathbf{x}}) + \frac{1}{2}(\mathbf{J}(\overline{\mathbf{x}}) + \mathbf{J}(\widehat{\mathbf{x}}))\,\widetilde{\mathbf{x}} = 0 \tag{2.82}$$

The displacement can be obtained by computing the inverse of the mean of the Jacobians :

$$\widetilde{\mathbf{x}} = -2\,(\mathbf{J}(\overline{\mathbf{x}}) + \mathbf{J}(\widehat{\mathbf{x}}))^{-1}\,\mathbf{f}(\widehat{\mathbf{x}}) \tag{2.83}$$

If $\mathbf{f}(\mathbf{x})$ is quadratic in $\mathbf{x}$, then the equation (2.82) is not an approximation anymore. Thus, we can estimate the true parameters of the warping in only one iteration. If the vector function $\mathbf{f}(\mathbf{x})$ is not quadratic, we can expect an improvement over the standard Newton method since the ESM has a higher order of convergence at the same computational cost per iteration.

2.2.2 Generalization to nonlinear systems on Lie groups

In order to understand the generalization of the methods presented in the previous section to nonlinear systems of equations defined on Lie groups the reader should be familiar with Lie group theory (see [Warner 71], [Varadarajan 74], [Hall 03]). The theorems proved when the Lie group is $\mathbb{R}^n$ can be generalized as in [Mahony 02].

2.2.2.1 Functional iteration on Lie groups

Consider the problem of iteratively finding a root of the equation :

$$\boldsymbol{f}(\boldsymbol{x}) = 0 \tag{2.84}$$

given the $(n \times 1)$ vector function $\boldsymbol{f} : G \mapsto \mathbb{R}^n$ where $\boldsymbol{x} \in G$. Let the dimension of the Lie group G be $\dim(G) = n$. Let me suppose exists $\overline{\boldsymbol{x}}$ such that $\boldsymbol{f}(\overline{\boldsymbol{x}}) = 0$. Similarly to the multidimensional case, we would like to solve the problem iteratively starting from an approximation $\widehat{\boldsymbol{x}}_0$ of the true solution $\overline{\boldsymbol{x}}$. The problem is to find an increment $\widetilde{\boldsymbol{x}}$ in order to generate a sequence :

$$\widehat{\boldsymbol{x}}_{k+1} = \widehat{\boldsymbol{x}}_k \cdot \widetilde{\boldsymbol{x}}_k \tag{2.85}$$

such that

$$\lim_{k\to\infty} \widehat{\boldsymbol{x}}_k = \overline{\boldsymbol{x}} \tag{2.86}$$

The equation (2.85) is a generalization of equation (2.70) since one can choose $G = \mathbb{R}^n$ (in this case the group composition is the addition). If we suppose that the approximation $\widehat{\boldsymbol{x}}$ is close enough to the true solution $\overline{\boldsymbol{x}}$ then the increments are close to the identity $\widehat{\boldsymbol{x}}^{-1} \cdot \overline{\boldsymbol{x}} \approx e$. Let $\mathbf{z} \in \mathbb{R}^n$ be a vector of coordinates of the Lie algebra $\mathfrak{g}$ of the Lie Group G such that

$$\boldsymbol{x}(\mathbf{z}) = \exp\left(\sum_{i=1}^{n} z_i \mathbf{A}_i\right)$$

where the exponential map is a map $\exp\colon \mathfrak{g} \mapsto G$ and where A_i are vector fields in $\mathfrak{g}$ (for matrix Lie groups $\mathbf{A}_i$ are $(n \times n)$ matrices). Then, suppose that $\widetilde{\mathbf{z}}$ exists such that $\boldsymbol{x}(\widetilde{\mathbf{z}}) = \widehat{\boldsymbol{x}}^{-1} \cdot \overline{\boldsymbol{x}}$. The Taylor series of $\boldsymbol{f}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}))$ about $\mathbf{z} = 0$ is :

$$\boldsymbol{f}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z})) = \boldsymbol{f}(\widehat{\boldsymbol{x}}) + \boldsymbol{\mathcal{J}}(\widehat{\boldsymbol{x}})\,\mathbf{z} + \frac{1}{2!}\boldsymbol{\mathcal{H}}(\widehat{\boldsymbol{x}}, \mathbf{z})\,\mathbf{z} + \frac{1}{3!}\boldsymbol{\mathcal{Q}}(\boldsymbol{x}^*, \mathbf{z}^2)\,\mathbf{z} \tag{2.87}$$

where the $(n \times n)$ matrices $\boldsymbol{\mathcal{J}}(\widehat{\boldsymbol{x}})$, $\boldsymbol{\mathcal{H}}(\widehat{\boldsymbol{x}}, \mathbf{z})$ and $\boldsymbol{\mathcal{Q}}(\widehat{\boldsymbol{x}}, \mathbf{z}^2)$ are :

$$\begin{aligned}
\boldsymbol{\mathcal{J}}(\widehat{\boldsymbol{x}}) &= \nabla \boldsymbol{f}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{y}))_{\mathbf{y}=0} && (2.88)\\
\boldsymbol{\mathcal{H}}(\widehat{\boldsymbol{x}}, \mathbf{z}) &= \nabla(\boldsymbol{\mathcal{J}}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{y}))\,\mathbf{z})|_{\mathbf{y}=0} && (2.89)\\
\boldsymbol{\mathcal{Q}}(\widehat{\boldsymbol{x}}, \mathbf{z}^2) &= \nabla(\boldsymbol{\mathcal{H}}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{y}), \mathbf{z})\,\mathbf{z})|_{\mathbf{y}=0} && (2.90)
\end{aligned}$$

Finally, we can follow the same developments described in the previous section to obtain the increment $\widetilde{\boldsymbol{x}}$ for each method.

2.2.2.2 Newton methods for nonlinear systems on Lie groups

The Newton-Raphson increment on Lie groups can be computed as follows :

$$\widetilde{\boldsymbol{x}} = \boldsymbol{x}\left(-\boldsymbol{\mathcal{J}}(\widehat{\boldsymbol{x}})^{-1}\boldsymbol{f}(\widehat{\boldsymbol{x}})\right)$$

while the Efficient Newton increment is :

$$\widetilde{\boldsymbol{x}} = \boldsymbol{x}\left(-\boldsymbol{\mathcal{J}}(\overline{\boldsymbol{x}})^{-1}\boldsymbol{f}(\widehat{\boldsymbol{x}})\right)$$

2.2.2.3 The ESM for nonlinear systems on Lie groups

The ESM increment on Lie groups can be computed as follows :

$$\widetilde{\boldsymbol{x}} = \boldsymbol{x}\left(-2\left(\boldsymbol{\mathcal{J}}(\widehat{\boldsymbol{x}}) + \boldsymbol{\mathcal{J}}(\overline{\boldsymbol{x}})\right)^{-1}\boldsymbol{f}(\widehat{\boldsymbol{x}})\right) \tag{2.91}$$

2.3 Optimization of nonlinear least squares problems

Consider the following general unconstrained optimization problem :

$$\min_{\mathbf{x}} c(\mathbf{x}) \tag{2.92}$$

where $c(\mathbf{x})$ is a positive cost function that should be minimized. The necessary conditions to obtain a (local) minimum of the cost function are :

$$\mathbf{g}_c(\mathbf{x}) = \nabla c(\mathbf{x}) = 0 \tag{2.93}$$

$$\mathbf{H}_c(\mathbf{x}) = \nabla \mathbf{g}_c(\mathbf{x}) > 0 \tag{2.94}$$

where $\mathbf{g}_c(\mathbf{x})$ is the $(1 \times n)$ gradient of the cost function and $\mathbf{H}_c(\mathbf{x})$ is the $(n \times n)$ Hessian matrix of the cost function. Thus, the problem is to solve the system of n equation in n unknowns defined in equation (2.93). Suppose that we have a system of m nonlinear equations and n unknowns (with $m > n$). In the presence of measurement errors the system $\mathbf{f}(\mathbf{x}) = 0$ could not have a solution. Moreover, since there are more equations than unknowns we probably do not know how to choose a particular set of n equations over m to solve the problem. Thus, we can rewrite the root-finding problem as a least squares minimization problem. We look for the minimum of a positive cost function $c(\mathbf{x}) = \frac{1}{2}\mathbf{f}(\mathbf{x})^\top \mathbf{f}(\mathbf{x})$:

$$\min_{\mathbf{x}} c(\mathbf{x}) = \min_{\mathbf{x}} \frac{1}{2}\mathbf{f}(\mathbf{x})^\top \mathbf{f}(\mathbf{x}) \tag{2.95}$$

starting from a close approximation $\widehat{\mathbf{x}}$ of the true root $\overline{\mathbf{x}}'$. In the particular case of a least-squares problem we have :

$$\mathbf{g}_c(\mathbf{x}) = \mathbf{f}(\mathbf{x})^\top \mathbf{J}(\mathbf{x}) \tag{2.96}$$

$$\mathbf{H}_c(\mathbf{x}) = \mathbf{J}(\mathbf{x})^\top \mathbf{J}(\mathbf{x}) + \sum_{i=1}^{n} f_i(\mathbf{x})\, \mathbf{H}_i(\mathbf{x}) \tag{2.97}$$

If we transform the root-finding problem into a minimization problem, the minimum of the cost function is obtained when $\mathbf{g}_c(\mathbf{x}) = 0$. Thus we will find either $\mathbf{f}(\overline{x}') = 0$ or $\mathbf{f}(\overline{x}') \in \ker(\mathbf{J}(\overline{x}'))$. Note that solution $\overline{\mathbf{x}}'$ of the non-linear least squares optimization coincides with the solution $\overline{\mathbf{x}}$ of the nonlinear systems only if $\mathbf{f}(\overline{\mathbf{x}}) = 0$. Because of aberrant measurements errors the estimated solution can be very far from the true solution. In this case, a robust optimization method should be used (see section 2.3.3).

2.3.1 The Newton optimization and approximated methods

The Newton optimization method performs a first order approximation of the gradient of the cost function (or equivalently a second-order approximation of the cost function). This implies the computation of the second derivatives (the Hessian matrices) of the vector function $\mathbf{f}$. We can apply the Newton-Raphson method to iteratively solve equation (2.93)

which is a system of n non-linear equations in n unknowns. Taking terms up to first order of the Taylor series of $\mathbf{g}_c(\mathbf{x})$ about $\widehat{\mathbf{x}}$ we obtain :

$$\mathbf{g}_c(\widehat{\mathbf{x}} + \mathbf{x}) \approx \mathbf{g}_c(\widehat{\mathbf{x}}) + \mathbf{x}^\top \mathbf{H}_c(\widehat{\mathbf{x}}) \tag{2.98}$$

Setting $\mathbf{g}_c(\widehat{\mathbf{x}} + \mathbf{x}) = 0$ we have to solve the following linear system :

$$\mathbf{g}_c(\widehat{\mathbf{x}}) + \widetilde{\mathbf{x}}^\top \mathbf{H}_c(\widehat{\mathbf{x}}) = 0 \tag{2.99}$$

whose solution is :

$$\widetilde{\mathbf{x}} = -\left(\mathbf{H}_c(\widehat{\mathbf{x}})\right)^{-1} \mathbf{g}_c(\widehat{\mathbf{x}})^\top \tag{2.100}$$

In the case of a least-squares optimization, using equations (2.96) and (2.97), the solution can be written as :

$$\widetilde{\mathbf{x}} = -\left(\mathbf{J}(\widehat{\mathbf{x}})^\top \mathbf{J}(\widehat{\mathbf{x}}) + \sum_{i=1}^{n} f_i(\widehat{\mathbf{x}})\, \mathbf{H}_i(\widehat{\mathbf{x}})\right)^{-1} \mathbf{J}(\widehat{\mathbf{x}})^\top \mathbf{f}(\widehat{\mathbf{x}}) \tag{2.101}$$

2.3.1.1 The Gauss-Newton method

The Gauss-Newton method consists in approximating the Hessian of the cost function. Supposing that the residuals at the solution are small $f_i(\widehat{\mathbf{x}}) \to f_i(\overline{\mathbf{x}}') \approx 0$ we can set :

$$\mathbf{H}_c(\widehat{\mathbf{x}}) \approx \mathbf{J}(\widehat{\mathbf{x}})^\top \mathbf{J}(\widehat{\mathbf{x}})$$

In this case the Gauss-Newton step can be written as :

$$\widetilde{\mathbf{x}} = -\left(\mathbf{J}(\widehat{\mathbf{x}})^\top \mathbf{J}(\widehat{\mathbf{x}})\right)^{-1} \mathbf{J}(\widehat{\mathbf{x}})^\top \mathbf{f}(\widehat{\mathbf{x}}) \tag{2.102}$$

where $\det(\mathbf{J}(\widehat{\mathbf{x}})^\top \mathbf{J}(\widehat{\mathbf{x}})) \neq 0$. More generally, we can use the pseudo-inverse of the matrix :

$$\widetilde{\mathbf{x}} = -\mathbf{J}(\widehat{\mathbf{x}})^+ \mathbf{f}(\widehat{\mathbf{x}}) \tag{2.103}$$

since $\mathbf{J}(\widehat{\mathbf{x}})^+ = \left(\mathbf{J}(\widehat{\mathbf{x}})^\top \mathbf{J}(\widehat{\mathbf{x}})\right)^{-1} \mathbf{J}(\widehat{\mathbf{x}})^\top$ if $\mathrm{rank}(\mathbf{J}(\widehat{\mathbf{x}})) = n$. Obviously, if $m = n$ then $\mathbf{J}(\widehat{\mathbf{x}})^+ = \mathbf{J}(\widehat{\mathbf{x}})^{-1}$ and the Gauss-Newton method applied to the minimization of the cost function $c(\mathbf{x})$ is equivalent to the Newton method applied to the root search of the function $\mathbf{f}(\mathbf{x})$.

2.3.1.2 The Efficient Gauss-Newton method

If we are able to measure $\mathbf{J}(\overline{\mathbf{x}})$ then we can use an efficient Gauss-Newton method :

$$\mathbf{g}_c(\overline{\mathbf{x}} + \mathbf{x}) \approx \mathbf{g}_c(\overline{\mathbf{x}}) + \mathbf{x}^\top \mathbf{H}_c(\overline{\mathbf{x}}) \tag{2.104}$$

Setting $\mathbf{g}_c(\overline{\mathbf{x}} + \widetilde{\mathbf{x}}) = \mathbf{g}_c(\widehat{\mathbf{x}})$ we have to solve :

$$\mathbf{g}_c(\widehat{\mathbf{x}}) = -\widetilde{\mathbf{x}}^\top \mathbf{H}_c(\overline{\mathbf{x}}) = 0 \tag{2.105}$$

whose solution is :

$$\widetilde{\mathbf{x}} = -\mathbf{J}(\overline{\mathbf{x}})^+ \mathbf{f}(\widehat{\mathbf{x}})$$

This method is efficient since $\mathbf{J}(\overline{\mathbf{x}})^+$ can be computed once and for all.

2.3.1.3 The Steepest Descent method

The Steepest Descent method consists in approximating the Hessian of the cost function as follows :

$$\mathbf{H}_c(\widehat{\mathbf{x}}) \approx \alpha \mathbf{I}$$

where α is a positive constant. In this case the Steepest Descent method step can be written as :

$$\widetilde{\mathbf{x}} = -\alpha \mathbf{J}(\widehat{\mathbf{x}})^\top \mathbf{f}(\widehat{\mathbf{x}})$$

The value of the parameter α can be obtained by solving a one dimensional minimization problem along the direction given by the vector $\widetilde{\mathbf{x}}$ [Dennis 83]. An efficient version of the Steepest Descent method can be obtained using $\overline{\mathbf{x}}$ instead of $\widehat{\mathbf{x}}$ in the Jacobian.

2.3.1.4 The Levemberg-Marquardt method

The Levemberg-Marquardt method consists in approximating the Hessian of the cost function as follows :

$$\mathbf{H}_c(\widehat{\mathbf{x}}) \approx \mathbf{J}(\widehat{\mathbf{x}})^\top \mathbf{J}(\widehat{\mathbf{x}}) + \alpha \mathbf{I}$$

where α is a positive constant. In this case the Levemberg-Marquardt step can be written as :

$$\widetilde{\mathbf{x}} = -\left(\mathbf{J}(\widehat{\mathbf{x}})^\top \mathbf{J}(\widehat{\mathbf{x}}) + \alpha \mathbf{I}\right)^{-1} \mathbf{J}(\widehat{\mathbf{x}})^\top \mathbf{f}(\widehat{\mathbf{x}})$$

Note that if $\alpha = 0$ then we obtain the Gauss-Newton method.

2.3.2 The ESM for nonlinear least squares optimization

If we apply the ESM to the non-linear system defined in equation (2.93) we obtain :

$$\widetilde{\mathbf{x}} = -\frac{1}{2}(\mathbf{H}_c(\widehat{\mathbf{x}}) + \mathbf{H}_c(\overline{\mathbf{x}}'))^{-1}\mathbf{g}_c(\widehat{\mathbf{x}})$$

The ESM method as it is defined in this equation would require the computation of Hessians matrices like the Newton method.

2.3.2.1 Approximated ESM

As for the Gauss-Newton method we can compute an approximation of the gradient of the cost function as follows. Consider the gradient of the cost function :

$$\mathbf{g}_c(\widehat{\mathbf{x}} + \mathbf{x}) = \mathbf{f}(\widehat{\mathbf{x}} + \mathbf{x})^\top \mathbf{J}(\widehat{\mathbf{x}} + \mathbf{x})$$

From equation (2.80) we compute the following expression for the Jacobian :

$$\mathbf{J}(\widehat{\mathbf{x}} + \mathbf{x}) = \frac{1}{2}\left(\mathbf{J}(\widehat{\mathbf{x}} + \mathbf{x}) + \mathbf{J}(\widehat{\mathbf{x}})\right) + \frac{1}{2}\mathbf{H}(\widehat{\mathbf{x}}, \mathbf{x}) + \frac{1}{4}\mathbf{Q}(\mathbf{x}^*, \mathbf{x}^2)$$

Thus, using the efficient second-order approximation :

$$\begin{aligned}\mathbf{g}_c(\widehat{\mathbf{x}}+\mathbf{x}) &= \frac{1}{2}\left(\mathbf{f}(\widehat{\mathbf{x}})+\frac{1}{2}(\mathbf{J}(\widehat{\mathbf{x}}+\mathbf{x})+\mathbf{J}(\widehat{\mathbf{x}}))\,\mathbf{x}-\frac{1}{12}\mathbf{Q}(\mathbf{x}^*,\mathbf{x}^2)\,\mathbf{x}\right)^\top \\ &\quad \left(\mathbf{J}(\widehat{\mathbf{x}}+\mathbf{x})+\mathbf{J}(\widehat{\mathbf{x}})+\mathbf{H}(\widehat{\mathbf{x}},\mathbf{x})+\frac{1}{2}\mathbf{Q}(\mathbf{x}^*,\mathbf{x}^2)\right) \end{aligned} \tag{2.106}$$

which can also be rewritten as follows :

$$\begin{aligned}\mathbf{g}_c(\widehat{\mathbf{x}}+\mathbf{x}) &= \frac{1}{2}\left(\mathbf{f}(\widehat{\mathbf{x}})+\frac{1}{2}\left(\mathbf{J}(\widehat{\mathbf{x}}+\mathbf{x})+\mathbf{J}(\widehat{\mathbf{x}})\right)\,\mathbf{x}\right)^\top(\mathbf{J}(\widehat{\mathbf{x}}+\mathbf{x})+\mathbf{J}(\widehat{\mathbf{x}}))+ \\ &+ \frac{1}{2}\mathbf{f}(\widehat{\mathbf{x}}+\mathbf{x})^\top\left(\mathbf{H}(\widehat{\mathbf{x}},\mathbf{x})+\frac{1}{2}\mathbf{Q}(\mathbf{x}^*,\mathbf{x}^2)\right) \\ &- \frac{1}{24}\mathbf{x}^\top\mathbf{Q}(\mathbf{x}^*,\mathbf{x}^2)^\top(\mathbf{J}(\widehat{\mathbf{x}}+\mathbf{x})+\mathbf{J}(\widehat{\mathbf{x}})) \end{aligned} \tag{2.107}$$

Setting $\mathbf{x}=\widetilde{\mathbf{x}}'$ we have $\widehat{\mathbf{x}}+\widetilde{\mathbf{x}}'=\overline{\mathbf{x}}'$ and $\mathbf{g}_c(\overline{\mathbf{x}}')=0$. Thus :

$$\begin{aligned}\mathbf{g}_c(\overline{\mathbf{x}}') &= \frac{1}{2}\left(\mathbf{J}(\overline{\mathbf{x}})+\mathbf{J}(\widehat{\mathbf{x}})\right)^\top\left(\mathbf{f}(\widehat{\mathbf{x}})+\frac{1}{2}\left(\mathbf{J}(\overline{\mathbf{x}})+\mathbf{J}(\widehat{\mathbf{x}})\right)\,\widetilde{\mathbf{x}}\right)+ \\ &+ \frac{1}{2}\mathbf{f}(\overline{\mathbf{x}}')^\top\left(\mathbf{H}(\widehat{\mathbf{x}},\widetilde{\mathbf{x}})+\frac{1}{2}\mathbf{Q}(\mathbf{x}^*,\widetilde{\mathbf{x}}^2)\right) \\ &- \frac{1}{24}\widetilde{\mathbf{x}}^\top\mathbf{Q}(\mathbf{x}^*,\widetilde{\mathbf{x}}^2)^\top(\mathbf{J}(\overline{\mathbf{x}}')+\mathbf{J}(\widehat{\mathbf{x}}))=0 \end{aligned} \tag{2.108}$$

that can be approximated by :

$$\mathbf{g}_c(\overline{\mathbf{x}}')\approx\frac{1}{2}\left(\mathbf{f}(\widehat{\mathbf{x}})+\frac{1}{2}\left(\mathbf{J}(\overline{\mathbf{x}})+\mathbf{J}(\widehat{\mathbf{x}})\right)\,\widetilde{\mathbf{x}}\right)^\top(\mathbf{J}(\overline{\mathbf{x}})+\mathbf{J}(\widehat{\mathbf{x}}))=0 \tag{2.109}$$

There are two different approximations. The first approximation concerns terms that depend on the residuals $\mathbf{f}(\overline{\mathbf{x}}')$. Indeed, in the ideal case $\mathbf{f}(\overline{\mathbf{x}}')\approx 0$. The second approximation concerns terms that are third order in $\widetilde{\mathbf{x}}$. Finally, we find the solution :

$$\widetilde{\mathbf{x}}\approx -2\left(\mathbf{J}(\widehat{\mathbf{x}})+\mathbf{J}(\overline{\mathbf{x}}')\right)^+\mathbf{f}(\widehat{\mathbf{x}}) \tag{2.110}$$

2.3.2.2 Comparison with standard optimization methods

The main advantage of the Efficient Second-order approximation Method is the cubic order of convergence. Another advantage is the large domain of convergence that allows to avoid some local minima close (i.e. when the second-order approximation is valid) to the global one. These advantages are demonstrate here with the help of two simple examples. Consider a (4×1) vector function $\mathbf{f}(\mathbf{x})$ quadratic in a (2×1) parameter vector $\mathbf{x}$:

$$\mathbf{f}(\mathbf{x})=\boldsymbol{B}\,\mathbf{x}+\boldsymbol{A}(\mathbf{x})\,\mathbf{x}$$

where $\boldsymbol{B}$ is a (4×2) matrix and $\boldsymbol{A}(\mathbf{x})$ is a (4×2) which is linear in $\mathbf{x}$. We choose different coefficients in $\boldsymbol{B}$ and $\boldsymbol{A}(\mathbf{x})$ for each example. The objective is to find the minimum of the SSD cost function :

$$c(\mathbf{x}) = \frac{1}{2}\mathbf{f}(\mathbf{x})^\top \mathbf{f}(\mathbf{x})$$

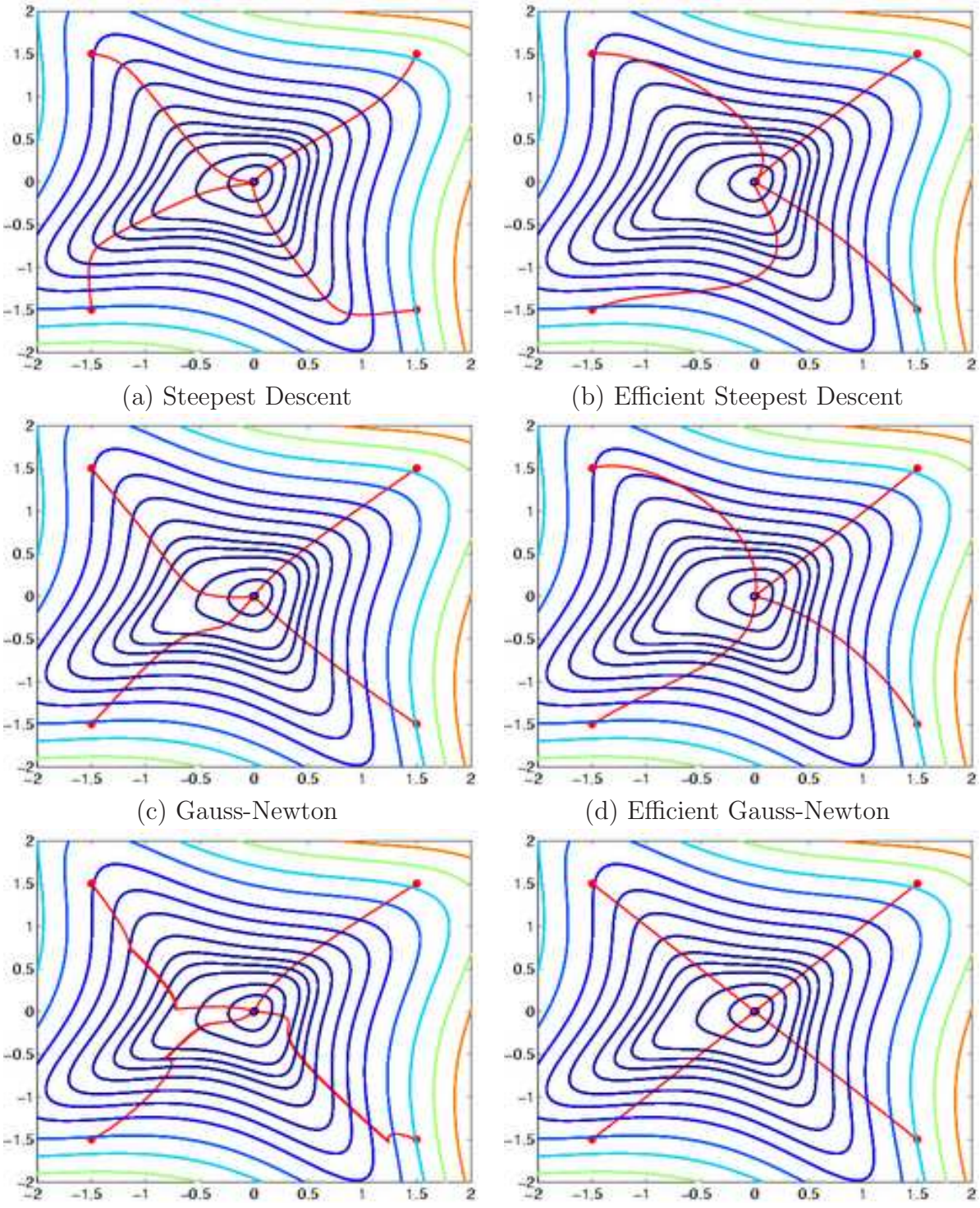
which, in this case, is of degree 4 in $\mathbf{x}$. In each example I will test the Steepest Descent and the Efficient Steepest Descent methods, the Gauss-Newton and the Efficient Gauss-Newton methods, the Newton method and the ESM. Thus, I suppose that we can measure the constant Jacobian $\mathbf{J}(\mathbf{0})$ and the varying Jacobian $\mathbf{J}(\mathbf{x}_0)$. The simulation is repeated 4 times with different starting points : $\mathbf{x}_0 \in \{(\pm 1.5, \pm 1.5)\}$. For each simulation I take at each iteration a very small step towards the solution in order to display a continuous trajectory from the starting point $\mathbf{x}_0$ to the minimum ($\mathbf{0}$ in our case). The length of the trajectory represents the convergence rate of each algorithm. Obviously, the ideal path (i.e. the shortest one) would be a straight line from $\mathbf{x}_0$ to $\mathbf{0}$.

2.3.2.2.1 Convergence rate

The results for 6 different minimization methods are given in Figure 2.1. The contours represent isolines of the SSD (i.e. the cost function has the same value for each point of the contour) while the red lines represent the paths for each starting point. Figure 2.3.2.2.2 shows that the varying Steepest Descent method always moves in a direction perpendicular to the isolines. For this reason, it has a slow convergence rate and cannot reach the minimum following a straight line. The paths for the constant Steepest Descent method are even longer (see the path lengths in Figure 2.3.2.2.2). The constant (Figure 2.3.2.2.2) and the varying (Figure 2.3.2.2.2) Gauss-Newton methods perform better than the constant and the varying Steepest Descent methods respectively. In fact, in the constant and the varying Gauss-Newton methods a rough approximation of the Hessian is used. Ill-conditioned and indefinite Hessian matrices cause the oscillations of the Newton method in Figure 2.3.2.2.2. Finally, the ESM method gives the best solution since the paths in Figure 2.3.2.2.2 are straight lines. Indeed, when the function $\mathbf{f}(\mathbf{x})$ is exactly quadratic we can correctly estimate the displacement in only one step and thus the correct descent direction regardless of the shape of the isolines.

2.3.2.2.2 Domain of convergence

In the second simulation, we choose a different quadratic function $\mathbf{f}(\mathbf{x})$ such that the corresponding SSD cost function has a local minimum very close to the global minimum. The Newton method and methods with varying Jacobian fall into the local minimum when the starting point is close to it (see Figures 2.3.2.2.2, 2.3.2.2.2 and 2.3.2.2.2). In this case, methods with constant Jacobian can eventually diverge (see Figures 2.3.2.2.2 and 2.3.2.2.2). Indeed, the constant Jacobian approximation is valid only in a neighborhood of the true solution. On the other hand, the ESM method follows the shortest path (see Figure 2.3.2.2.2). Thus, if $\mathbf{f}(\mathbf{x})$ is locally quadratic the ESM method is able to avoid local minima.

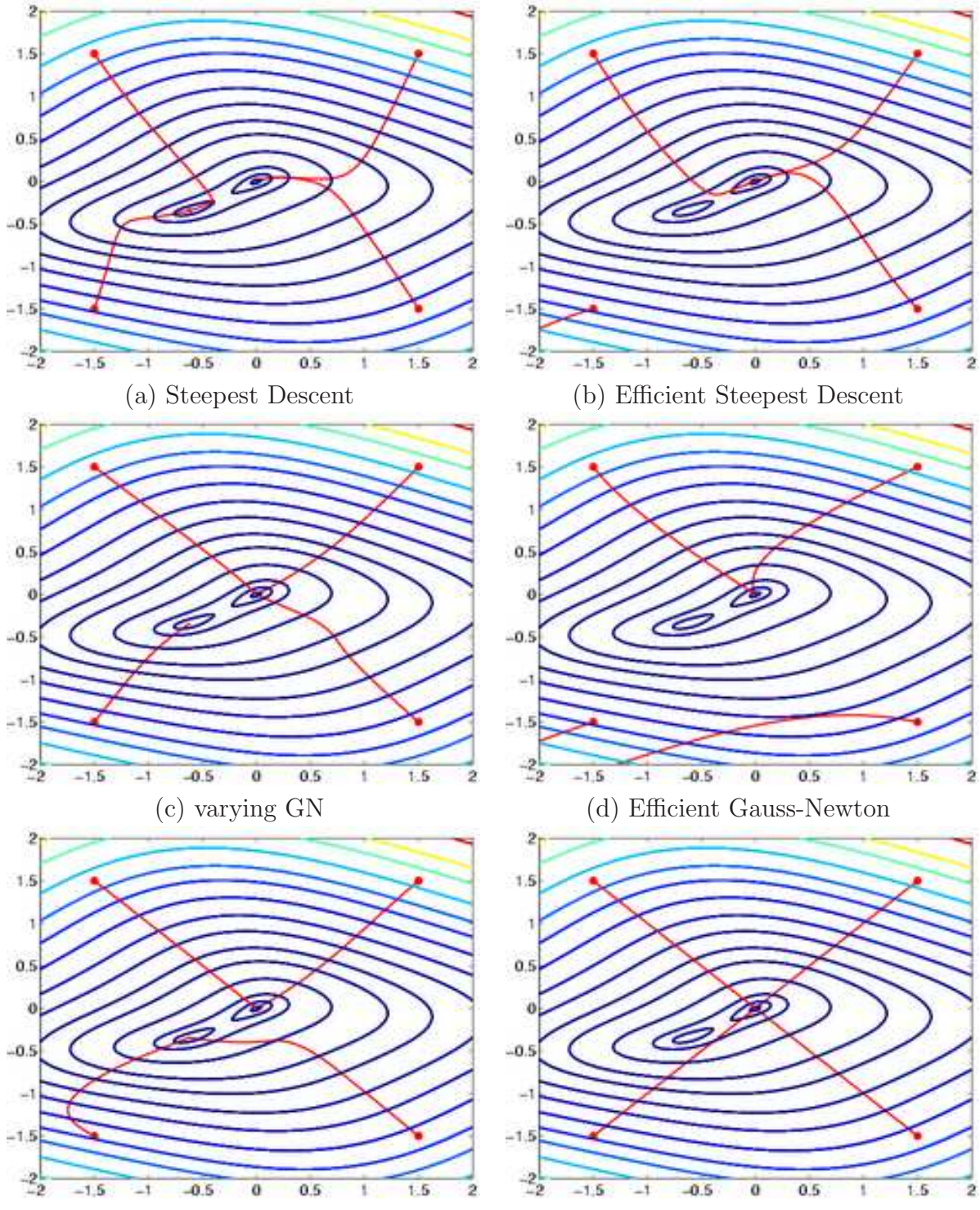


(a) Steepest Descent (b) Efficient Steepest Descent

(c) Gauss-Newton (d) Efficient Gauss-Newton

(e) Newton (f) ESM

FIG. 2.1 – Comparing the behavior of six different minimization methods.



(a) Steepest Descent (b) Efficient Steepest Descent

(c) varying GN (d) Efficient Gauss-Newton

(e) Newton (f) ESM

FIG. 2.2 – Comparing the behavior of six different minimization methods.

2.3.3 Robust optimization

When dealing with robot vision applications in a real world some measurements may be aberrant. These aberrant measures are called *outliers* and in theory only one of them can make a least squares algorithm find a solution arbitrarily far from the true solution. Least squares problems are not very robust because the objective function can grow indefinitely and outliers may become dominating w.r.t. measurements that verify the model. Alternatively, it is preferable to use robust methods that modify the objective function so as to limit the influence of the most significant residues. The principal consequence is, usually, a slower convergence speed of the optimization algorithms. Indeed, it is very difficult to distinguish, at first, outliers and inliers. Therefore, some inliers may be initially filtered which leads to a reduction in the convergence speed. Nevertheless, the most difficult case to solve arises when small but aberrant residues move the minimum of the objective function.

Several robust optimization methods have been proposed in the literature. A comparison of the most known algorithm applied to robot vision applications was proposed in [Malis 06]. I will describe here the methods that have worked better in practice.

2.3.3.1 The M-estimators

The principle of the M-estimators is to modify the objective function of the least squares by penalizing the largest residues. The objective function is then defined by :

$$c(\mathbf{x}) = \sum_{i=1}^{n} \rho(f_i(\mathbf{x})) \tag{2.111}$$

where the function ρ is at least C^0. The gradient and the Hessian of the cost function are :

$$\begin{aligned}
\mathbf{g}_c(\mathbf{x}) &= \sum_{i=1}^{n} \frac{\partial \rho(f_i)}{\partial f_i} \frac{\partial f_i(\mathbf{x})}{\partial \mathbf{x}} \\
&= \left[\begin{array}{cccc} \frac{\partial \rho(f_1)}{\partial f_1} & \frac{\partial \rho(f_2)}{\partial f_2} & \cdots & \frac{\partial \rho(f_m)}{\partial f_m} \end{array} \right] \mathbf{J}(\mathbf{x}) \qquad (2.112) \\
\mathbf{H}_c(\mathbf{x}) &= \sum_{i=1}^{n} \frac{\partial^2 \rho(f_i)}{\partial f_i^2} \frac{\partial f_i(\mathbf{x})^\top}{\partial \mathbf{x}} \frac{\partial f_i(\mathbf{x})}{\partial \mathbf{x}} + \sum_{i=1}^{n} \frac{\partial \rho(f_i)}{\partial f_i} \frac{\partial^2 f_i(\mathbf{x})}{\partial \mathbf{x}^2} \\
&= \mathbf{J}(\mathbf{x})^\top \mathbf{D}\, \mathbf{J}(\mathbf{x}) + \sum_{i=1}^{n} \frac{\partial \rho(f_i)}{\partial f_i} \mathbf{H}_i(\mathbf{x}) \qquad (2.113)
\end{aligned}$$

where $\mathbf{D} = \text{diag}\left(\frac{\partial^2 \rho(f_1)}{\partial f_1^2}, \frac{\partial^2 \rho(f_2)}{\partial f_2^2}, \cdots, \frac{\partial^2 \rho(f_m)}{\partial f_m^2} \right)$. Then, we can apply the Newton-Raphson method.

Various functions ρ were proposed in the literature. Among all the possible robust estimators ρ, the most popular are those proposed by [Beaton 74] and [Huber 81]. The

function ρ proposed by [Huber 81] is defined by :

$$\rho(f_i) = \begin{cases} \frac{1}{2} f_i^2 & \text{if } f_i^2 \leq c \\ c\left(|f_i| - \frac{c}{2}\right) & \text{if } f_i^2 > c \end{cases} \tag{2.114}$$

where $c = 1.345\widehat{\sigma}$ and $\widehat{\sigma}$ is a robust estimation of the standard deviation of the inliers. It is usually defined by the *Median Absolute Deviation* (MAD) given by :

$$\widehat{\sigma} = 1.48 \text{ median}(|\mathbf{f} - \text{median}(\mathbf{f})|). \tag{2.115}$$

In the function ρ proposed by Huber, the weakest residues are regarded as in a least squares problem whereas the most significant residues are quickly limited (but not canceled).

The function ρ proposed by [Beaton 74] is defined by :

$$\rho(f_i) = \begin{cases} \frac{c^2}{6}\left[1 - \left(1 - \left(\frac{f_i}{c}\right)^2\right)^3\right] & \text{if } f_i^2 \leq c \\ \frac{c^2}{6} & \text{if } f_i^2 > c \end{cases} \tag{2.116}$$

where $c = 4.6851\widehat{\sigma}$ and where $\widehat{\sigma}$ is a robust estimation of the standard deviation of the inliers. As in the previous case, in the function ρ proposed by Beaton-Tukey, the weakest residues are regarded as in a least squares problem whereas the most significant residues are quickly canceled (completely this time).

2.3.3.2 The iteratively re-weighted least squares

An advantage of the M-estimation is that it can be implemented using a simple iterative re-weighted least squares algorithm. Considering a linear problem, the actual least squares problem aims at solving for $\mathbf{x}$ the following linear system $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{x}$ and $\mathbf{b}$ are vectors and $\mathbf{A}$ is a matrix. Iterative re-weighted least squares algorithms aim at solving the following system :

$$\mathbf{WAx} = \mathbf{Wb} \tag{2.117}$$

where $\mathbf{W} = \text{diag}(w_1, \ldots, w_m)$ is a diagonal matrix where the weight w_i reflects the confidence of each feature. The weights w_i, are usually given computed as follows :

$$w_i = \frac{1}{f_i} \frac{\partial \rho(f_i)}{\partial f_i} \tag{2.118}$$

where the ρ can be any robust function (see for example those presented in the previous section). The iteratively re-weighted algorithm estimates the value of $\mathbf{x}$ by solving the weighted least squares system of the equation (2.117), and reiterates until convergence. These methods act like automatic outlier rejectors since large residual values lead to very small weights.

Chapitre 3

Vision-based parametric estimation

Vision-based parametric estimation has been widely studied by computer vision and robotic scientists. It is an inverse problem since the values of some parameters of a model must be obtained from the observed visual data. An exhaustive description of all possible solutions is beyond the scope of this document. The reader may refer to well-known textbooks like [Faugeras 93, Hartley 00, Faugeras 01, Ma 03]. However, it is worth noting that there are two main approaches to solving this problem : correspondence-based methods and direct methods. Both have their advantages and drawbacks, which motivated the design of hybrid approaches (i.e. a combination of both methods).

Correspondence-based methods separate the data association problem from the solution of the non-linear system of equations introduced in Chapter 1. The data association problem is solved first, by finding the matching between corresponding pixels. The correspondences may be dense like optical flow [Horn 81] or sparse like features [Torr 00]. With respect to optical flow methods, feature-based methods are correspondence-based methods that can deal with larger inter-frame displacements in the images.

Direct methods [Horn 88, Stein 00, Irani 00] simultaneously solve the data association problem and the non-linear system of equations introduced in Chapter 1. The problem is formulated as the minimization of a cost function that is generally solved using the standard local numerical methods introduced in Chapter 2. Thus, direct methods are more adapted to solving incremental estimation problems, which is the case of video-sequences acquired with a sufficiently high frame rate.

The contributions of my research work have been the modeling and the design of efficient, accurate and robust estimation methods to meet the requirements needed for real-time applications and, in particular, vision-based robot control. I will show how to solve incremental direct image registration problems (visual tracking) and localization and/or mapping problems and how to apply the ESM described in Chapter 2.

The present chapter is organized as follows. First, feature-based estimation methods are presented. Their advantages and limitations are discussed. Then, direct methods are described with a particular focus on incremental approaches.

3.1 Feature-based methods

Suppose we have a collection of images acquired by a vision system. Feature-based methods are so called since they are based on the extraction of a set of features (like points, lines, contours, ...) from the images. They are correspondence-based methods since the matching problem is solved before the parametric estimation. Thus, feature-based methods are generally composed of three steps :

- Feature extraction ;
- Feature matching ;
- Parametric estimation (e.g. intrinsic parameters, structure, motion, ...).

Each of these steps is now discussed.

3.1.1 Features extraction

The raw images are filtered to extract some features (points, edges, contours, ...) corresponding to characteristic geometric elements of the observed scene. Several feature detectors have been proposed in the literature. Most of them are specialized in extracting key-points like [Harris 88], [Lowe 04], [Mikolajczyk 04], [Tuytelaars 04], or edges [Canny 86]. Higher level features (straight lines, closed contours,...) can be obtained by chaining several elementary features. Let me consider for example key-point features since they are generally more likely to be found in an image. An ideal filter $\boldsymbol{\varphi}$ would be able to extract from each image the exact coordinates of n points that correspond to 3D points of the scene :

$$\mathbf{s}_i = \boldsymbol{\varphi}(\boldsymbol{\psi}(\boldsymbol{\phi}_i, \boldsymbol{\kappa}_i, \boldsymbol{\gamma}, \mathbf{T}_i)) = (\mathbf{p}_{i1}, \mathbf{p}_{i2}, ..., \mathbf{p}_{in}) \tag{3.1}$$

The ideal filter should also be able to extract exactly the same points in all the images, if they are visible. Unfortunately this ideal filter does not exist and the accuracy and repeatability (precision) of common extractor filters are limited.

3.1.1.1 Accuracy

Accuracy in feature detection is extremely important. Indeed, the error that is made in the extraction will never be corrected in the following steps. Detection accuracy is affected by several factors (see for example [Deriche 90]). Firstly, the filter should be invariant to the photometric parameters $\boldsymbol{\phi}$, to the geometry of the scene and to the viewpoint. Secondly, the filter should be robust to noise. To achieve these invariances, filters are generally based on image processing (e.g. image gradients, smoothing, ...) that affects the accuracy of the extraction. Note also that, the majority of filters should also be invariant to the camera intrinsic parameters. Furthermore, most filters have been designed and optimized for perspective cameras. Their application to other cameras, like catadioptric sensors (due to their non uniform resolution), is not straightforward.

3.1.1.2 Repeatability

Feature association in two different images can only be performed if we are able to extract the same features in both images. Thus, another important characteristic of

feature extraction is repeatability. Perfect repeatability is extremely difficult to achieve. Indeed, the extraction of the features generally depends on a threshold to decide if the feature exists or not. When imaging conditions change the same feature may not be detected in different images.

3.1.2 Features matching

Once we have extracted a collection of features in several images we need to match them. Matching consists in finding which feature in the reference image corresponds to which feature in the current image. Exhaustive features association may be performed by considering all possible combinations of features and solving the underlying projection model. However, this approach is computationally expensive. Thus, feature matching algorithms are generally simplified by computing some characteristic descriptors for each feature. Then, using a similarity measure on the descriptors we can measure the most similar couples of features in the two images.

In general, feature descriptors are not invariant to all involved parameters (illumination, geometry, motion, ...). Moreover, the similarity measure cannot tolerate gross errors on the descriptors. Thus, several mismatched features are generally present at the end of the matching process. Some of the false matches (outliers) can be eliminated "a posteriori" using geometric constraints. Other false matches will not be eliminated since they will be compatible with the geometric constraints and will be considered as inliers. However, they will contribute to reducing the accuracy of the parametric estimation.

3.1.2.1 Descriptors

Descriptors are needed to simplify the data association. They are generally computed by considering a region around the interest points (the simplest descriptor being the region itself). Several descriptors have been proposed for interest points like [Lowe 04, Mikolajczyk 04, Tuytelaars 04]. They are generally robust to affine image transformations and to affine illumination changes. Therefore, the performance of these descriptors decreases with increasing perspective deformations or illumination changes.

3.1.2.2 Similarity measures

A similarity measure is needed to discriminate corresponding features. For example, when we consider as a descriptor all the intensities of a region around an interest point, a standard similarity measure is the zero-mean normalized cross correlation (ZNCC). Any distance between two descriptors can be used as a similarity measure. For example, [Lowe 04] uses an Euclidean distance between the (128×1) vectors containing the SIFT descriptors.

3.1.3 Parametric estimation

For simplicity, let me consider interest points as visual features and a monocular camera. In addition, suppose that the data association problem has been perfectly solved.

If it is not the case, we can use robust estimation techniques, like RANSAC [Fischler 81], the Hough transform [Duda 72] or M-estimators [Huber 81], to eliminate the outliers. If we assume a perfect matching between points, we can estimate the parameters of the projection model given in equation (1.20) by solving the following system of non-linear equations :

$$\boldsymbol{\pi}(\boldsymbol{\kappa}_i, \boldsymbol{\tau}(\mathbf{T}_i, \mathbf{m}_j)) = \overline{\mathbf{p}}_{ij}$$

where $i \in 1, 2, \ldots, n_i$ and $j \in 1, 2, \ldots, n_p$ (n_i and n_i being the number of images and points respectively). These equations depend on the camera intrinsic parameters $\boldsymbol{\kappa}_i$, the camera extrinsic parameters $\mathbf{T}_i$ and the structure of the rigid body $\mathbf{m}_j$. We can see in the following sections how the problems presented in Section 1.3.1.1, Section 1.3.1.2 and Section 1.3.1.3 have been solved by feature-based methods.

3.1.3.1 Calibration of the vision system

Suppose that a metric model of the structure is known (e.g. the calibration grid in Figure 3.1.3.1). For example, we know the coordinates $\mathbf{m}_j$ of several 3D points in a reference frame and we know that they correspond to the point $\overline{\mathbf{p}}_{ij}$ in the i-th image (the correspondence is generally given by the user). Then we can solve the following system of equations :

$$\boldsymbol{\pi}(\boldsymbol{\kappa}_i, \boldsymbol{\tau}(\mathbf{T}_i, \overline{\mathbf{m}}_j)) = \overline{\mathbf{p}}_{ij} \tag{3.2}$$

where the unknowns are the camera intrinsic parameters $\widehat{\boldsymbol{\kappa}}_i$ and the camera extrinsic parameters $\widehat{\mathbf{T}}_i$. Obviously we need at least as many equations as unknowns. If the camera is not zooming (i.e. the camera intrinsic parameters are constant) then $\widehat{\boldsymbol{\kappa}}_i = \widehat{\boldsymbol{\kappa}}$ and we obtain an over-constrained camera calibration : the number of equations is greater than the number of unknowns. If we have more equations than unknowns, the camera calibration can be defined as a non-linear least-squares optimization method. Calibration methods are based on the solution of the nonlinear system (3.2). For pinhole cameras see [Faugeras 87, Tsai 87, Weng 92, Zhang 00] while for central catadioptric cameras see [Geyer 99, Fabrizio 02, Mei 07]. For example, using 26 images of the grid in figure 3.1.3.1 with the method proposed by [Faugeras 87], we obtain the following focal length $f = 685 \pm 3$ and principal point $(u_0, v_0) = (322, 229) \pm (6, 40)$.

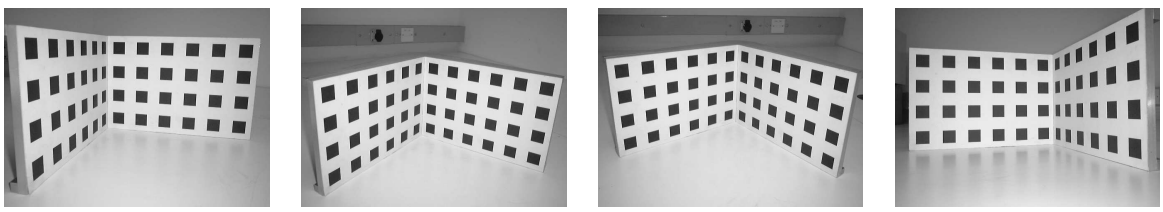


Image 1 Image 2 Image 3 Image 4

FIG. 3.1 – Images of a calibration grid.

3.1.3.2 Localization and/or mapping

Suppose that we have a calibrated vision system but the structure is unknown. Then the simultaneous localization and mapping problem can be formulated as the solution of the following system of non-linear equations :

$$\boldsymbol{\pi}(\overline{\boldsymbol{\kappa}}_i, \boldsymbol{\tau}(\mathbf{T}_i, \mathbf{m}_j)) = \overline{\mathbf{p}}_{ij} \tag{3.3}$$

Many approaches exist to solve this problem [Hartley 00, Faugeras 01, Ma 03]. They can be classified into two groups : bundle adjustment or incremental. Those based on bundle adjustment [Triggs 00] define a cost function containing all possible images and search simultaneously for all the parameters. Not only is an initial estimation of the optimization algorithms generally needed but also bundle adjustment supposes that all possible images have already been collected. Hierarchical bundle adjustment can be used to reduce the computational cost [Royer 07]. However, incremental approaches are more suitable for real-time robotics applications [Davison 03][Nister 06] [Se 05].

When we consider a few images several algorithms exist to efficiently solve the system of equations (3.3). For example, if we consider the image pairwise we can simplify the equations and estimate the well known essential matrix [Faugeras 93]. When we consider triplets of images we can estimate the trifocal tensor and if we consider quadruplets of images we can estimate the quadrifocal tensor (see [Hartley 00, Faugeras 01, Ma 03] for details).

Considering two images at a time is of particular interest for vision-based control applications based on the teaching-by-showing approach. In this case the system of equations (3.3) can be solved as follows. Let me set $\mathbf{m}_{ij} = \boldsymbol{\tau}(\mathbf{T}_i, \mathbf{m}_j)$. Then, we can "pseudo-inverse" the function $\boldsymbol{\pi}$ in order to obtain :

$$\mathbf{m}_{ij} \propto \boldsymbol{\pi}^{+}(\overline{\boldsymbol{\kappa}}_i, \overline{\mathbf{p}}_{ij}) \tag{3.4}$$

The "pseudo-inverse" function $\boldsymbol{\pi}^{+}$ is such that $\boldsymbol{\pi}(\overline{\boldsymbol{\kappa}}_i, \boldsymbol{\pi}^{+}(\overline{\boldsymbol{\kappa}}_i, \overline{\mathbf{p}}_{ij})) = \overline{\mathbf{p}}_{ij}$. Then using equation (1.4) we find that two image points in the i-th and k-th images are related by the following relationship :

$$\boldsymbol{\pi}^{+}(\overline{\boldsymbol{\kappa}}_i, \overline{\mathbf{p}}_{ij})^{\top}\, {}^{i}\mathbf{E}_k\, \boldsymbol{\pi}^{+}(\overline{\boldsymbol{\kappa}}_k, \overline{\mathbf{p}}_{kj}) = 0 \tag{3.5}$$

The essential matrices ${}^{i}\mathbf{E}_k$ can be computed from these equations [Longuet-Higgins 81] [Hartley 97] [Nister 04]. Then, since ${}^{i}\mathbf{E}_k = [{}^{i}\mathbf{t}_k]_{\times}\, {}^{i}\mathbf{R}_k$ we can extract the rotation ${}^{i}\mathbf{R}_k$ and the translation ${}^{i}\mathbf{t}_k$ up to a scale factor [Faugeras 93]. Finally, we can also reconstruct the structure up to this scale factor.

A degenerate case happens when the observed scene is a plane [Longuet-Higgins 84]. Dealing with degenerate cases is quite difficult. A possible solution is to maintain multiple hypotheses and to select the best model [Torr 98]. However, switching between several models can induce discontinuities in the estimation that one would like to avoid when using the estimated parameters in feedback control loops. Another solution is to solve the system of equations (3.3) using virtual parallax approaches [Boufama 95]. The equation can be rewritten in order to obtain the following system of equations :

$$\boldsymbol{\pi}^{+}(\overline{\boldsymbol{\kappa}}_i, \overline{\mathbf{p}}_{ij}) \propto {}^{i}\mathbf{H}_k \boldsymbol{\pi}^{+}(\overline{\boldsymbol{\kappa}}_k, \overline{\mathbf{p}}_{kj}) + \beta_j\, {}^{i}\mathbf{t}_k \tag{3.6}$$

where ${}^i\mathbf{H}_k$ is the homography matrix related to a virtual plane and β_j is a scalar that takes into account the distance (parallax) of the 3D point with respect to the virtual plane. Eliminating $\beta_j\,{}^i\mathbf{t}_k$ from the equations we can directly estimate the homography matrix ${}^i\mathbf{H}_k$ without knowing "a priori" if the scene is planar or not [Malis 00a] [Espiau 02].

If we know "a priori" that we are observing a planar scene we can take this information into account since in this case we have no parallax $\beta_j = 0$, $\forall j$. Thus, the points on the plane are related by a homography matrix (see equation (1.7)) :

$$\boldsymbol{\pi}^+(\overline{\boldsymbol{\kappa}}_i, \overline{\mathbf{p}}_{ij}) \propto {}^i\mathbf{H}_k \boldsymbol{\pi}^+(\overline{\boldsymbol{\kappa}}_k, \overline{\mathbf{p}}_{kj}) \tag{3.7}$$

These equations are linear in the entries of the homography matrices. Once the matrices ${}^i\mathbf{H}_k$ have been estimated we can extract the rotation and the translation up to a scale factor, and the normal to the plane. There are generally two possible solutions to the homography decomposition. To find these solutions, several numerical decomposition methods have been proposed [Faugeras 88, Zhang 96]. With such numerical methods, theoretical analysis of the influence of calibration errors on the reconstruction of the rotation, translation and normal vector is impossible. This theoretical analysis is extremely useful to prove the robustness of vision-based control laws [Malis 99, Malis 01a, Malis 02f]. This is the reason why we have explicitly computed the analytical solutions of the decomposition :

$$\mathbf{R}_a = f_r(\mathbf{H}) \tag{3.8}$$
$$\mathbf{t}_a = f_t(\mathbf{H}) \tag{3.9}$$
$$\mathbf{n}_a = f_n(\mathbf{H}) \tag{3.10}$$

The exact expressions of the functions $f_r(\mathbf{H})$, $f_t(\mathbf{H})$ and $f_n(\mathbf{H})$ are too long to be presented here and can be found in [Malis 07b]. The explicit analytical expression of one solution of the decomposition as a function of the other solution can also be obtained :

$$\mathbf{R}_b = \mathbf{R}_a + \frac{2}{\rho^2}\left[\nu\,\mathbf{t}_a\,\mathbf{n}_a^\top - \mathbf{t}_a\mathbf{t}_a^\top\mathbf{R}_a - \|\mathbf{t}_a\|^2\,\mathbf{R}_a\,\mathbf{n}_a\mathbf{n}_a^\top - 2\,\mathbf{R}_a\,\mathbf{n}_a\mathbf{t}_a^\top\mathbf{R}_a\right] \tag{3.11}$$
$$\mathbf{t}_b = \frac{\|\mathbf{t}_a\|}{\rho}\,\mathbf{R}_a\,\left(2\,\mathbf{n}_a + \mathbf{R}_a^\top\mathbf{t}_a\right) \tag{3.12}$$
$$\mathbf{n}_b = \frac{1}{\rho}\left(\|\mathbf{t}_a\|\,\mathbf{n}_a + \frac{2}{\|\mathbf{t}_a\|}\mathbf{R}_a^\top\mathbf{t}_a\right) \tag{3.13}$$

where $\rho = \|2\,\mathbf{n}_a + \mathbf{R}_a^\top\mathbf{t}_a\|$ and $\nu = 2\,(\mathbf{n}_a^\top\mathbf{R}_a^\top\mathbf{t}_a + 1)$. Again, these relationships are extremely useful for the design of vision-based control laws. For example, in [Vargas 05] we found that by mixing both solutions of the decomposition in an appropriate control law it is possible to converge into a particular camera configuration where a distinction can be made between the true and the false solution.

Localization (pose reconstruction)

A common problem is to reconstruct the pose when the camera parameters and the structure are known :

$$\boldsymbol{\pi}(\overline{\boldsymbol{\kappa}}, \boldsymbol{\tau}(\mathbf{T}_i, \overline{\mathbf{m}}_j)) = \overline{\mathbf{p}}_{ij} \tag{3.14}$$

This problem is of particular interest in robotic applications since the pose can be directly used to control the robot. There are two classes of methods to solve this problem : iterative and non-iterative methods.

Iterative methods minimize an appropriate cost function (e.g. least-squares error) built from the equations [Lowe 91, Kumar 94b, Dementhon 95, Horaud 97, Lu 00]. They are very accurate but they need an initial estimation of the pose to avoid local minima of the cost function.

Non-iterative methods are generally based on a "linearization" (i.e. lifting the problem to a higher dimensional space) of the equations [Dhome 89, Fiore 01, Ansar 03, Moreno 07]. The accuracy of these algorithms is generally lower since the non-linear constraints on the rotation are not directly enforced. However, they can be used to initialize the iterative algorithms.

Mapping (structure reconstruction)

If the displacements between the images are known the problem becomes simpler since only the structure must be recovered. This problem is called "structure from known motion". The system of non-linear equations to be solved becomes :

$$\boldsymbol{\pi}(\overline{\boldsymbol{\kappa}}, \boldsymbol{\tau}(\overline{\mathbf{T}}_i, \mathbf{m}_j)) = \overline{\mathbf{p}}_{ij} \tag{3.15}$$

This is the problem we have to solve when we consider a calibrated stereo system. The same problem appears when we can measure the pose (or integrate the velocity) of a well calibrated robot. Several approaches have been proposed [Matthies 89], [Smith 94], [Chaumette 96], [De Luca 07b].

Note that the problem can also be "easily" solved without completely knowing the pose. Indeed, it is well-known that the affine reconstruction from perspective images is very easy if the motion of the camera between two images is a pure translation [Moons 96]. The direction of the translation can be arbitrary and unknown (if the amplitude of the translation is also unknown the mapping will be recovered up to an unknown scale factor). This problem can be called "structure from partially known motion". The translation can be performed by a perfectly calibrated robot. However, if the robot is not well calibrated or mobile (i.e. with unavoidable drifting) it is not possible to perform an open-loop pure translation. Thus, we have proposed in [Malis 03d] an active affine reconstruction method in which the pure translation is performed closing the loop with a 2.5 D visual servoing technique [Malis 99]. Note also that for an affine reconstruction the camera does not need to be calibrated.

3.1.3.3 Self-calibration of the vision system

Self-calibration is even more difficult than SLAM since the camera intrinsic parameters are also unknowns. The problem is to solve the following non-linear system of equations :

$$\boldsymbol{\pi}(\boldsymbol{\kappa}_i, \boldsymbol{\tau}(\mathbf{T}_i, \mathbf{m}_j)) = \overline{\mathbf{p}}_{ij}$$

Since the camera intrinsic parameters are unknowns, we cannot compute the "pseudo-inverse" of the function $\boldsymbol{\pi}$. This problem is generally not well-posed and a sufficiently rich motion of the camera must be supposed (camera rotations are needed). Constraints on the camera intrinsic parameters must be added if they are not constant. Bundle adjustment approaches generalize well to this problem [Triggs 00]. However, it is generally more difficult to find an initialization for the optimization algorithms.

The self-calibration problem becomes easier if we consider pinhole cameras. In this case, it is possible to factor out the camera intrinsic parameters to obtain $\mathbf{m}_{ij} \propto \mathbf{K}_i^{-1}\underline{\overline{\mathbf{p}}}_{ij}$. Thus we obtain an uncalibrated equation, equivalent to equation (3.5), for pinhole cameras :

$$\underline{\overline{\mathbf{p}}}_{ij}^{\top} {}^i\mathbf{F}_k \underline{\overline{\mathbf{p}}}_{kj} = 0 \tag{3.16}$$

where ${}^i\mathbf{F}_k = \widehat{\mathbf{K}}_i^{-\top} {}^i\mathbf{E}_k \widehat{\mathbf{K}}_k^{-1}$ is the uncalibrated essential matrix (generally called the fundamental matrix [Faugeras 93]). Similarly, we can obtain the uncalibrated equivalent of the virtual parallax in equation (3.6) :

$$\underline{\overline{\mathbf{p}}}_{ij} \propto {}^i\mathbf{G}_k \underline{\overline{\mathbf{p}}}_{kj} + \beta_j {}^i\mathbf{e}_k \tag{3.17}$$

where ${}^i\mathbf{G}_k = \widehat{\mathbf{K}}_i {}^i\mathbf{H}_k \widehat{\mathbf{K}}_k^{-1}$ is an uncalibrated homography and ${}^i\mathbf{e}_k = \widehat{\mathbf{K}}_i {}^i\mathbf{t}_k$ is the epipole.

Self-calibration methods of pinhole cameras are generally based on the recovery of the uncalibrated essential matrices for non-planar scenes or on the recovery of the uncalibrated homography matrices for planar scenes [Hartley 00].

An example of a self-calibration method based on the estimation of the uncalibrated essential matrices is given in the next section. Then, a more detailed description of our contributions to the self-calibration from uncalibrated homographies is discussed.

3.1.3.3.1 Non planar structure

After estimating the uncalibrated essential matrices [Faugeras 93] the self-calibration problem is solved by imposing some constraints [Triggs 97, Mendonça 99, Pollefeys 99].

The idea behind the method proposed in [Mendonça 99] inspired us to propose a similar constraint for the self-calibration from homographies. Suppose we start from approximated values of $\widehat{\mathbf{K}}_i$ to compute the calibrated essential matrices :

$${}^i\widehat{\mathbf{E}}_k = \widehat{\mathbf{K}}_i^{\top} {}^i\widehat{\mathbf{F}}_k \widehat{\mathbf{K}}_k$$

The singular values of the matrices are $\text{svd}({}^i\widehat{\mathbf{E}}_k) = (\widehat{\sigma}'_{ik}, \widehat{\sigma}''_{ik}, 0)$. They should be equal $\widehat{\sigma}'_{ik} = \widehat{\sigma}''_{ik}$. However, this is not the case if the estimated $\widehat{\mathbf{K}}_i$ are not correct. Thus, we can build a cost function to minimize the "difference" between the two singular values.

3.1.3.3.2 Planar structure

For pinhole cameras, the uncalibrated homography matrix ${}^i\mathbf{G}_k$ links two corresponding points in the i-th and k-th images :

$$\overline{\mathbf{p}}_i \propto {}^i\mathbf{G}_k \overline{\mathbf{p}}_k \tag{3.18}$$

where ${}^i\mathbf{G}_k = \mathbf{K}_i \, {}^i\mathbf{H}_k \mathbf{K}_k^{-1}$. If we have enough information (at least 4 corresponding points for each image pair), then we can estimate ${}^i\widehat{\mathbf{G}}_k$ for each couple of images i and k (obviously ${}^i\widehat{\mathbf{G}}_k = \mathbf{I}$ when $i = k$). If the ${}^i\widehat{\mathbf{G}}_k$ were correctly estimated they should satisfy the constraints :

$${}^i\widehat{\mathbf{G}}_k \propto {}^i\widehat{\mathbf{G}}_j \, {}^j\widehat{\mathbf{G}}_k \tag{3.19}$$

This is not the case in general due to image noise and errors in features extraction and matching. Plane-based self-calibration methods like the one proposed by [Triggs 98] suffer from these errors. In order to reduce their influence, we proposed imposing the constraints between uncalibrated homographies using an iterative method [Malis 00b]. By imposing the constraints, we obtained an accurate self-calibration from planar scenes with an unknown metric structure. We do not use any key image but all the images are equally treated, averaging the uncertainty over all of them. Assuming that the principal point is known, the method can be applied for the self-calibration of a camera with varying focal length [Malis 00c]. The global approach has been described in [Malis 02g] and it is based on the following idea. Suppose we start from approximated values of $\widehat{\mathbf{K}}_i$ to compute the calibrated homography matrices :

$${}^i\widehat{\mathbf{H}}_k \propto \widehat{\mathbf{K}}_i^{-1} \, {}^i\widehat{\mathbf{G}}_k \, \widehat{\mathbf{K}}_k \tag{3.20}$$

Each estimated calibrated-homography matrix ${}^i\widehat{\mathbf{G}}_k$ can be decomposed to find the normal vector $\widehat{\mathbf{n}}_k$. These vectors should be such that the matrices ${}^i\widehat{\mathbf{H}}_k \, [\widehat{\mathbf{n}}_k]_\times$ have two equal singular values. This is not the case if the $\widehat{\mathbf{K}}_i$ are not correct. Thus, we can build an error to be minimized. Figure 3.2 shows four images of a planar structure used to test our self-calibration algorithm. The images were taken with the same camera calibrated in Section 3.1.3.1 with 26 images of a calibration grid. The self-calibrated parameters were $f = 678$ and $(u_0, v_0) = (355, 216)$ while the calibrated parameters were $f = 685$ and $(u_0, v_0) = (322, 229)$. Therefore, we have 1% error on the measure of the focal length, which is a satisfactory result considering that we used four images only. Once we have estimated the intrinsic parameters we can also recover the poses and the normal vector.

Image 1 Image 2 Image 3 Image 4

FIG. 3.2 – Images of the façade of St. Catharine's College, Cambridge (UK).

3.2 Direct methods

Direct methods perform data association and parameter estimation simultaneously. The definition of direct methods given in [Irani 00] states that all pixels in the image (or in an area of interest) must be involved in the estimation process. However, I do not believe that this is important and I use a different definition of direct methods that allows pixels selection. An example of pixel selection is to choose non-homogeneous regions of the image and to discard homogeneous ones. Direct methods are more accurate than feature-based methods not only because they use much more information but also because they avoid the extraction of a set of distinct features from each image separately. Thus, direct methods can be used for images that do not contain distinctive features : only sufficient image gradient along different directions is needed. In addition, direct methods allow us to explicitly "a priori" enforce structural constraints that feature based methods try to impose "a posteriori".

Parametric estimation with direct methods is generally defined as an optimization problem, and is composed of three steps :

- register (warp) the current image into the reference frame with current parameters. The modeling of an appropriate warping function is then fundamental ;
- compute the "difference" between the reference and the warped images. Then, compute an appropriate cost function (e.g. the sum of squared differences (SSD)) ;
- compute an increment of the parameters that decreases the cost function and update the current parameters.

The modeling of the warping function, the choice of the cost function and the design of efficient numerical methods to minimize the cost function are central issues. In particular, direct methods are limited by the performances of numerical optimization methods. Indeed, in order to meet real-time requirements we generally use optimization methods that have a limited convergence domain (the computational cost of global optimization methods is generally too high). For this reason, direct methods are believed to fall into the class of incremental approaches, and to be only applicable when the difference between two images is small enough.

The following two sections detail our research work on these topics. In particular, if all the parameters are unknowns, only the image registration problem is considered. Solving the image registration problem is equivalent to the estimation of the epipolar geometry. On the other hand, if the camera parameters are known we can directly perform localization and/or mapping in the Euclidean space. For both problems, the ESM described in Chapter 2 can be applied to improve the performances (convergence rate and convergence domain) of incremental direct methods. Even if the ESM is shown to outperform standard local optimization methods, it remains a local method. If the initial estimate of the parameters is too different from the true solution any local method will fail. For this reason, we are currently studying automatic real-time initialization and re-initialization techniques. Promising results have been obtained in [Benhimane 08].

3.2.1 Incremental image registration

Image registration of rigid and deformable surfaces is an active field of research. It has many applications for example in medical imagery, augmented reality and robotics. To solve this problem, several approaches have been proposed in the literature [Brown 92, Zitova 03]. In this document, I focus on iterative direct methods that are potentially real-time since we are interested in robotic applications, such as visual servoing. Additionally, I will not consider methods that rely on an off-line learning step like [Gleicher 97, Cootes 98, Black 98, La Cascia 00, Fleet 00].

Direct approaches minimize a similarity measure between a reference template and a region of the current image warped with appropriate geometric and photometric parameters. The underlying assumption of incremental direct methods is that the deformations between two views of the surface are sufficiently small. This will typically be the case in video sequences or after a correct initialization. In this document, methods based on optical flow [Horn 81] [Horn 88] are not considered since they suppose too small a displacement between two images.

Figure 3.2.1 shows an example for a monocular camera. The reference template $\boldsymbol{\mathcal{T}}$ can be the whole image $\boldsymbol{\mathcal{I}}_0$. All possible unknown parameters are in $\boldsymbol{x}$. Once we find the optimal parameters $\overline{\boldsymbol{x}}_1$ to align the image $\boldsymbol{\mathcal{I}}_1$ with the reference template $\boldsymbol{\mathcal{T}}$ we look for the incremental parameters $\boldsymbol{x}$ to align the image $\boldsymbol{\mathcal{I}}_2$ with $\boldsymbol{\mathcal{T}}$, and so on. In some applications, like SLAM in large scale environments the part of the scene observed in the reference template may be no longer visible. In this case, a new reference template must be selected (see Section 3.2.2).

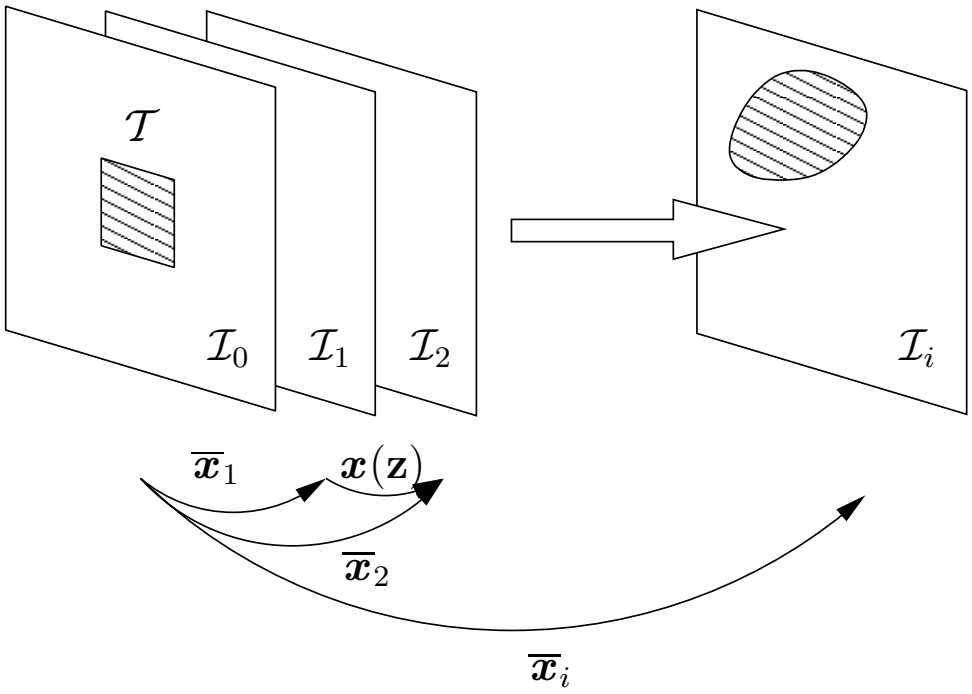


FIG. 3.3 – Incremental image registration.

Initial work on incremental direct image registration supposed that the scene was planar and parallel to the image plane [Lucas 81], and that the observed surface was Lambertian. Then all possible motions of a plane and more complex object shapes were considered [Bergen 92, Kumar 94a, Szeliski 95, Irani 02]. These methods generally minimize the sum of squared differences (SSD) using standard iterative optimization methods like the Gauss-Newton or the Levemberg-Marquardt methods. The same optimization approach can be used for the direct image registration of deformable surfaces [Bartoli 04].

One important step towards real-time applications with a fast frame rate has been improving the efficiency of the standard optimization method. Two approaches are possible for building efficient algorithms. The first one is to keep the same convergence rate (the number of iterations needed to obtain the minimum of the similarity measure) while reducing the computational cost per iteration. This can be achieved by pre-computing partially [Hager 98] or completely [Baker 01] the Jacobian used in the minimization. The main limitation of these approaches is that they can only be applied to certain classes of warp.

An alternative approach for building efficient algorithms is to keep the same computational cost per iteration while increasing the convergence rate. This can be achieved for example by using the Efficient Second-order approximation Method described in Chapter 2. This approach was initially applied to the estimation of a homography for the visual tracking of planar surfaces with a perspective camera in [Benhimane 04]. Then, we extended our approach to the image registration of regular rigid or deformable surfaces [Malis 07a]. In order to cope with occlusions and specularities we used in [Malis 07a] the robust optimization techniques presented in Section 2. Despite improved results, robust optimization techniques tend to decrease the efficiency. Thus, we have proposed in [Silveira 07b] to model the illumination changes. The proposed approach is generic : it does not require either the characteristics of the light sources (e.g. number, power, pose), or the reflectance properties of the surface.

If we compare the ESM to existing optimization techniques, a greater efficiency is obtained by reducing the number of iterations needed to converge to the minimum of the similarity measure. The convergence domain is also larger than the domains of standard optimization techniques. Therefore, the ESM algorithm is particularly adapted to image registration. For example, we showed in [Vercauteren 07] that the ESM theory provides interesting theoretical roots to the different variants of Thirion's demons algorithm [Thirion 98]. The demons algorithm proposed by Thirion is one of the most efficient methods for non-rigid image registration and it is related to gradient descent optimization [Pennec 99]. One of the main results of the theoretical analysis in [Vercauteren 07] was to show that the symmetric forces variants of the algorithm [Thirion 98, Wang 05, Rogel 06] are related to the ESM scheme. This study thus explains why, from a theoretical point of view, the symmetric forces demons algorithm is more efficient in practice.

3.2.1.1 Generic warping model

For the sake of simplicity, consider in this section that the observed scene is Lambertian and static with respect to the light sources. Thus, the photometric parame-

ters $\boldsymbol{\phi}_i$ are constant and we can set $\boldsymbol{\mathcal{I}}_i(\mathbf{p}) = \boldsymbol{\mathcal{I}}(\boldsymbol{\phi}_i, \mathbf{p})$. The more general case of non-Lambertian scenes will be discussed in the next section. Consider a generic warping function $\mathbf{w} : \mathbb{R}^2 \mapsto \mathbb{R}^2$ which transforms the coordinates $\overline{\mathbf{p}}_{0j}$ of the j-th pixel in the reference template into the current image coordinates :

$$\mathbf{p}_{ij} = \mathbf{w}(\boldsymbol{g}_i, \overline{\mathbf{p}}_{0j})$$

where $\boldsymbol{g}_i \in G$ (G being a Lie group) contains the intrinsic camera parameters, the pose of the camera and the parameters describing the structure of the scene. This function is detailed in [Malis 07a]. The structure of the scene is described by a small number of parameters (this number obviously depends on the complexity of the scene). For example, regular surfaces can be approximated using the Radial Basis Functions (RBF) [Carr 97] (see Section 3.2.1.2).

Let me set $\boldsymbol{x}_i = \boldsymbol{g}_i$ and suppose that we have an initial approximation $\widehat{\boldsymbol{x}}_i$ of the true parameters $\overline{\boldsymbol{x}}_i$. Let $\mathbf{z}$ be the vector which contains the coordinates of the Lie algebra of G. Thus, $\mathbf{z}$ is a local parametrization of G and, via the exponential map, $\boldsymbol{x}(\mathbf{z}) \in G$. Using the same notation as Chapter 2, the incremental image registration problem can be formulated as the solution of the following non-linear system defined on the Lie group G :

$$f_k(\widehat{\boldsymbol{x}}_i \cdot \boldsymbol{x}(\mathbf{z})) = \boldsymbol{\mathcal{I}}_i(\mathbf{w}(\widehat{\boldsymbol{x}}_i \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_{0j})) - \overline{\boldsymbol{\mathcal{I}}}_{0j} = 0$$

where $k = \{1, 2, \ldots, s_u s_v\}$ ($(s_u \times s_v)$ being the size of the reference image or template). The coordinates $\overline{\mathbf{p}}_{0j}$ of the pixels of the reference template are known and constant. We suppose that $\overline{\boldsymbol{x}}_i$ exist such that the current image $\boldsymbol{\mathcal{I}}_i$ can be registered with the reference image $\boldsymbol{\mathcal{I}}_i(\mathbf{w}(\overline{\boldsymbol{x}}, \overline{\mathbf{p}}_{0j})) = \overline{\boldsymbol{\mathcal{I}}}_{0j}$. Finally, let me name $\widetilde{\mathbf{z}}$ the optimal increment such that we obtain the exact solution of the problem : $\boldsymbol{x}(\widetilde{\mathbf{z}}) = \widehat{\boldsymbol{x}}^{-1} \cdot \overline{\boldsymbol{x}} = \widetilde{\boldsymbol{x}}$.

3.2.1.1.1 Computing the Jacobians for the ESM

In order to apply the ESM (see equation (2.91)), we need to compute the Jacobian $\boldsymbol{\mathcal{J}}(\widehat{\boldsymbol{x}})$ (called the current Jacobian) :

$$\boldsymbol{\mathcal{J}}(\widehat{\boldsymbol{x}}) = \nabla \boldsymbol{f}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}))|_{\mathbf{z}=0}$$

and the Jacobian $\boldsymbol{\mathcal{J}}(\overline{\boldsymbol{x}})$ (called the reference Jacobian) :

$$\boldsymbol{\mathcal{J}}(\overline{\boldsymbol{x}}) = \nabla \boldsymbol{f}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}))|_{\mathbf{z}=\widetilde{\mathbf{z}}}$$

Without loss of generality, only the j-th row of the current and reference Jacobians is computed. The full-sized Jacobians can easily be obtained by stacking all the rows together. In the general case, a small part of the reference Jacobian must be approximated. However, the experimental results demonstrate that even with this approximation the ESM still has superior performance with respect to standard numeric methods. On the other hand, in the particular case when the warping function is a group action of G on $\mathbb{R}^2$ on the left, the reference Jacobian can be completely computed from image data. This is the case for example when we track a planar object with an uncalibrated pinhole camera

($\boldsymbol{x} \in \mathbb{SL}(3)$ represents a homography) [Benhimane 04]. Another example is when we track rigid objects with a calibrated stereo pair ($\boldsymbol{x} \in \mathbb{SE}(3)$ represent the displacement of the stereo pair) [Comport 07].

Current Jacobian

The j-th row of the current Jacobian is :

$$\boldsymbol{\mathcal{J}}_k(\widehat{\boldsymbol{x}}) = \nabla f_k(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}))|_{\mathbf{z}=0} = \nabla \boldsymbol{\mathcal{I}}(\mathbf{w}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j))|_{\mathbf{z}=0}$$

Let me rewrite the function $\boldsymbol{\mathcal{I}}$ as follows :

$$\boldsymbol{\mathcal{I}}(\mathbf{w}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}})) = \boldsymbol{\mathcal{I}}(\mathbf{w}(\widehat{\boldsymbol{x}}, \mathbf{w}^{-1}(\widehat{\boldsymbol{x}}, \mathbf{w}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j))))$$

Thus, using the chain rules for derivatives we obtain :

$$\left.\frac{\partial \boldsymbol{\mathcal{I}}(\mathbf{w}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j))}{\partial \mathbf{z}}\right|_{\mathbf{z}=0} = \left.\frac{\partial \boldsymbol{\mathcal{I}}(\mathbf{w}(\widehat{\boldsymbol{x}}, \mathbf{q}))}{\partial \mathbf{q}}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j} \left.\frac{\partial \mathbf{w}^{-1}(\widehat{\boldsymbol{x}}, \mathbf{w}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j))}{\partial \mathbf{z}}\right|_{\mathbf{z}=0}$$

It is important to note that the first part of the derivative is a (1×2) vector that can be computed directly from the current image data. The vector contains the gradient of the warped image (i.e. the image being warped with $\mathbf{w}(\widehat{\boldsymbol{x}})$) computed at $\overline{\mathbf{p}}_j$. The second part of the derivative can be decomposed into two parts :

$$\left.\frac{\partial \mathbf{w}^{-1}(\widehat{\boldsymbol{x}}, \mathbf{w}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j))}{\partial \mathbf{z}}\right|_{\mathbf{z}=0} = \left.\frac{\partial \mathbf{w}^{-1}(\widehat{\boldsymbol{x}}, \mathbf{q})}{\partial \mathbf{q}}\right|_{\mathbf{q}=\mathbf{w}(\widehat{\boldsymbol{x}}, \overline{\mathbf{p}}_j)} \left.\frac{\partial \mathbf{w}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}), \mathbf{p})}{\partial \mathbf{z}}\right|_{\mathbf{z}=0}$$

The first part is :

$$\left.\frac{\partial \mathbf{w}^{-1}(\widehat{\boldsymbol{x}}, \mathbf{q})}{\partial \mathbf{q}}\right|_{\mathbf{q}=\mathbf{w}(\widehat{\boldsymbol{x}}, \mathbf{p})} = \left(\left.\frac{\partial \mathbf{w}(\widehat{\boldsymbol{x}}, \mathbf{q})}{\partial \mathbf{q}}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j}\right)^{-1}$$

and the second part is :

$$\left.\frac{\partial \mathbf{w}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j)}{\partial \mathbf{z}}\right|_{\mathbf{z}=0} = \left.\frac{\partial \mathbf{w}(\boldsymbol{x}, \overline{\mathbf{p}}_j)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\widehat{\boldsymbol{x}}} \left.\frac{\partial \widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=0}$$

where

$$\left.\frac{\partial \widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=0} = \left.\frac{\partial \widehat{\boldsymbol{x}} \cdot \boldsymbol{x}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{e}} \left.\frac{\partial \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=0}$$

Finally, the j-th row of the current Jacobian can be written as follows :

$$\boldsymbol{\mathcal{J}}_k(\widehat{\boldsymbol{x}}) = \left.\frac{\partial \boldsymbol{\mathcal{I}}(\mathbf{w}(\widehat{\boldsymbol{x}}, \mathbf{q}))}{\partial \mathbf{q}}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j} \left(\left.\frac{\partial \mathbf{w}(\widehat{\boldsymbol{x}}, \mathbf{q})}{\partial \mathbf{q}}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j}\right)^{-1} \left.\frac{\partial \mathbf{w}(\boldsymbol{x}, \overline{\mathbf{p}}_j)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\widehat{\boldsymbol{x}}} \left.\frac{\partial \widehat{\boldsymbol{x}} \cdot \boldsymbol{x}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{e}} \left.\frac{\partial \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=0} \tag{3.21}$$

Therefore, the current Jacobian can be computed from image data only.

Reference Jacobian

The j-th row of the reference Jacobian is :

$$\boldsymbol{\mathcal{J}}_k(\overline{\boldsymbol{x}}) = \nabla f_i(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}))|_{\mathbf{z}=\widetilde{\mathbf{z}}} = \nabla \boldsymbol{\mathcal{I}}(\mathbf{w}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j))|_{\mathbf{z}=\widetilde{\mathbf{z}}}$$

Let me rewrite the function as follows :

$$\boldsymbol{\mathcal{I}}(\mathbf{w}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j)) = \boldsymbol{\mathcal{I}}(\mathbf{w}(\overline{\boldsymbol{x}}, \mathbf{w}^{-1}(\overline{\boldsymbol{x}}, \mathbf{w}(\overline{\boldsymbol{x}} \cdot \widetilde{\boldsymbol{x}}^{-1} \cdot \boldsymbol{x}(\mathbf{z}), \mathbf{p}))))$$

As for the current Jacobian, the derivative can be decomposed into two parts :

$$\left.\frac{\partial \boldsymbol{\mathcal{I}}(\mathbf{w}(\widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j))}{\partial \mathbf{z}}\right|_{\mathbf{z}=\widetilde{\mathbf{z}}} = \left.\frac{\partial \boldsymbol{\mathcal{I}}(\mathbf{w}(\overline{\boldsymbol{x}}, \mathbf{q}))}{\partial \mathbf{q}}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j} \left.\frac{\partial \mathbf{w}^{-1}(\overline{\boldsymbol{x}}, \mathbf{w}(\overline{\boldsymbol{x}} \cdot \widetilde{\boldsymbol{x}}^{-1} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j))}{\partial \mathbf{z}}\right|_{\mathbf{z}=\widetilde{\mathbf{z}}}$$

Again, the first part of the Jacobian is a (1×2) vector that can be computed directly from the reference image data. It is very important to highlight that the gradient of the reference image can be computed without explicitly knowing the true solution $\overline{\mathbf{x}}$. The the second part of the derivative is again decomposed into two parts :

$$\left.\frac{\partial \mathbf{w}^{-1}(\overline{\boldsymbol{x}}, \mathbf{w}(\overline{\boldsymbol{x}} \cdot \widetilde{\boldsymbol{x}}^{-1} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j))}{\partial \mathbf{z}}\right|_{\mathbf{z}=\widetilde{\mathbf{z}}} = \left.\frac{\partial \mathbf{w}^{-1}(\overline{\boldsymbol{x}}, \mathbf{q})}{\partial \mathbf{q}}\right|_{\mathbf{q}=\mathbf{w}(\overline{\boldsymbol{x}}, \overline{\mathbf{p}}_j)} \left.\frac{\partial \mathbf{w}(\overline{\boldsymbol{x}} \cdot \widetilde{\boldsymbol{x}}^{-1} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j)}{\partial \mathbf{z}}\right|_{\mathbf{z}=\widetilde{\mathbf{z}}}$$

The first part is :

$$\left.\frac{\partial \mathbf{w}^{-1}(\overline{\boldsymbol{x}}, \mathbf{q})}{\partial \mathbf{q}}\right|_{\mathbf{q}=\mathbf{w}(\overline{\boldsymbol{x}}, \overline{\mathbf{p}}_j)} = \left.\left(\frac{\partial \mathbf{w}(\overline{\boldsymbol{x}}, \mathbf{q})}{\partial \mathbf{q}}\right)^{-1}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j}$$

and the second part is :

$$\left.\frac{\partial \mathbf{w}(\overline{\boldsymbol{x}} \cdot \widetilde{\boldsymbol{x}}^{-1} \cdot \boldsymbol{x}(\mathbf{z}), \overline{\mathbf{p}}_j)}{\partial \mathbf{z}}\right|_{\mathbf{z}=\widetilde{\mathbf{z}}} = \left.\frac{\partial \mathbf{w}(\boldsymbol{x}, \overline{\mathbf{p}}_j)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\overline{\boldsymbol{x}}} \left.\frac{\partial \overline{\boldsymbol{x}} \cdot \widetilde{\boldsymbol{x}}^{-1} \cdot \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=\widetilde{\mathbf{z}}}$$

where

$$\left.\frac{\partial \widehat{\boldsymbol{x}} \cdot \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=\widetilde{\mathbf{z}}} = \left.\frac{\partial \overline{\boldsymbol{x}} \cdot \boldsymbol{x}}{\partial \boldsymbol{x}}\right|_{x=e} \left.\frac{\partial \widetilde{\boldsymbol{x}}^{-1} \cdot \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=\widetilde{\mathbf{z}}}$$

Due to the parametrization of the increment by using Lie Algebra, we have the following formula :

$$\left.\frac{\partial \widetilde{\boldsymbol{x}}^{-1} \cdot \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=\widetilde{\mathbf{z}}} \widetilde{\mathbf{z}} = \left.\frac{\partial \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=0} \widetilde{\mathbf{z}} \tag{3.22}$$

This formula shows that we do not need to know $\widetilde{\mathbf{z}}$ to compute this last Jacobian. This explains why the parametrization using Lie algebra is a key point of our algorithm. Another parametrization may imply that this part of the Jacobian matrix depends on $\widetilde{\mathbf{z}}$. As a consequence, in that case, the Jacobian is only an approximation.

Finally, the j-th row of the reference Jacobian can be written as follows :

$$\boldsymbol{\mathcal{J}}_k(\overline{\boldsymbol{x}}) = \left.\frac{\partial \boldsymbol{\mathcal{I}}(\mathbf{w}(\overline{\boldsymbol{x}},\mathbf{q}))}{\partial \mathbf{q}}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j} \left(\left.\frac{\partial \mathbf{w}(\overline{\boldsymbol{x}},\mathbf{q})}{\partial \mathbf{q}}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j}\right)^{-1} \left.\frac{\partial \mathbf{w}(\boldsymbol{x},\overline{\mathbf{p}}_j)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\overline{\boldsymbol{x}}} \left.\frac{\partial \overline{\boldsymbol{x}}\cdot\boldsymbol{x}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{e}} \left.\frac{\partial \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=0} \tag{3.23}$$

The third and fourth Jacobians generally depend on the unknown value $\overline{\boldsymbol{x}}$ of the parameters. Thus, we use their approximated value $\widehat{\boldsymbol{x}}$.

Computing the Jacobians when the warping is a group action

Let me suppose that the warping function is a group action of G on $\mathbb{R}^2$ on the left :

$$\mathbf{w}(\widehat{\boldsymbol{x}}\cdot\boldsymbol{x},\mathbf{p}_j) = \mathbf{w}(\widehat{\boldsymbol{x}},\mathbf{w}(\boldsymbol{x},\mathbf{p}_j))$$

which means that :

$$\mathbf{w}^{-1}(\widehat{\boldsymbol{x}},\mathbf{w}(\widehat{\boldsymbol{x}}\cdot\boldsymbol{x},\mathbf{p}_j)) = \mathbf{w}(\boldsymbol{x},\overline{\mathbf{p}}_j)$$

Thus, the following identity is verified :

$$\left.\frac{\partial \mathbf{w}^{-1}(\widehat{\boldsymbol{x}},\mathbf{w}(\widehat{\boldsymbol{x}}\cdot\boldsymbol{x},\overline{\mathbf{p}}_j))}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=e} = \left.\frac{\partial \mathbf{w}(\boldsymbol{x},\overline{\mathbf{p}}_j)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=e}$$

Therefore, the second, third and fourth Jacobians in equation (3.21) can be replaced by a simpler Jacobian

$$\left(\left.\frac{\partial \mathbf{w}(\widehat{\boldsymbol{x}},\mathbf{q})}{\partial \mathbf{q}}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j}\right)^{-1} \left.\frac{\partial \mathbf{w}(\boldsymbol{x},\overline{\mathbf{p}}_j)}{\partial \mathbf{z}}\right|_{\boldsymbol{x}=\widehat{\boldsymbol{x}}} \left.\frac{\partial \widehat{\boldsymbol{x}}\cdot\boldsymbol{x}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=e} = \left.\frac{\partial \mathbf{w}(\boldsymbol{x},\overline{\mathbf{p}}_j)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=e}$$

Thus, the current Jacobian becomes :

$$\boldsymbol{\mathcal{J}}_k(\widehat{\boldsymbol{x}}) = \left.\frac{\partial \boldsymbol{\mathcal{I}}(\mathbf{w}(\widehat{\boldsymbol{x}},\mathbf{q}))}{\partial \mathbf{q}}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j} \left.\frac{\partial \mathbf{w}(\boldsymbol{x},\overline{\mathbf{p}}_j)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=e} \left.\frac{\partial \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=0}$$

The same manipulation can be used for the reference Jacobian :

$$\left.\frac{\partial \mathbf{w}^{-1}(\overline{\boldsymbol{x}},\mathbf{w}(\overline{\boldsymbol{x}}\cdot\boldsymbol{x},\overline{\mathbf{p}}_j))}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=e} = \left.\frac{\partial \mathbf{w}(\boldsymbol{x},\overline{\mathbf{p}}_j)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=e}$$

Thus, the second, third and fourth Jacobians in equation (3.23) can be replaced by a simpler Jacobian

$$\left(\left.\frac{\partial \mathbf{w}(\overline{\boldsymbol{x}},\mathbf{q})}{\partial \mathbf{q}}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j}\right)^{-1} \left.\frac{\partial \mathbf{w}(\boldsymbol{x},\overline{\mathbf{p}}_j)}{\partial \mathbf{z}}\right|_{\boldsymbol{x}=\overline{\boldsymbol{x}}} \left.\frac{\partial \overline{\boldsymbol{x}}\cdot\boldsymbol{x}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=e} = \left.\frac{\partial \mathbf{w}(\boldsymbol{x},\overline{\mathbf{p}}_j)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=e}$$

Therefore, the reference Jacobian becomes :

$$\boldsymbol{\mathcal{J}}_k(\overline{\boldsymbol{x}}) = \left.\frac{\partial \boldsymbol{\mathcal{I}}(\mathbf{w}(\overline{\boldsymbol{x}},\mathbf{q}))}{\partial \mathbf{q}}\right|_{\mathbf{q}=\overline{\mathbf{p}}_j} \left.\frac{\partial \mathbf{w}(\boldsymbol{x},\overline{\mathbf{p}}_j)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=e} \left.\frac{\partial \boldsymbol{x}(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=0}$$

The reference Jacobian can now be completely computed from image data only.

The warping function is a group action when we want to register, with a pinhole camera, a Lambertian planar object that freely moves in the Cartesian space. The 8 parameters that define the transformation of the object are encoded in the (3×3) uncalibrated homography matrix $\mathbf{G} \in \mathbb{SL}(3)$. Thus, the warping function for a planar object can be obtained by setting $\boldsymbol{x}_i \equiv {}^i\mathbf{G}_0$. The current image point $\mathbf{p}_{ij}$ can be written as follows :

$$\mathbf{p}_{ij} = \mathbf{w}({}^i\mathbf{G}_0, \overline{\mathbf{p}}_{0j}) = \left(\frac{g_{11}\,\overline{u}_{0j} + g_{12}\,\overline{v}_{0j} + g_{13}}{g_{31}\,\overline{u}_{0j} + g_{32}\,\overline{v}_{0j} + g_{33}}, \frac{g_{21}\,\overline{u}_{0j} + g_{22}\,\overline{v}_{0j} + g_{23}}{g_{31}\,\overline{u}_{0j} + g_{32}\,\overline{v}_{0j} + g_{33}} \right)$$

It is easy to verify that such warping is a group action of $\mathbb{SL}(3)$ on $\mathbb{R}^2$ on the left :

$$\mathbf{w}(\widehat{\mathbf{G}}, \mathbf{w}(\mathbf{G}, \overline{\mathbf{p}}_{0j})) = \mathbf{w}(\widehat{\mathbf{G}}\,\mathbf{G}, \overline{\mathbf{p}}_{0j})$$

3.2.1.2 Generic photometric model

In [Silveira 07b], we proposed a unified method to deal with the incremental image registration problem of either Lambertian or non-Lambertian objects under shadows, inter-reflections, glints as well as ambient, diffuse and specular reflections which may vary in power, type, number and space. The method is based on the following generic model of illumination changes :

$$\boldsymbol{\mathcal{I}}'(\boldsymbol{x}_i, \overline{\mathbf{p}}_{0j}) = \widetilde{\boldsymbol{\mathcal{I}}}(\boldsymbol{\alpha}_i, \overline{\mathbf{p}}_{0j}) \cdot \boldsymbol{\mathcal{I}}_i(\mathbf{w}(\boldsymbol{g}_i, \overline{\mathbf{p}}_{0j})) + \beta_i$$

where $\boldsymbol{x}_i = (\boldsymbol{g}_i, \boldsymbol{\alpha}_i, \beta_i)$. The global illumination change is taken into account in β_i while diffuse and specular illumination changes are taken into account in $\boldsymbol{\alpha}_i$. The function $\widetilde{\boldsymbol{\mathcal{I}}}(\boldsymbol{\alpha}_i, \overline{\mathbf{p}}_{0j})$ approximate all possible illumination changes. We model the function as a parametric surface depending on a limited number of parameters (less parameters than the available equations). We have tested various techniques to model such a surface. For example, a regular surface can be approximated using the Radial Basis Functions (RBF) [Carr 97] with a thin-plate spline function γ :

$$\widetilde{\boldsymbol{\mathcal{I}}}(\boldsymbol{\alpha}, \overline{\mathbf{p}}_{0j}) = (\alpha_{c+1}, \alpha_{c+2}, \alpha_{c+3})^\top \mathbf{p} + \sum_{k=1}^{n} \alpha_k \gamma(\|\overline{\mathbf{p}}_{0j} - \mathbf{c}_k\|)$$

where $\{\mathbf{c}_k\}$ are n image points (called centers) that can be selected, for example, on a regular grid. Note that the function $\widetilde{\boldsymbol{\mathcal{I}}}$ is linear in the unknowns $\boldsymbol{\alpha}$. The use of RBFs allows the surface to be regularized but they may fail to accurately capture discontinuities. Thus, we have also tested another suitable strategy for dealing with discontinuous surfaces. In this case, $\widetilde{\boldsymbol{\mathcal{I}}}(\boldsymbol{\alpha}_i, \overline{\mathbf{p}}_{0j})$ is approximated by a discretized surface which evolves with time (see the figure 3.2.1.2).

The system of non-linear equations can now be written as follows :

$$f_k(\boldsymbol{x}) = \boldsymbol{\mathcal{I}}'(\boldsymbol{x}_i, \overline{\mathbf{p}}_{0j}) - \overline{\boldsymbol{\mathcal{I}}}_{0j} = 0$$

Again, we applied the ESM to the least-squares optimization problem. The computation of the Jacobians related to the photometric parameters is detailed in [Silveira 07b].

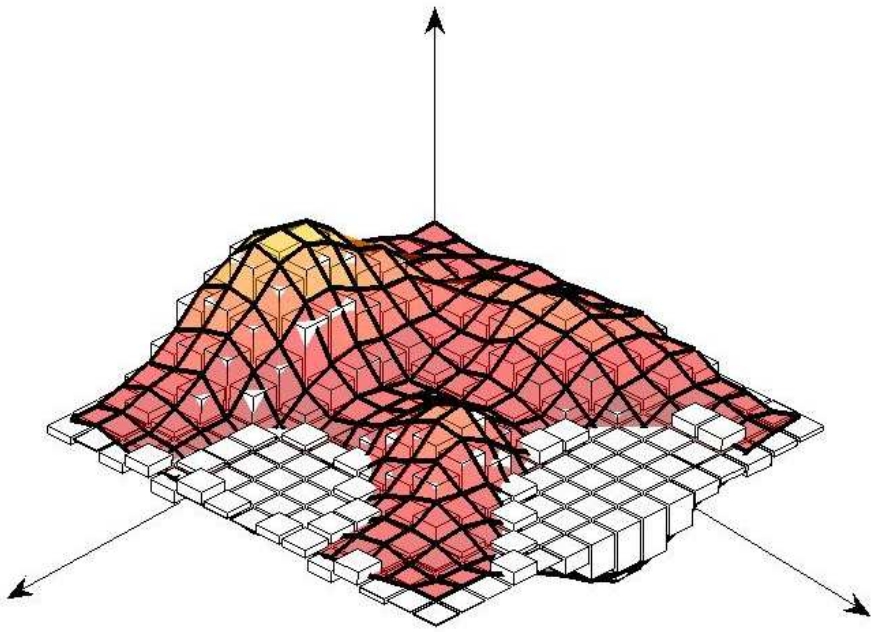


FIG. 3.4 – Approximating the surface $\widetilde{\boldsymbol{\mathcal{I}}}(\boldsymbol{\alpha}, \mathbf{p})$ placing centers on a regular grid. The discretized surface (represented by boxes) can handle discontinuities while RBFs provide a regular (smooth) approximation.

3.2.1.3 Experimental results

In this section I briefly present four sets of experiments to show the performances of our image registration approach. Figure 3.5 shows the image registration of a planar surface. The ESM allows the images acquired at 100 Hz to be aligned at run time with a reference template of size (400×200) pixels. Figure 3.6 shows the image registration under arbitrary illumination changes. Note the strong perspective and photometric deformations (shadows, specularities,...). Details of the experiments can be found in [Silveira 07b]. Figure 3.7 and Figure 3.8 show the image registration of rigid surfaces. In these experiments we used RBFs to approximate the regular surfaces. The centers are placed on the corners of the grid. The regular grid is deformed in the current image according to the surface of the objects, showing that the structure is correctly estimated. After camera self-calibration, the surface can be reconstructed up to a scale factor (see for example [Malis 07a]). Figure 3.9 and Figure 3.10 show the image registration of non rigid surfaces. More results on non-rigid image registration can be found in [Vercauteren 07]. In the present sequences, we did not apply our approach to dealing with illumination changes. Thus, the results can be considerably improved for the heart sequence since a

lot of specularities perturb the the registration.

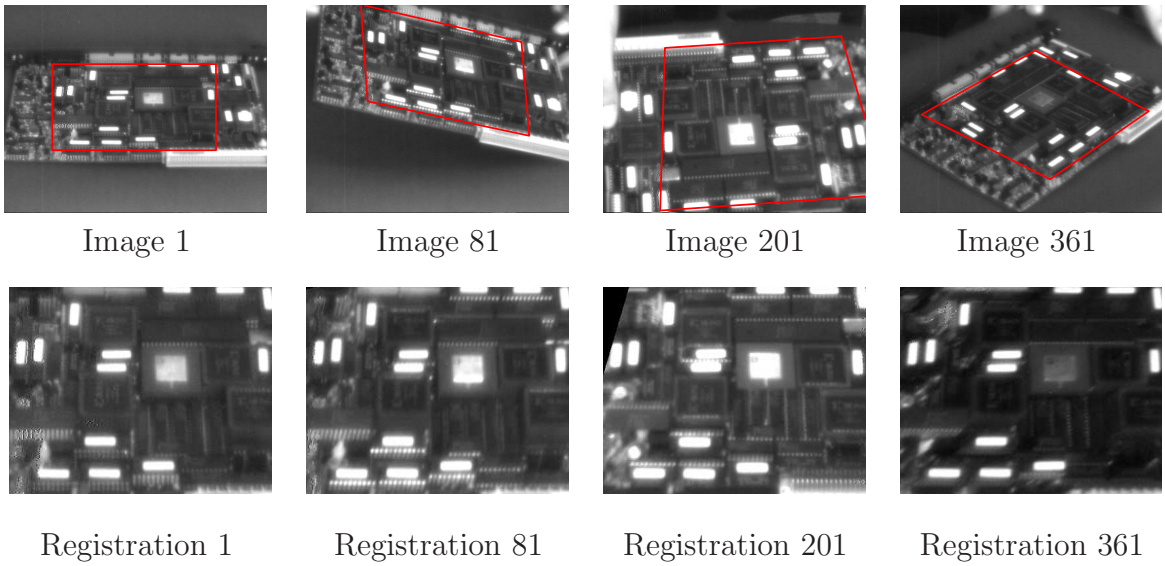


FIG. 3.5 – Image registration of a planar surface with an uncalibrated pinhole camera in a sequence of 361 images. The size of the selected template has (400× 200) pixels. The image registration with the ESM ran in real-time at 100 Hz. The bottom row shows the area of interest aligned with respect to the template.

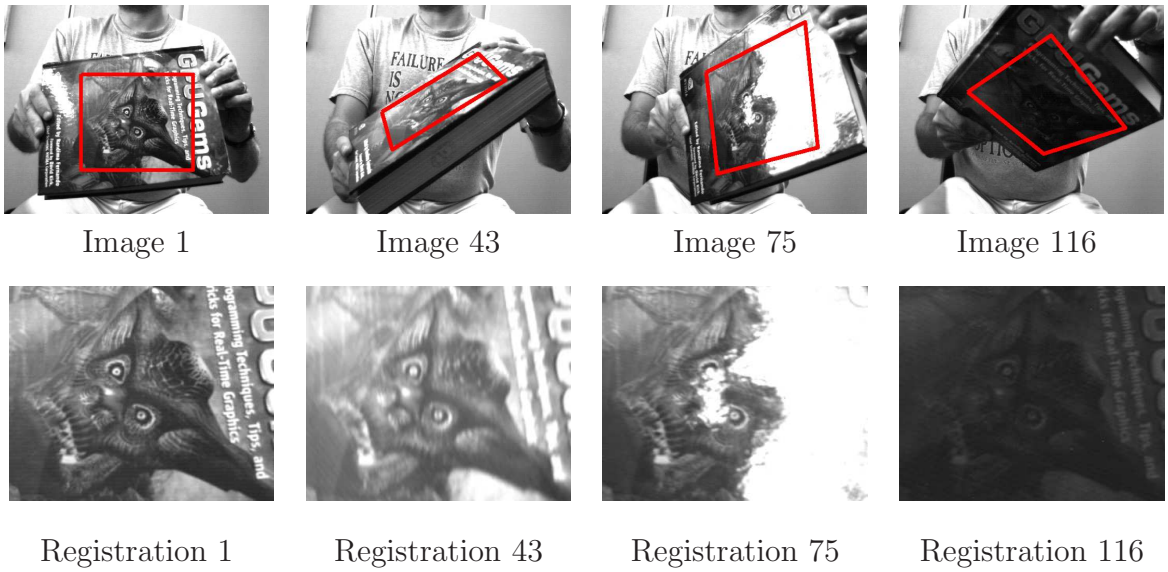


FIG. 3.6 – Image registration with an uncalibrated pinhole camera under generic illumination conditions in a sequence of 116 images. The bottom row shows the area of interest registered with respect to the template. The compensation of the illumination changes is not shown in the images.

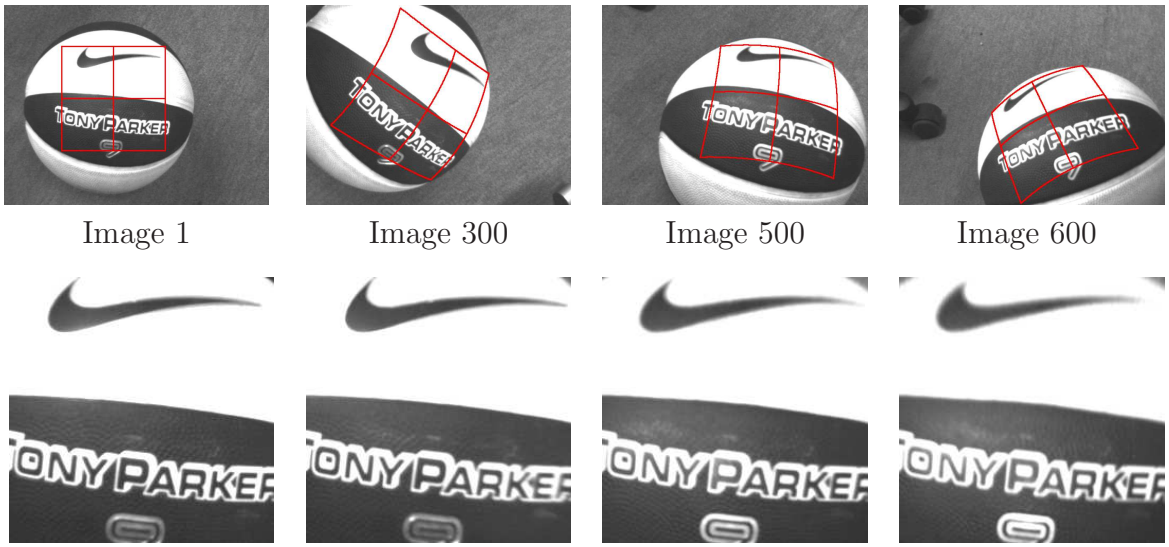


FIG. 3.7 – Image registration with an uncalibrated pinhole camera in a sequence of 600 images. The top row shows the (4×4) regular grid used to align the area of interest in the sequence. The bottom row shows the area of interest registered with respect to the template.

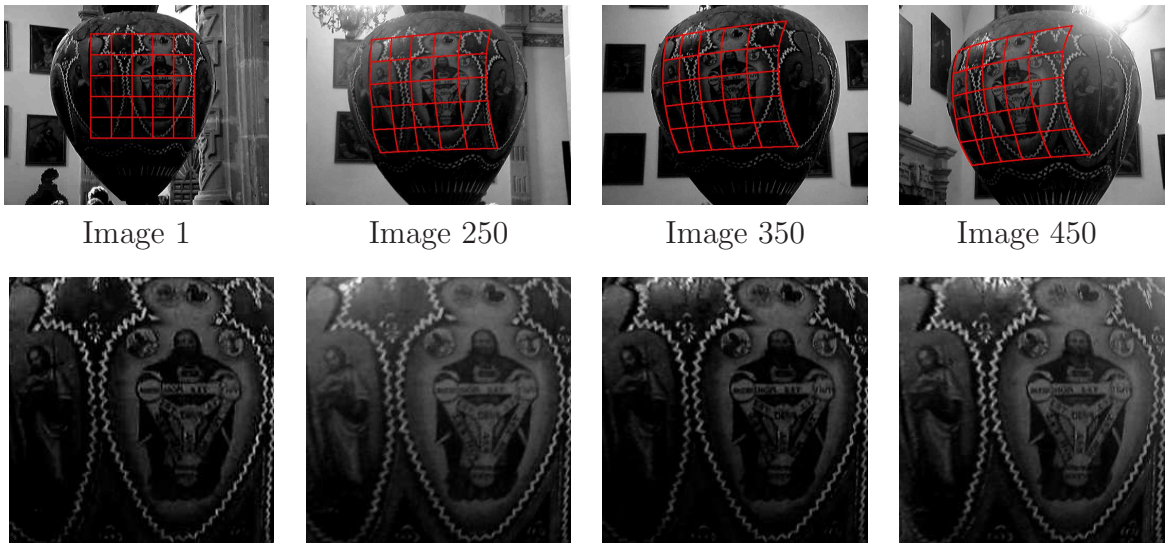


FIG. 3.8 – Image registration with an uncalibrated pinhole camera in a sequence of 450 images. The top row shows the (6×6) regular grid used to align the area of interest in the sequence. The bottom row shows the area of interest registered with respect to the template.

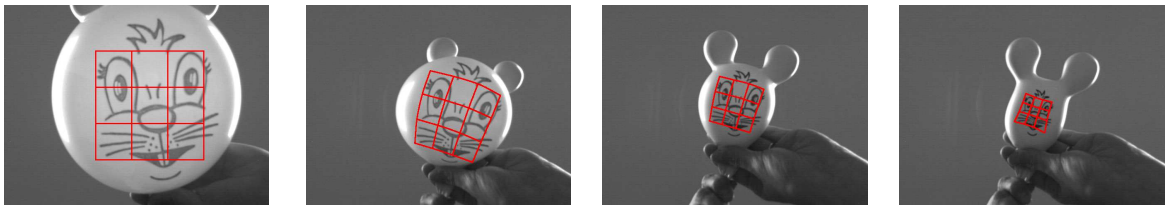
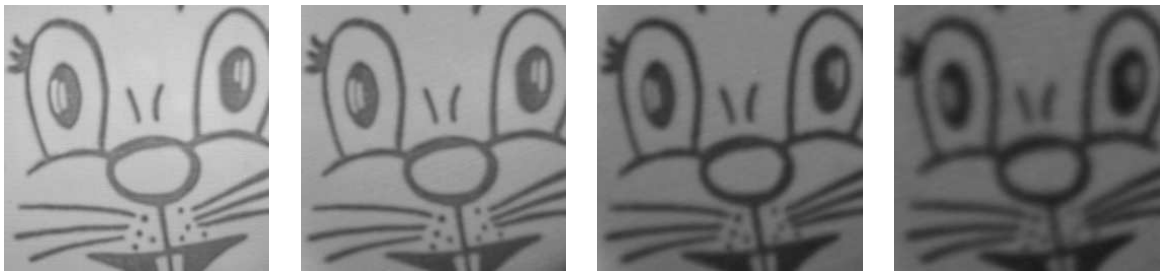


Image 1 Image 800 Image 1000 Image 1082



Registration 1 Registration 800 Registration 1000 Registration 1082

FIG. 3.9 – Image registration of a deformable surface with an uncalibrated pinhole camera in a sequence of 1082 images. The top row shows the (4×4) regular grid used to align the area of interest in the sequence. The bottom row shows the area of interest registered with respect to the template.

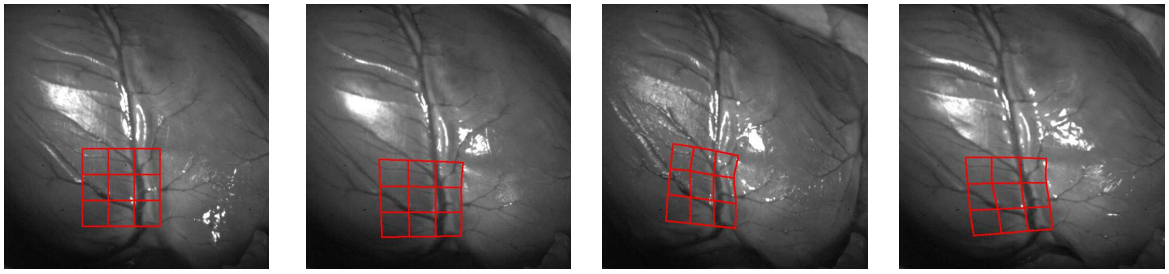
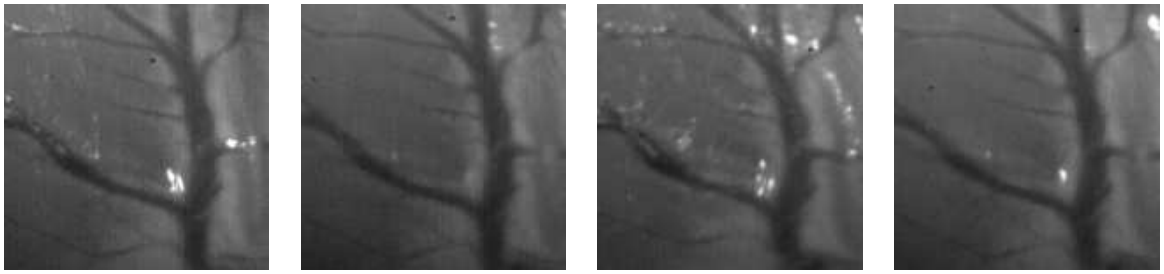


Image 1 Image 50 Image 130 Image 220



Registration 1 Registration 50 Registration 130 Registration 220

FIG. 3.10 – Image registration of a deformable surface with an uncalibrated pinhole camera.

3.2.2 Localization and/or Mapping

Our approach to visual simultaneous localization and/or mapping consists in performing direct image registration with a calibrated camera in order to recover explicitly the pose of the camera and/or structure parameters (in image registration these parameters are mixed with the camera intrinsic parameters). If the structure is known and the illumination conditions do not change, then the only unknown is the pose of the camera [Benhimane 06b]. If the structure is unknown and we observe a planar scene we can estimate a calibrated homography matrix that contains both the pose and the normal to the plane [Mei 06a]. The pose and the normal to the plane are recovered by decomposing the homography. However, if we simultaneously observe several planes rigidly linked between them, it is difficult to impose the constraint that their related homographies must contain the same pose. Thus, instead of registering each plane separately, we obtain more accurate results if we register all planes at the same time having as unknowns the pose and all the normal vectors [Mei 06c]. Note that in these last two papers we have shown how the approach can be applied to any central catadioptric camera.

When dealing with large scale environments, the part of the scene visible in the reference templates may disappears. Thus, it is necessary to regularly update/insert new reference templates. Figure 3.11 illustrates that situation. At the i-th image the region $\mathcal{R}_{02}$ disappear. A new region $\mathcal{R}_{i1}$ is included.

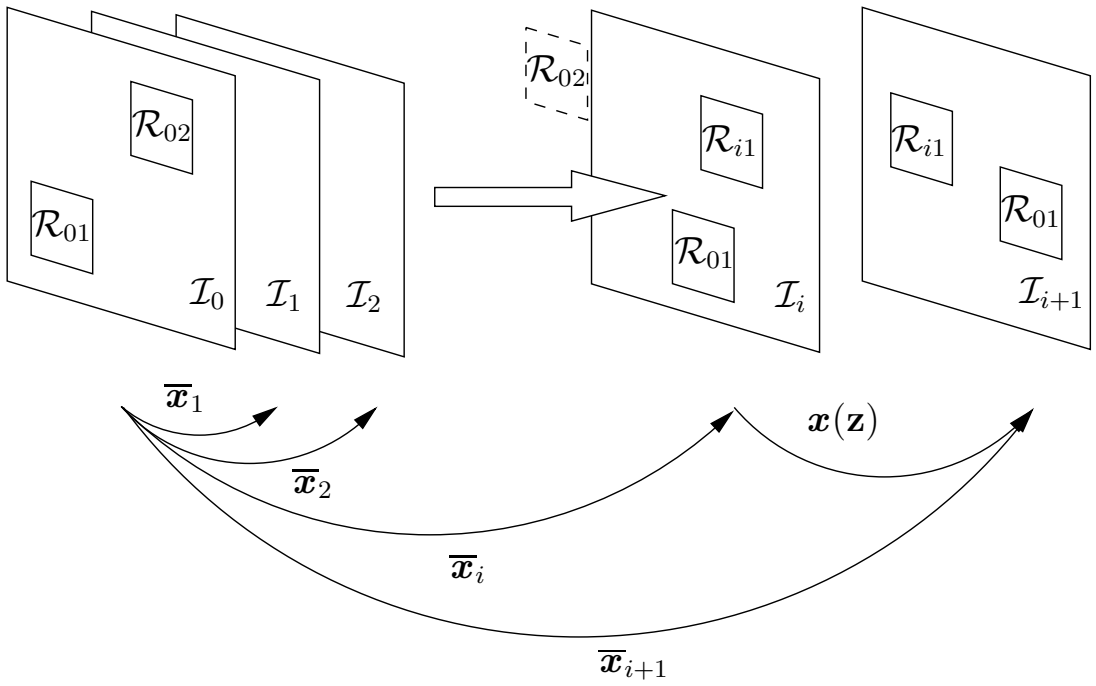


FIG. 3.11 – Incremental visual SLAM.

The problem of how to include new visual information in the estimation is particularly difficult when we consider a monocular camera since the structure of the scene is not

observable when the camera does not translate (for small translations the estimation of the structure is an ill conditioned problem). This causes an important problem for the initialization of the visual SLAM. Moreover, for monocular cameras the localization is obtained only up to scale (another sensor or "a priori" information is needed to recover the scale). Thus, an error in the estimation of the scale can have serious consequences on the accuracy of the localization and mapping.

Using a stereo-vision system (or even more cameras) can simplify the problem but not solve it completely. Indeed, when the baseline (the distance between the two cameras of the stereo-vision system) is small with respect to the distance to the observed scene the stereo-vision system will behave like a single camera.

Therefore, it is important to study both monocular and stereo configurations. The following two sections briefly review our contributions to these topics.

3.2.2.1 Monocular vision

Standard approaches to monocular SLAM often do not consider the estimation of variable illumination and separate the estimation of structure and motion into two distinct processes, e.g. [Molton 04]. When the estimation of structure and motion is integrated into a unified framework, for example using an Extended Kalman Filter, it assumes differential image displacements which limit the system to very slow camera motions, e.g. [Jin 03].

In [Silveira 07c] we have proposed a unified direct image registration approach for monocular SLAM that integrates the generic photometric model and fully exploits the ESM scheme to handle larger camera displacements. We have also made other contributions. First of all, we have handled the observability problems in the initialization step. Then, we have parametrized the structure in order to directly enforce positive depths in the optimization. We have shown that standard methods need to add new features to track more frequently. Hence, the proposed method allows the drift to be reduced by maintaining longer the estimation of the displacement with respect to the same reference frame.

3.2.2.2 Stereo vision

The visual SLAM can be simplified by considering a stereo-vision system [Comport 07]. In this case, the structure can be directly estimated from the dense stereo correspondences (the disparity map). Thus, the only unknown become the pose of the vision system. We applied the ESM optimization by defining a quadrifocal warping function $\mathbf{w} : \mathbb{SE}(3) \times \mathbb{R}^2 \times \mathbb{R}^2 \mapsto \mathbb{R}^2 \times \mathbb{R}^2$ that allows the points in the reference left and right images to be transformed into the points in the current left and right images. Let $\mathbf{p} = (\mathbf{p}_l, \mathbf{p}_r)$ be a vector containing the coordinates of the same point in the left and right reference images. The current points can be obtained as follows : $\mathbf{p}' = \mathbf{w}(\boldsymbol{g}, \mathbf{p})$ where $\boldsymbol{g} \in \mathbb{SE}(3)$ contains the pose of the current stereo pair. The warping function also depends on the parameters of the stereo pair (intrinsic camera parameters and the relative pose of the two cameras) but they are known and constant. We showed that the warping function

proposed in [Comport 07] is a group action of $\mathbb{SE}(3)$ on $\mathbb{R}^2 \times \mathbb{R}^2$ on the left. Thus, setting $\boldsymbol{x} = \mathbf{g}$ the ESM scheme can be applied without any approximation (see section 3.2.1.1) in order to align the current pair of images with the reference pair. Robustness with respect to occlusions and errors in the reconstruction of the disparity map is handled using the robust optimization techniques presented in Chapter 2.

Since we are dealing with large scale environments the reference pair must be updated regularly (see Figure 3.2.2.2). A set of key reference image-pairs are used to initialize tracking locally around the reference positions. These reference pairs provide a calibrated set of highly redundant dense correspondences to perform tracking and pose estimation. As will be shown by the experiments in Section 3.2.2.3, this leads to impressive results in real-scenes with occlusions, large inter-frame displacements, and very little drift over very long sequences of images.

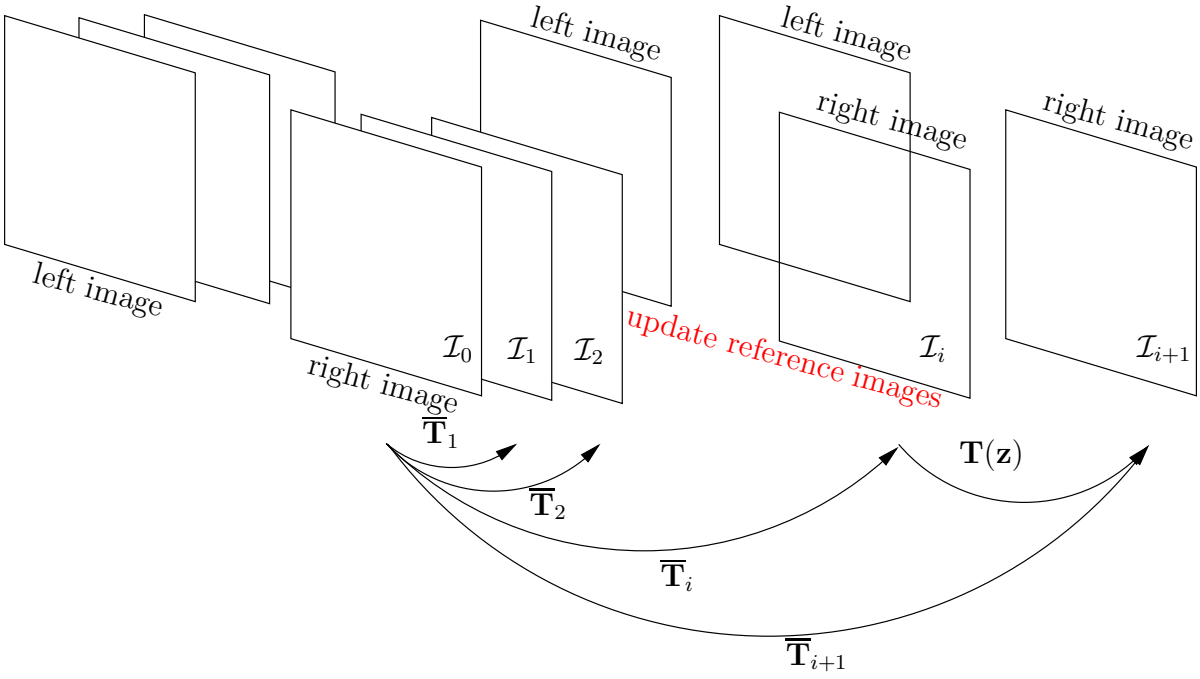


FIG. 3.12 – Incremental stereo visual SLAM.

3.2.2.3 Experimental results

In this section, I briefly describe three experiments that validate the accuracy of the direct image registration approach for visual SLAM.

Figure 3.13 shows a visual SLAM experiment with a catadioptric camera mounted on the mobile robot Anis. The odometry of the robot has been used as a ground truth (the experiment was run over a short distance) together with the assumption that two walls in the corridor are perpendicular. We manually selected four reference templates (figure 3.13). The two templates on the same wall were considered to be on the same plane (i.e. only one homography was estimated). For each homography, a translation up to a scale

factor and a rotation can be extracted (the ambiguity was solved by using several frames). The scale was then fixed by an off-line measure of the distance of the camera to one of the planes. The angles estimated between the planes were 87° and 91°. One of the templates was obstructed starting from image 100 but the median gives a very robust estimate : the mean absolute error was [0.01, 0.01, 0.01] meters for the translations and [1.6, 2.2, 1.0] degrees for the rotations. Details of the experiment can be found in [Mei 06a]. Jointly tracking all the templates provides even more precise results [Mei 06c].

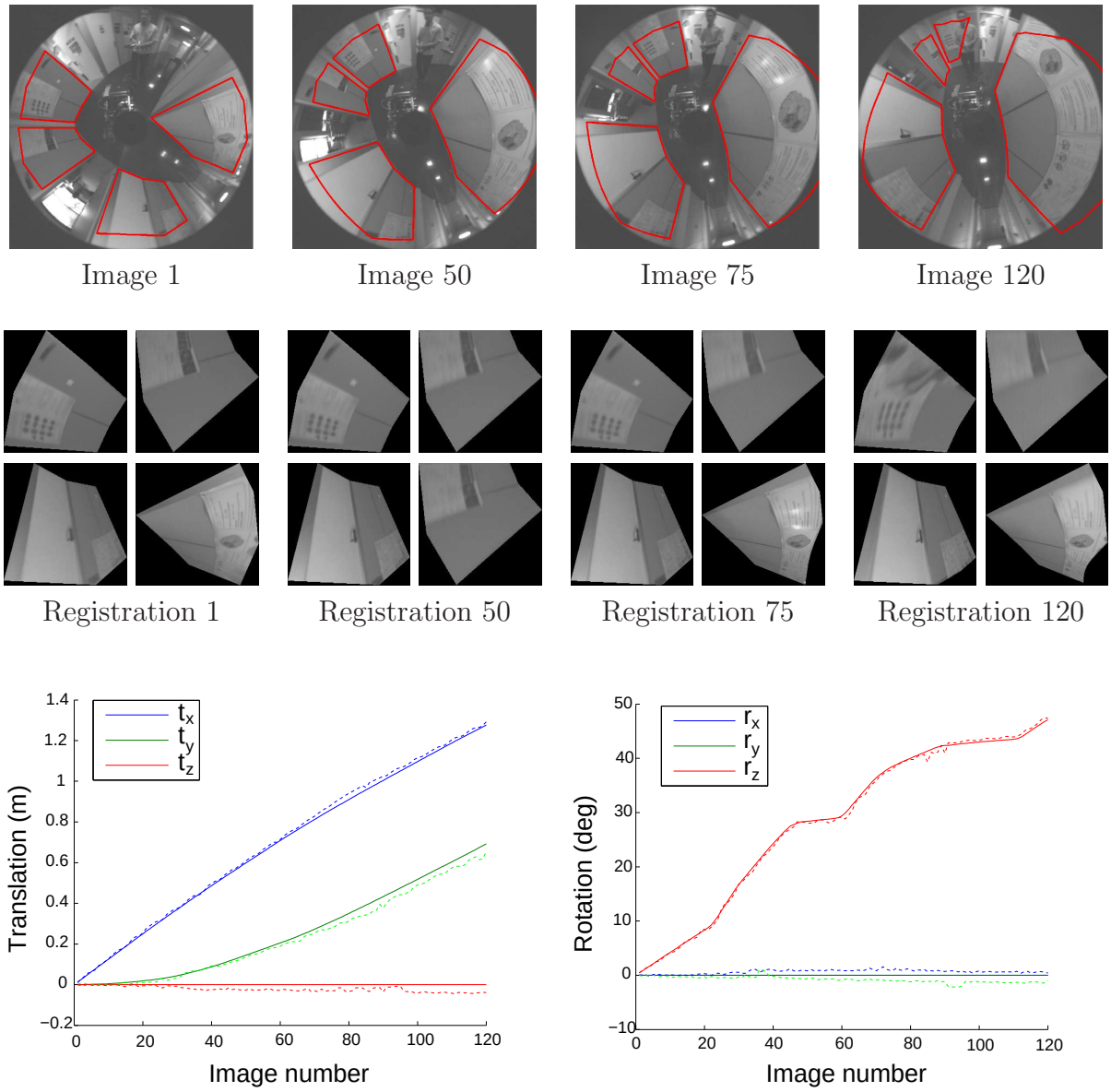
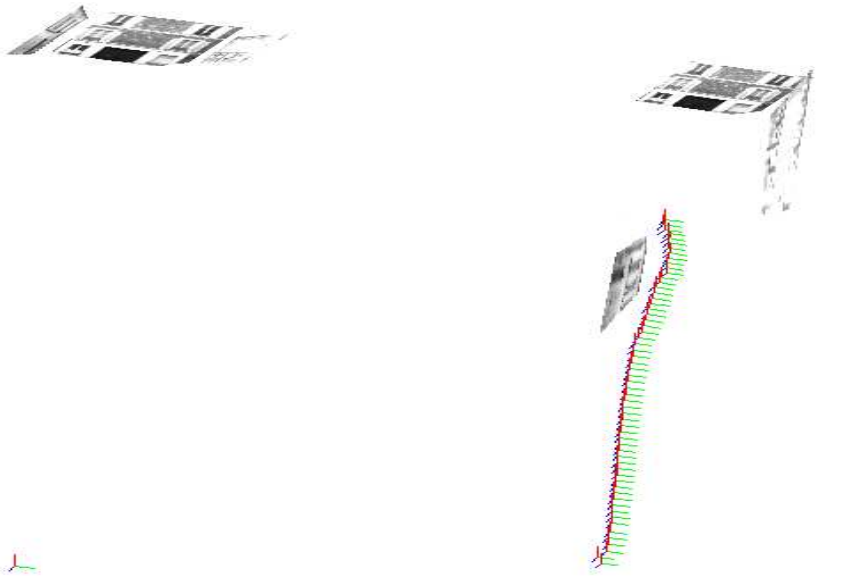


FIG. 3.13 – Direct visual SLAM with a calibrated omnidirectional camera mounted on a mobile robot. The plots show the estimated translations and rotations in dotted lines and the odometry of the robot in full lines.

Figure 3.14 shows a visual SLAM experiment with a pinhole camera. The camera is

mounted on a car travelling in Versailles, France. At the beginning of the experiment, the structure of the scene is unknown and it can be seen as belonging to the plane at infinity (see left image in Figure 3.14). Three planar regions were selected automatically using the approach proposed in [Silveira 06a]. As the camera progresses, the structure and the trajectory are recovered as shown in the right image of Figure 3.14. More details on this experiment, as well as comparisons with other visual SLAM approaches, can be found in [Silveira 07c].



Initialization Localization and mapping

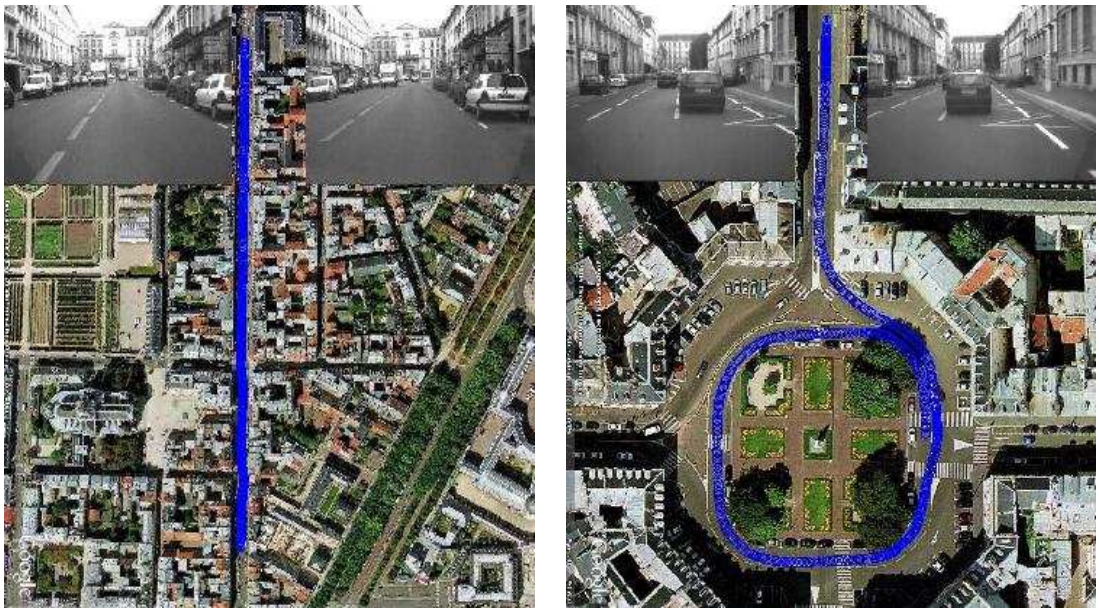
FIG. 3.14 – Direct visual SLAM with a calibrated pinhole camera. The image on the left shows that at the initialization the structure is unknown. The image on the right shows the structure and the localization (the trajectory of the camera with respect to the initial frame) being incrementally reconstructed.

Figure 3.15 shows two visual SLAM experiments with a stereo-vision system mounted on a car travelling along different streets in Versailles, France. More details on these experiments can be found in [Comport 07].

The image on the left in Figure 3.15 is that of a relatively straight road. The distance travelled by the car has been measured using road markings in the images and satellite views with a precision of $2.9cm/pixel$ for the Versailles region. The path length measured by both Google earth and our algorithm was about $440m$. Even if this is a qualitative result only, the path followed by the car corresponds well to the straight road. Throughout the sequence several moving vehicles pass in front of the cameras and, at one stage, a car is overtaken.

The image on the right in Figure 3.15, is particularly illustrative since a full loop of the roundabout was performed. In particular, this enables the drift to be measured at the

crossing point in the trajectory. In the case of this roundabout the drift at the crossing point was approximately $20cm$ in the vertical direction to the road-plane. Considering that the trajectory around the roundabout is approximately $200m$ long (measured using Google earth), this makes a drift of 0.01%.



Staright Line Roundabout

FIG. 3.15 – Accurate visual odometry with a calibrated stereo-vision system. The trajectory shown in blue has been manually superimposed on satellite images. Typical stereo images are shown at the top.

Chapitre 4

Vision-based robot control

Vision-based robot control (also called visual servoing) consists in using visual information in feedback loops. Vision processing and control design have to be combined with their respective specificities and their requirements taken into account. This topic has motivated a lot of research over the last 30 years. The reader may refer to numerous books, tutorials and surveys that have been published on the subject, like [Hashimoto 93, Hutchinson 96, Malis 02b, Chaumette 06, Chaumette 07]. The design of a vision-based robot control scheme depends not only on the task but also on the characteristics of the robot itself. Eye-to-hand systems have been considered in [Allen 93, Hager 95, Horaud 98]. Visual servoing for non-holonomic robots has been studied in [Fang 05, Maya 06] and for non-holonomic mobile manipulators in [Fruchard 06, De Luca 07a]. Visual Servoing of under-actuated robots has been considered in [Mahony 01, Hamel 04]. This chapter focuses on positioning tasks with a monocular eye-in-hand holonomic system.

In this context, we have made several contributions. A first contribution was the improvement of standard image-based control laws using the ESM scheme. We were also able to analyze theoretically the robustness of the local stability of several image-based control laws with respect to errors on structure parameters. A second contribution was the design of robust image-based control schemes that do not need an exact measure of the intrinsic parameters of the vision system and that do not need any "a priori" knowledge of the structure of the observed scene. Again, particular emphasis has been given to the theoretical analysis of the robustness of the control laws with respect to errors on the uncertain parameters. Finally, I proposed two methods that go beyond the standard visual servoing approaches : the invariant visual servoing to extend the image-based approach and the controlled visual SLAM to extend the position-based approach.

The present chapter is organized as follows. First, standard image-based visual servoing schemes are discussed. I will show how they can be improved by using the ESM scheme and discuss the results of the stability analysis. Then, the research work on robust image-based control schemes will be described. Finally, the two methods that go beyond standard visual servoing approaches will be presented.

4.1 Design of image-based control laws

As already mentioned in Chapter 1, teaching-by-showing approaches have been introduced in order to avoid the explicit reconstruction of the pose of the vision system [Weiss 87]. If we suppose that the imaging conditions do not change after acquisition of the reference image $\boldsymbol{h}(\overline{\boldsymbol{\eta}}, 0)$ (e.g. the camera intrinsic parameters are constant, we observe a rigid object, the illumination does not change, ...) and that we have enough visual information, then the robot is correctly positioned when the current visual information coincides with that of the reference. Let me rewrite here the state equations (1.43) :

$$\begin{aligned} \dot{\boldsymbol{\varepsilon}} &= \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})\, \mathbf{v} \\ \mathbf{y} &= \boldsymbol{h}(\overline{\boldsymbol{\eta}}, \mathbf{x}) \end{aligned}$$

where the vector $\boldsymbol{\varepsilon} = \boldsymbol{\delta}(\boldsymbol{h}(\overline{\boldsymbol{\eta}}, \mathbf{x}), \boldsymbol{h}(\overline{\boldsymbol{\eta}}, 0))$ is such that if (and only if) $\boldsymbol{\varepsilon} = 0$ then $\mathbf{x} = 0$. The problem considered here is to design a stable control law $\mathbf{v}$ (i.e. the velocity of the camera) in order to regulate $\boldsymbol{\varepsilon}$ to zero. For holonomic robots, the design of such a control law has several analogies with the numerical optimization methods presented in Chapter 2. These analogies are briefly reviewed in Section 4.1.1 (a more detailed discussion can be found in [Malis 04a]). The analogies between vision-based control laws and numerical optimization are important in order to understand why the application of the ESM scheme to vision based control leads to several improvements over standard control laws. These improvements are discussed in Section 4.1.2.

4.1.1 Analogies between numerical optimization and control

In order to use the same notations as Chapter 2 let me write :

$$\mathbf{f}(\mathbf{x}) = \boldsymbol{\delta}(\boldsymbol{h}(\overline{\boldsymbol{\eta}}, \mathbf{x}), \boldsymbol{h}(\overline{\boldsymbol{\eta}}, 0)) = \boldsymbol{\varepsilon}$$

Such that $\mathbf{f}(0) = 0$. The derivative of $\mathbf{f}(\mathbf{x})$ is :

$$\dot{\mathbf{f}}(\mathbf{x}) = \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})\, \mathbf{v}$$

Suppose that the matrix $\mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})$ is full rank $\forall \mathbf{x}$. Let me build the following Lyapunov function :

$$c(\mathbf{x}) = \frac{1}{2}\mathbf{f}(\mathbf{x})^\top \mathbf{f}(\mathbf{x})$$

which can also be viewed as a cost function. Our objective is to go from a starting point $\mathbf{x}_0$ to the equilibrium point. The time derivative of the Lyapunov function is :

$$\dot{c}(\mathbf{x}) = \mathbf{f}(\mathbf{x})^\top \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})\, \mathbf{v}$$

If a control law $\mathbf{v}$ exists such that $\dot{c}(\mathbf{x}) < 0$ then the system is asymptotically stable.

4.1.1.1 Standard control laws

Several vision-based robot control methods have been proposed. For example :
The Jacobian transpose control law (e.g. [Hashimoto 93]) :

$$\mathbf{v} = -\lambda \, \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})^{\top} \, \mathbf{f}(\mathbf{x})$$

where $\lambda > 0$ and

$$\dot{c}(\mathbf{x}) = -2 \, \alpha \, \mathbf{f}(\mathbf{x})^{\top} \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x}) \, \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})^{\top} \, \mathbf{f}(\mathbf{x}) < 0$$

The Jacobian inverse control law (e.g. [Espiau 92]) :

$$\mathbf{v} = -\mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})^{-1} \mathbf{f}(\mathbf{x})$$

where

$$\dot{c}(\mathbf{x}) = -\mathbf{f}(\mathbf{x})^{\top} \mathbf{f}(\mathbf{x}) < 0$$

The Damped Least-squares control law (e.g. [Wampler 86, Nakamura 86]) :

$$\mathbf{v} = -(\mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})^{\top} \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x}) + \gamma \mathbf{I})^{-1} \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})^{\top} \mathbf{f}(\mathbf{x})$$

where $\gamma \geq 0$ and

$$\dot{c}(x) = -\mathbf{f}(\mathbf{x})^{\top} \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x}) (\mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})^{\top} \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x}) + \gamma \mathbf{I})^{-1} \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})^{\top} \mathbf{f}(\mathbf{x}) < 0$$

These control laws are all globally asymptotically stable.

Supposing that we can measure the matrix at the equilibrium point $\mathbf{L}(\overline{\boldsymbol{\eta}}, 0)$, the authors of [Espiau 92] proposed the following control law :

$$\mathbf{v} = -\mathbf{L}(\overline{\boldsymbol{\eta}}, 0)^{-1} \mathbf{f}(\mathbf{x})$$

where

$$\dot{c}(\mathbf{x}) = -\mathbf{f}(\mathbf{x})^{\top} \mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x}) \mathbf{L}(\overline{\boldsymbol{\eta}}, 0)^{-1} \mathbf{f}(\mathbf{x}) < 0$$

which is stable if $\mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x}) \mathbf{L}(\overline{\boldsymbol{\eta}}, 0)^{-1} > 0$. This certainly happens in a neighborhood of the equilibrium point.

4.1.1.2 Standard optimization methods

Each vision-based control method corresponds to a standard optimization method. The Damped Least-squares control law corresponds to the Levemberg-Marquardt optimization method, the Jacobian transpose control law corresponds to Steepest descent optimization method, and the Jacobian inverse control law corresponds to the Gauss-Newton method. Finally, the control law using the interaction matrix computed at the equilibrium corresponds to the Efficient Gauss-Newton method presented in Chapter 2.

The main difference between control and optimization methods is that in optimization methods the measure of the cost function at the new location is obtained from a virtual model. When we control a robot, it physically moves to a new location to acquire a new measure of the cost function. Thus, we do not necessarily need a model of the cost function to compute its value at the new location. On the other hand, the behavior of the robot may be perturbed in order to obtain the information needed by the optimization algorithm.

4.1.2 The ESM for vision-based control

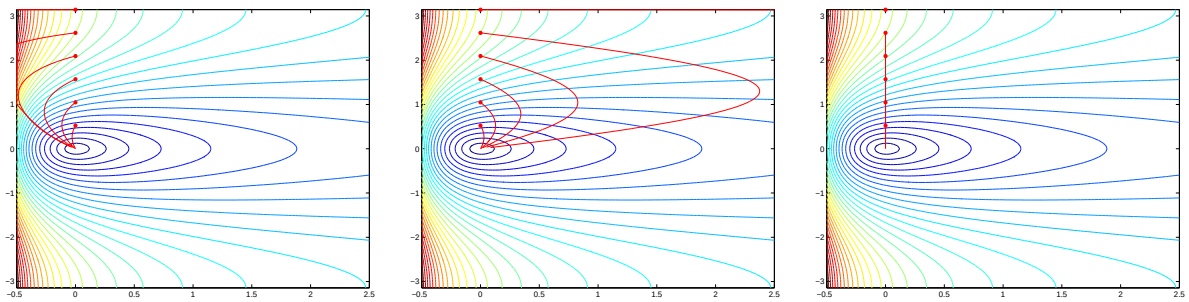
4.1.2.1 Improving image-based visual servoing schemes

Similarly to vision-based estimation methods, most vision-based control approaches are based on the extraction of visual features [Espiau 92, Cervera 03]. Let me consider, for example, interest points as visual features. The standard approach consists in building from the selected features a task function $\boldsymbol{\epsilon}$ diffeomorphic to the camera pose. Then, we can use the standard control laws described in the previous section to regulate the task function to zero. This approach generally works very well if the starting pose of the camera is not too far from the reference pose. Otherwise, the behavior of the robot in the Cartesian space may be not satisfactory. For example, it is now well known [Chaumette 98] that if the initial camera displacement is a rotation around the $\vec{z}$ axis an undesirable motion is induced by using standard control laws. For example, if we use the Gauss-Newton control law (the interaction matrix is updated at each iteration) the camera moves backward while rotating. On the other hand, if we use the Gauss-Newton control law (the interaction matrix is constant and computed at the equilibrium) the camera moves forward while rotating. Figure 4.1 illustrates these two problems. The images show the isolines of the cost function projected into the subspace $(\vec{t_z}, \vec{r_z})$. We repeated several simulations with an increasing initial rotation. Since the initial movement is a pure rotation, the ideal path (in the Cartesian space) should be a straight line perpendicular to the $\vec{t_z}$ axis (i.e. $t_z = 0$). On the contrary, we observe that as r_z reaches $\pm\pi$ the translational motion becomes bigger since the isolines become perpendicular to the $\vec{t_z}$ axis (i.e. the steepest descent direction is along $\vec{t_z}$).

These problems can be solved by using the ESM scheme presented in Chapter 2. The ESM control law is :

$$\mathbf{v} = -\frac{1}{2}\left(\mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x}) + \mathbf{L}(\overline{\boldsymbol{\eta}}, 0)\right)^{-1} \mathbf{f}(\mathbf{x})$$

Figure 4.1 shows that using this control law the camera performs a pure rotation around the $\vec{z}$ axis. The benefits of using the ESM control law are not limited to the behavior in this particular case and a more detailed discussion can be found in [Malis 04a]. Note that, around the equilibrium point the three control laws have the same behavior. Indeed, if $\mathbf{x} \approx 0$ then $\mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x}) \approx \mathbf{L}(\overline{\boldsymbol{\eta}}, 0) \approx \frac{1}{2}\left(\mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x}) + \mathbf{L}(\overline{\boldsymbol{\eta}}, 0)\right)$. Thus, the stability and robustness analysis presented in the next section applies to all of them.



Efficient Gauss-Newton Gauss-Newton ESM

FIG. 4.1 – Comparison between standard control methods and the ESM.

4.1.2.2 Stability and robustness analysis

In an image-based visual servoing approach, the task function can be computed directly from image data (e.g. the features extracted in the image). On the other hand, the interaction matrix $\mathbf{L}(\overline{\boldsymbol{\eta}}, \mathbf{x})$ that links the derivative of the task function to the velocity of the camera depends on the camera intrinsic parameters and may also depend on some information about the structure of the target (all these parameters are in the vector $\overline{\boldsymbol{\eta}}$). For example, when the selected features are interest points the interaction matrix depends on the depths (the Z coordinates) of the corresponding 3D points. A good estimation of the interaction matrix is necessary in order to build a stable control law. It was observed experimentally that a rough estimation of the parameters $\widehat{\boldsymbol{\eta}}$ was sufficient for a stable control :

$$\mathbf{v} = -\frac{1}{2}\left(\mathbf{L}(\widehat{\boldsymbol{\eta}}, \mathbf{x}) + \mathbf{L}(\widehat{\boldsymbol{\eta}}, 0)\right)^{-1}\mathbf{f}(\mathbf{x})$$

However, it is important to theoretically understand how big the estimation error $\|\widehat{\boldsymbol{\eta}} - \overline{\boldsymbol{\eta}}\|$ on the parameters can be while still having a stable control law. Due to the complexity of the theoretical analysis very few results have been reported in the literature. Results have been obtained only in a few simple special cases [Espiau 93] [Cheah 98] [Deng 02], often considering a simplified camera model and always supposing the 3D structure is perfectly estimated. We have studied the robustness of standard image-based visual servoing control laws with respect to uncertainties on the structure of the target [Malis 03a] [Malis 02a]. We proved theoretically that even small errors on the depths may lead to unstable control laws [Malis 03c]. The proof has been extended to any central camera in [Mezouar 04]. We not only provided necessary and sufficient conditions for the local stability but also sufficient conditions that can more easily be tested. From these conditions we can measure the "size" of the possible errors on the depths.

Figure 4.2 illustrates the results of the theoretical analysis with the example of a planar target. When the target is planar, the depths are related to the normal vector $\mathbf{n}$ to the plane. Without loss of generality, let me suppose here that $||\mathbf{n}| = 1$. Then $\mathbf{n}$ can be written as a function of two parameters $\mathbf{n}(\theta, \phi) = (\cos(\theta)\sin(\phi), \sin(\theta)\sin(\phi), \cos(\phi))$. Thus all the estimated depth $\widehat{Z}_i$ can be obtained using an approximation of $\widehat{\mathbf{n}}(\widehat{\theta}, \widehat{\phi})$. The figure shows the stability regions for a pinhole and for a catadioptric camera as a function of $(\widehat{\theta}, \widehat{\phi})$ for 8 or 16 points on the same plane. The true normal is $\mathbf{n} = (0.5, 0, 0.866)$ (i.e. the black cross at $\theta = 0$ and $\phi = \pi/6$). If we choose the estimated parameters $(\widehat{\theta}, \widehat{\phi})$ in the green region the control law will be locally asymptotically stable. On the other hand, if we choose the estimated parameters in the red region the system control law is locally unstable. The normals obtained for parameters in the blue region are discarded since we obtain at least a negative depth, which is impossible. Note that the cameras have similar stability regions. Increasing the number of points on the target decreases the unstable region but does not eliminate it completely. More complete results can be found in [Malis 03c, Mezouar 04].

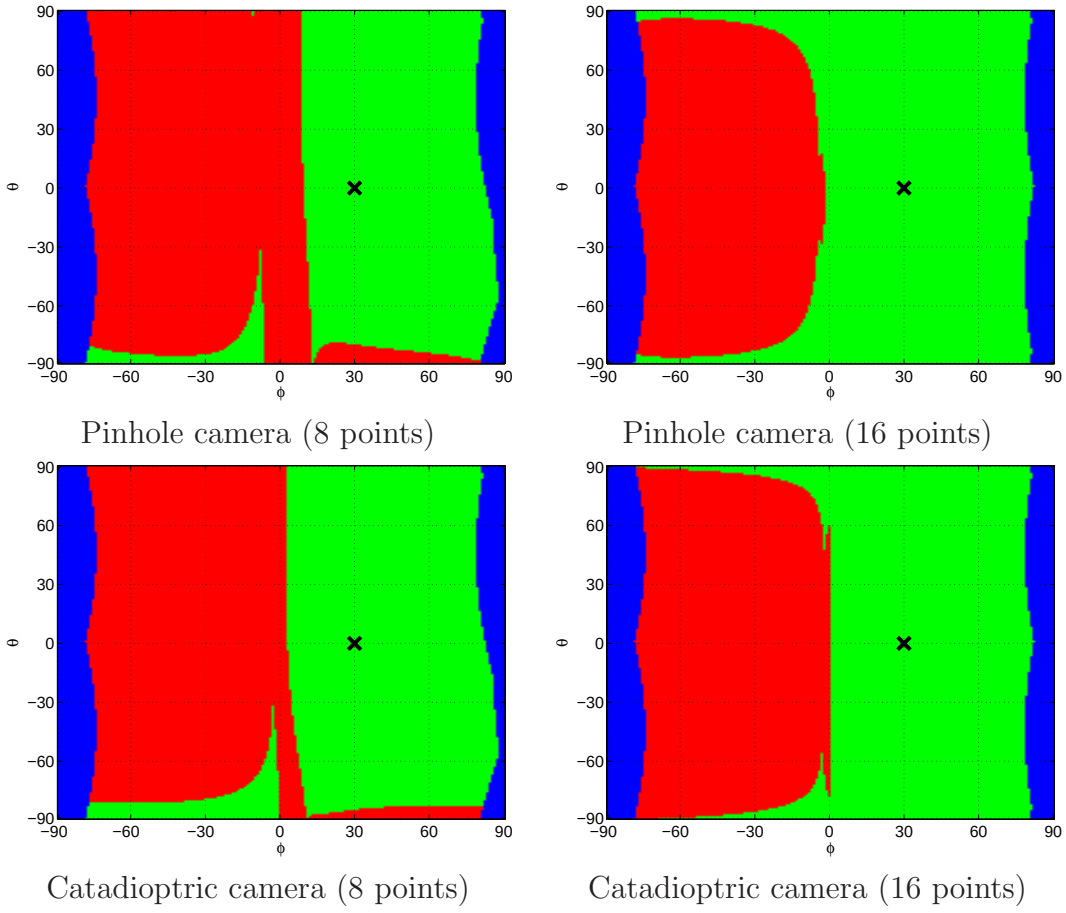


FIG. 4.2 – Stability regions for planar targets with 8 or 16 points observed by a pinhole or a catadioptric camera. Parameters (ϕ, θ) selected in the green regions lead to stable control while parameters selected in the red regions lead to unstable control.

4.2 Design of robust visual servoing schemes

Despite the improvements in the design of image-based control laws, the theoretical analysis of the robustness shows that for the considered control laws we need an estimation of the structure of the target. Therefore, it is important to investigate control methods that do not explicitly need this estimation.

4.2.1 Increasing the robustness of image-based approaches

4.2.1.1 A new class of visual servoing schemes

In [Malis 02f] we proposed a new class of visual servoing schemes that encompasses several image-based visual servoing approaches such as [Malis 99, Basri 99]. The new class of visual servoing schemes is based on the reconstruction of the pose (up to a scale factor for the translation) of the camera from two images. The key idea of the new class

is to estimate the rotation from two views. Therefore the rotation is directly controlled and decoupled from the control of the translation. The translation can be controlled in several different ways which differentiate between the possible methods within the class. This class of methods has interesting properties. Since the rotation control is decoupled from the translation control, it is possible to "easily" study the stability and robustness of the control law with respect to errors on the intrinsic and extrinsic camera parameters (the extrinsic parameters are the pose of the camera with respect to the robot end-effector) [Malis 01a, Malis 02f]. Again, the analysis of the robustness of the control law is very important since it allowed us to show that the "size"of the calibration error that the control laws can tolerate is very large.

The new class of visual servoing schemes is based on the assumption that we can estimate the rotation matrix directly from the current and reference images. In the more general case (i.e. both for non-planar and planar objects), the rotation can be extracted from the homography matrix (see Chapter 3). Theoretically, the homography matrix can be estimated from any sufficient visual information (e.g. a set of points, of lines, ...). For example, we proposed in [Chesi 00] a complete vision-based control system with respect to planar contours. In the system we integrated the visual matching [Chesi 99], the visual tracking [Drummond 99] and the visual servoing [Malis 99]. However, this integration work highlighted two problems in our approach. First of all, the feature-based estimation for the task function was specific to planar contours. A different system should be built for different features. Secondly, it is impossible to compute the rotation from the homography matrix using image data only (the current and reference image). Indeed, there are two possible solutions when we decompose an homography matrix. This second problem led us to propose a different vision-based control method that is detailed in the next section.

The solution to the first problem has been to estimate the homography directly from image data without any feature extraction [Malis 04d, Malis 05]. We use the ESM for the direct image registration of a planar surface (see Section 3). This allowed us to simplify the design of the visual servoing approach and to increase its flexibility since the planar surface can contain any information. Figures 4.3 and 4.4 show a vision-based car-platooning experiment that is performed in a real outdoor environment.

Figure 4.3 gives an overview of the system while Figure 4.4 illustrates more details of the experiment. Two electric vehicles of type "Cycab" are used : one as a guider car and the other as a follower car. A driver guides the first car while the follower car is controlled by a position-based control scheme. The control scheme takes into account that the vehicle is non-holonomic and tries to keep the distance between the two vehicles constant and equal to the initial distance. The relative position is given by the ESM visual tracking system. The pan-tilt turret is controlled in order to keep the guider car in the field of view of the camera during the experiment.

In the starting situation, when the guider car is in front of the follower car, a window of (100×100) pixels is selected to be the reference pattern. In order to have a metric reconstruction, the camera was roughly calibrated and the distance between the two cars is given to the control process. It is the distance between the camera of the follower car and a poster stacked on the back windshield of the guider car. Tracking this reference pattern provides the relative position between the two vehicles. The blue square indicates the tracked region. In the right column, the reprojections of the tracked region using the estimated homographies are shown. The first row of the figure corresponds to the initial position. The ESM tracking algorithm performs well although the experiment takes place outdoors and sun reflection on the tracked region occurs. The current pattern reprojections are very close to the reference one. More details on the experiment can be found in [Benhimane 05].

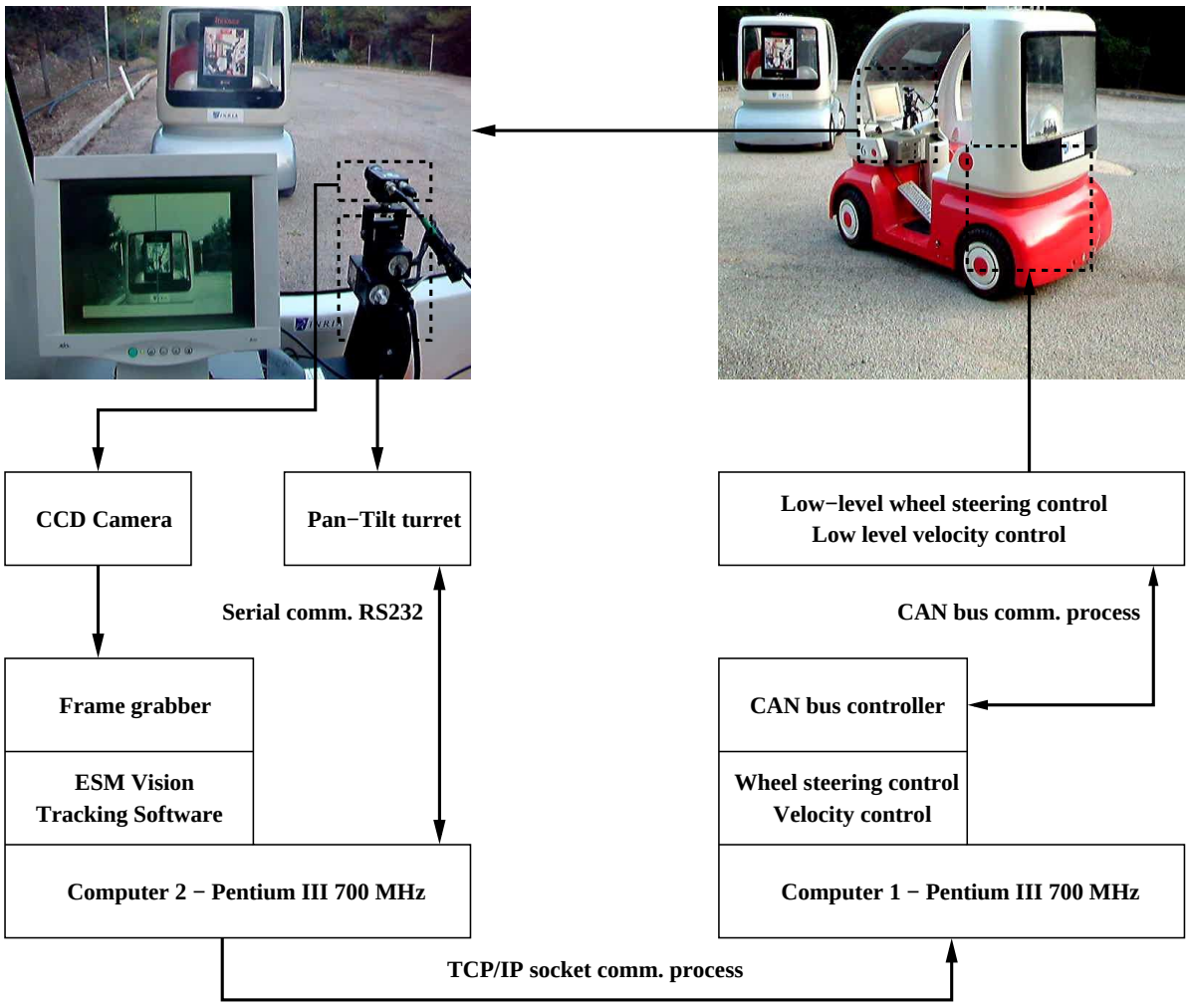


FIG. 4.3 – Overview of the compete system.

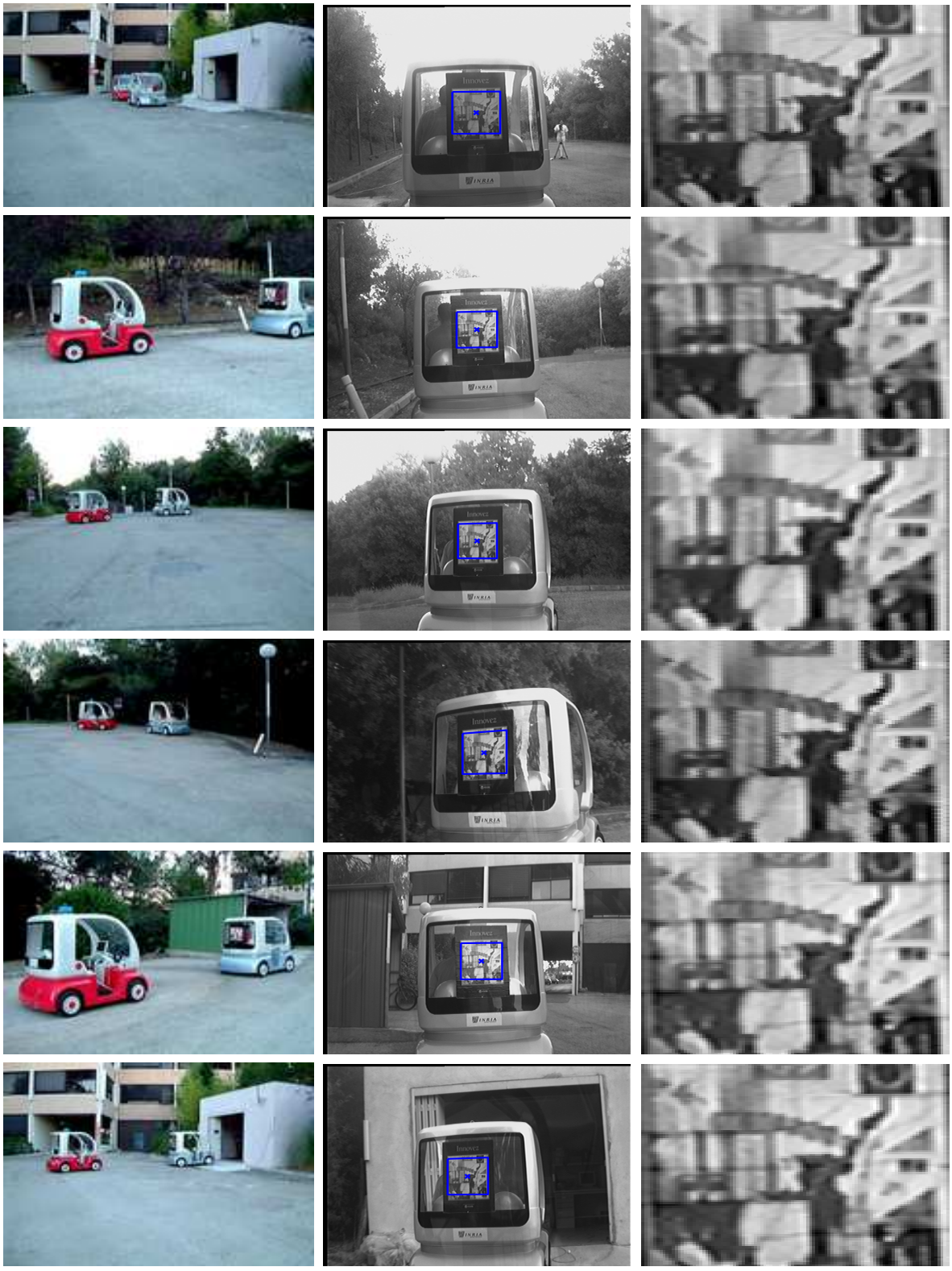


FIG. 4.4 – Several images of a platooning application. The first column shows the red car automatically following the leader. The second column shows the area of interest in the current image. The third column shows the registered images that prove that the homography has been correctly estimated.

4.2.1.2 Direct visual servoing

As already mentioned, our objectives have been to find a task function that is diffeomorphic to the pose of the camera and to find a stable control law that can easily be computed just from visual data extracted from the current and the reference images (independently of the shape of the object). If the target is not planar these objectives can be fulfilled by the visual servoing schemes presented in the previous section. On the other hand, when the target is planar, image data from the reference and current image only is not sufficient to compute the control law. In this case, the two images of the target are related by a homography matrix. The decomposition of the homography generally has two solutions and we need additional information to distinguish the true solution from the false one.

Therefore, we studied how the two solution of the decomposition are related (see Chapter 3). In [Vargas 05] we proved that the two solution of the homography can be "mixed" in the computation of a stable control law. For example, we defined a task function containing the average of the two translations and of the two rotations (on the $\mathbb{SO}(3)$ group). Alternatively we can compute the average of the two rotations and control the remaining d.o.f.s as for the 2.5D visual servoing [Malis 07b]. With such an approach we can completely avoid the use of "a priori" information to choose the true solution of the homography decomposition. However, contrary to the 2.5D visual servoing scheme the stability analysis becomes extremely difficult since the rotation control is no longer decoupled from the translation control.

In [Benhimane 06a] we proposed a task function that is locally diffeomorphic to the camera pose and that can be directly computed from the homography matrix. Therefore, we completely avoid the decomposition process. We also proposed a simple control law and we theoretically proved its local stability. Experimentally we observed that the stability region is very large (see the experiments below). In order to build a globally stable control scheme we should plan a trajectory and work with small errors. This was not possible with the task function proposed in [Benhimane 06a] since it was only locally diffeomorphic to the camera pose. Another limitation of the approach was that it was designed for planar targets only.

Recently, we have generalized the approach to non-planar targets and we have modified the task function to allow for path planning [Silveira 07a]. We have also proposed a simple control law and we have theoretically prove its local stability. The general task function that we have proposed is the following :

$$\boldsymbol{\varepsilon} = \begin{bmatrix} (\mathbf{H} - \mathbf{I})\underline{\mathbf{q}} + \rho\,\mathbf{t} \\ \vartheta(\mathbf{H})\,\boldsymbol{\mu}(\mathbf{H}) \end{bmatrix} \tag{4.1}$$

where $\mathbf{H}$ is a homography matrix related to a virtual plane, $\mathbf{t}$ is the translation, $\underline{\mathbf{q}}$ is a chosen control point in the reference image that corresponds to a point on the target, and ρ is a scalar that encodes the parallax of this point with respect to the virtual plane. Finally, $[\boldsymbol{\mu}(\mathbf{H})]_\times = \mathbf{H} - \mathbf{H}^\top$ and $\vartheta(\mathbf{H})$ is detailed in [Silveira 07a]. Note that the task function can be computed from image data only (i.e. we do not measure $\mathbf{t}$ and ρ separately but the vector $\rho\,\mathbf{t}$, which is measured from the epipolar geometry). This task

function generalizes the one proposed in [Benhimane 06a] for planar targets. Indeed, for planar targets the parallax is zero $\rho = 0$ and if we set $\vartheta(\mathbf{H}) = 1$ we obtain the same task function proposed in [Benhimane 06a]. The modification that allows us to perform trajectory planning is on the scale factor $\vartheta(\mathbf{H})$ (see [Silveira 07a]).

The experimental validation of the direct visual servoing approach is in progress. I will describe here two experiments for the planar case [Benhimane 06a] and two promising simulations for the general case [Silveira 07a].

We have tested the control scheme proposed in [Benhimane 06a] on the Cartesian robot of the LAGADIC research team at INRIA Rennes. The robot is accurately calibrated and it provides a ground truth for measuring the accuracy of the positioning task. A calibrated camera is mounted on the end-effector of the robot and observes a planar target. Starting from another pose (the initial pose) which allows the target to be seen from a different view point, the robot is controlled using the control law in order to get back to the reference pose. At the initial pose the translation displacement is 0.68 meters and the rotation displacement is 96 degrees. We use the ESM visual tracking algorithm [Benhimane 04] to register the area of interest and to estimate at the same time an uncalibrated homography matrix $\mathbf{G}$. We use the center of gravity of the template as the control point.

In the first experiment, illustrated in Figure 4.5, we used the true camera intrinsic parameters matrix $\mathbf{K}$ to obtain a calibrated homography $\mathbf{H} = \mathbf{K}^{-1}\mathbf{G}\mathbf{K}$ from which the control law is computed. The control law is stable and both translation and rotation velocities converge to zero. At the convergence, the robot is back to its reference pose and the visual information coincides with the visual information of the reference pose. As shown in Figure 4.5, the camera displacement converges to zero very accurately (less than 1 mm error for the translation and less than 0.1 degrees for the rotation).

Figure 4.6 illustrates a second experiment that is performed under similar conditions (the same initial camera displacement, an unknown normal vector to the plane, an unknown camera/object distance...). On the other hand, we use a very bad estimation of the camera intrinsic parameters to estimate the homography matrix $\widehat{\mathbf{H}} = \widehat{\mathbf{K}}^{-1}\mathbf{G}\widehat{\mathbf{K}}$. However, the control law is robust to camera calibration errors : the translation and the rotation velocities converge to zero. At the convergence, the visual information coincides again with the visual information of the reference image and the camera displacement converges to zero (as in the previous experiment we obtain around 1 mm error for the translation and around 0.1 degrees for the rotation).

We have performed two simulations to validate the general control scheme proposed in [Silveira 07a]. We used the image registration algorithm proposed in [Malis 07a] to measure an uncalibrated homography $\mathbf{G}$ and the vector $\rho\,\mathbf{e}$ from which we obtained $\widehat{\mathbf{H}} = \widehat{\mathbf{K}}^{-1}\,\mathbf{G}\,\widehat{\mathbf{K}}$ and $\rho\,\widehat{\mathbf{t}} = \rho\,\widehat{\mathbf{K}}^{-1}\mathbf{e}$.

The first simulation in Figure 4.7 shows that the new control scheme encompasses the previous one specifically designed for planar targets. Additional simulations can be found in [Silveira 07a] that show how path planning can handle larger initial displacements than the previous approach.

The second simulation in Figure 4.8 shows that the general control scheme works well with non-planar targets and for very large displacements. We planned a straight line

trajectory in the image for the control point in order to keep the target in the field of view of the camera. In this simulation we also added errors in the estimated camera intrinsic parameters in order to demonstrate the robustness of the control law.

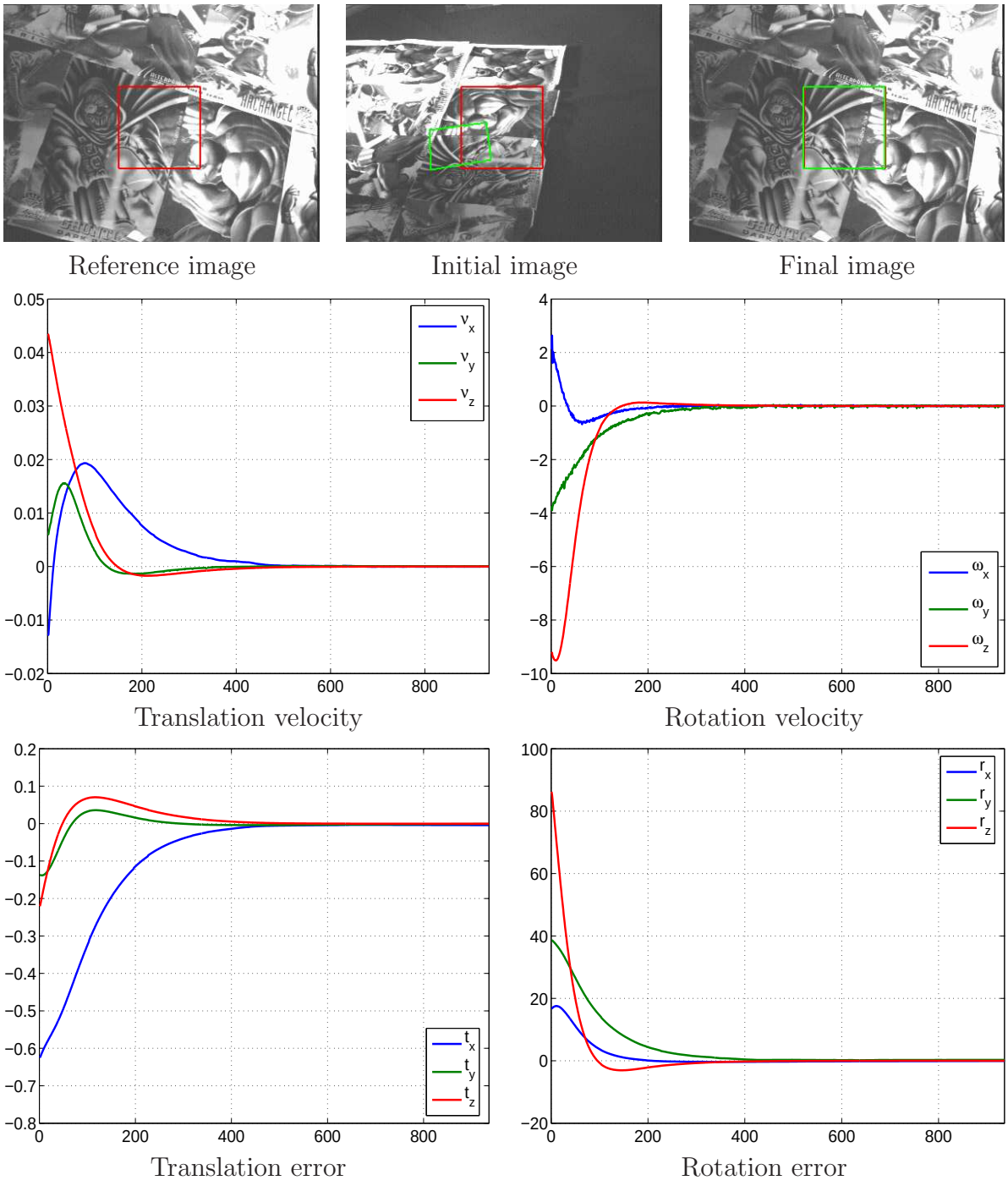


FIG. 4.5 – Direct visual servoing with respect to a planar target. A calibrated pinhole camera is mounted on a calibrated Cartesian robot.

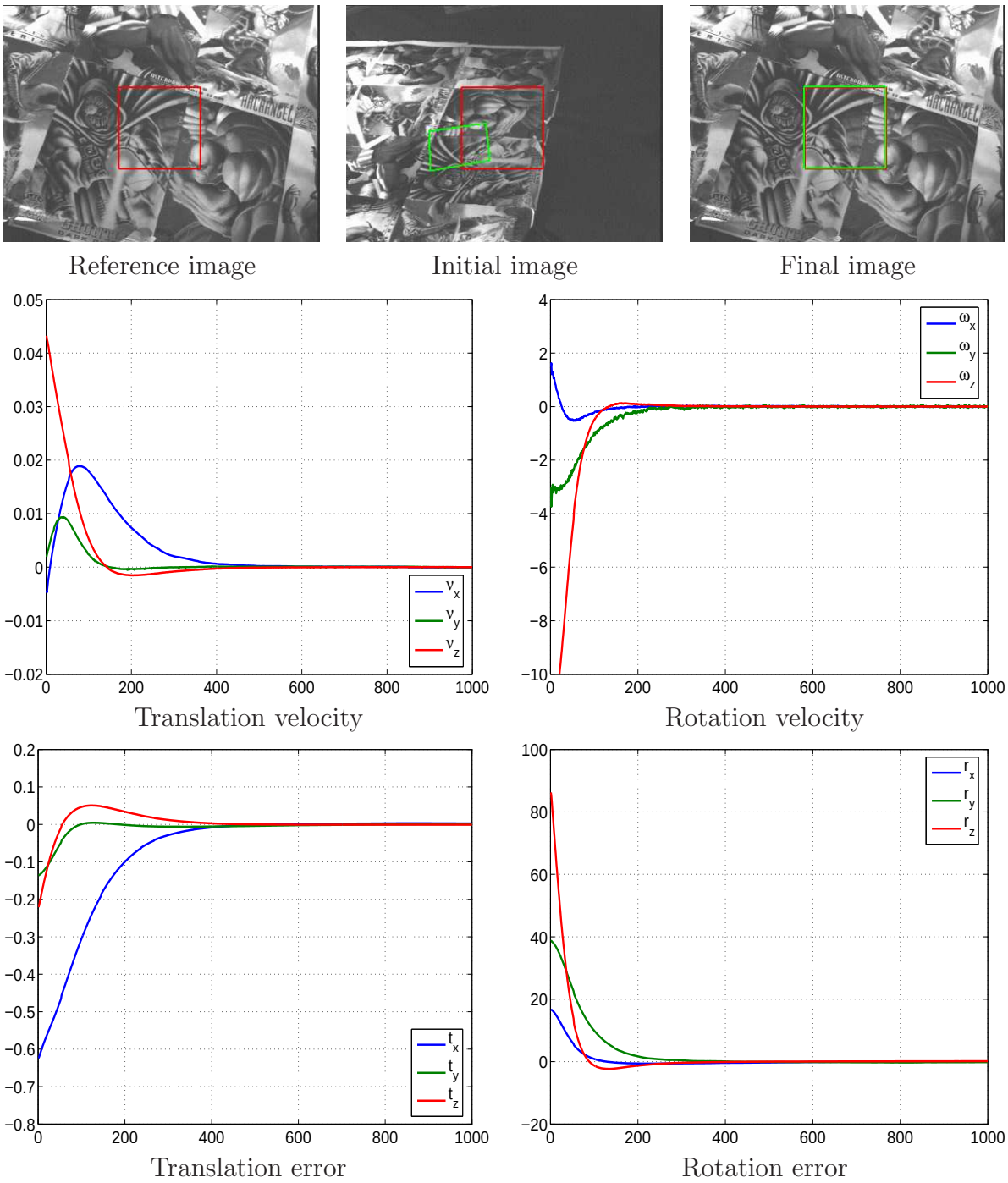
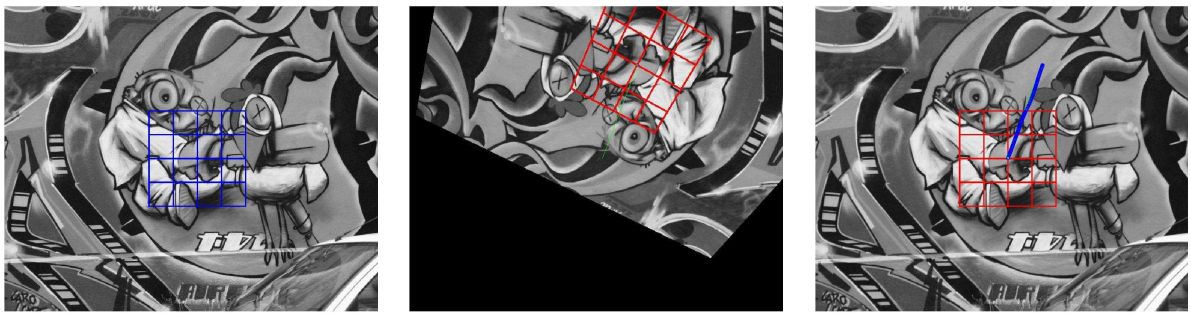
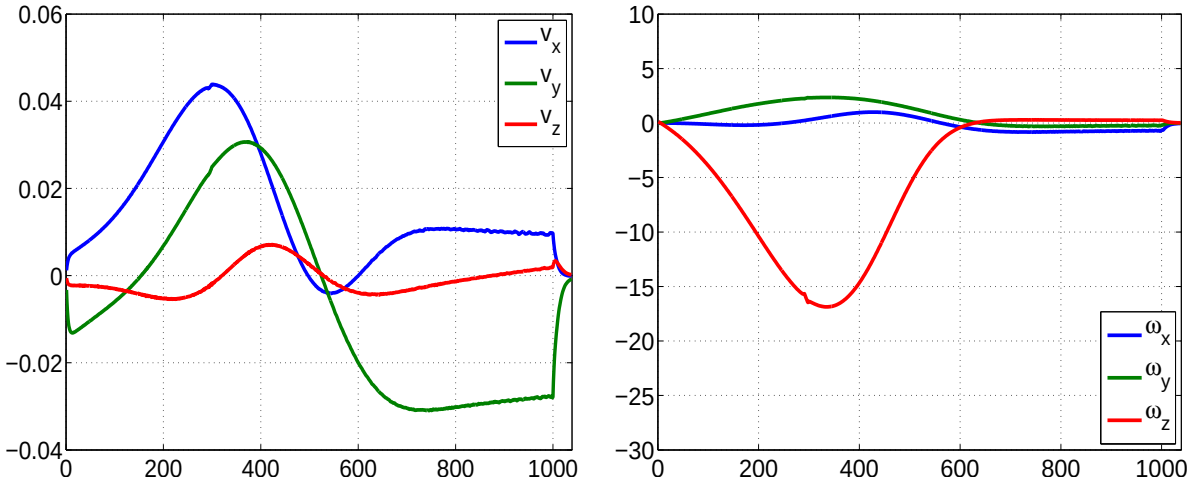


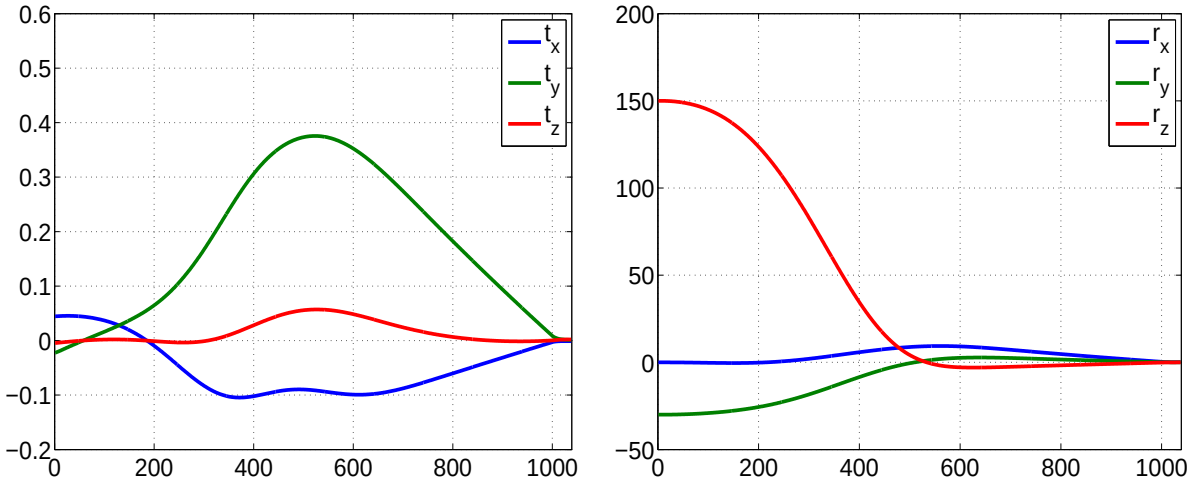
FIG. 4.6 – Direct visual servoing with respect to a planar target. An uncalibrated pinhole camera is mounted on a calibrated Cartesian robot.



Reference image Initial image Final image

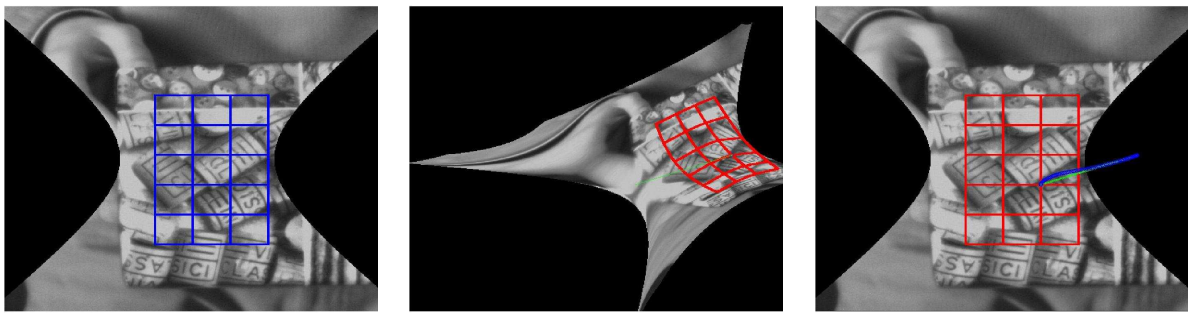


Translation velocity Rotation velocity

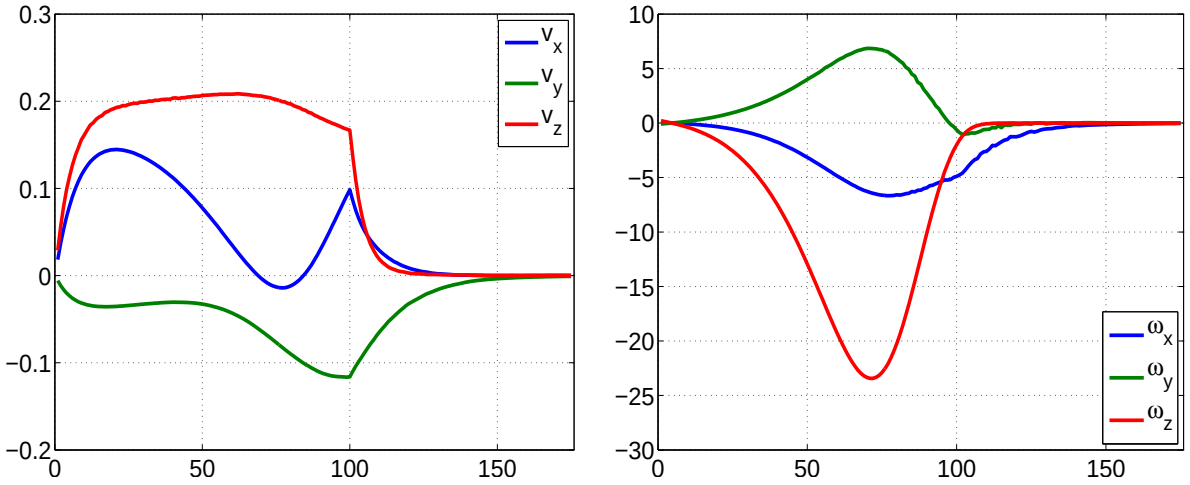


Translation error Rotation error

FIG. 4.7 – Direct visual servoing with respect to a planar target using the generic control law.

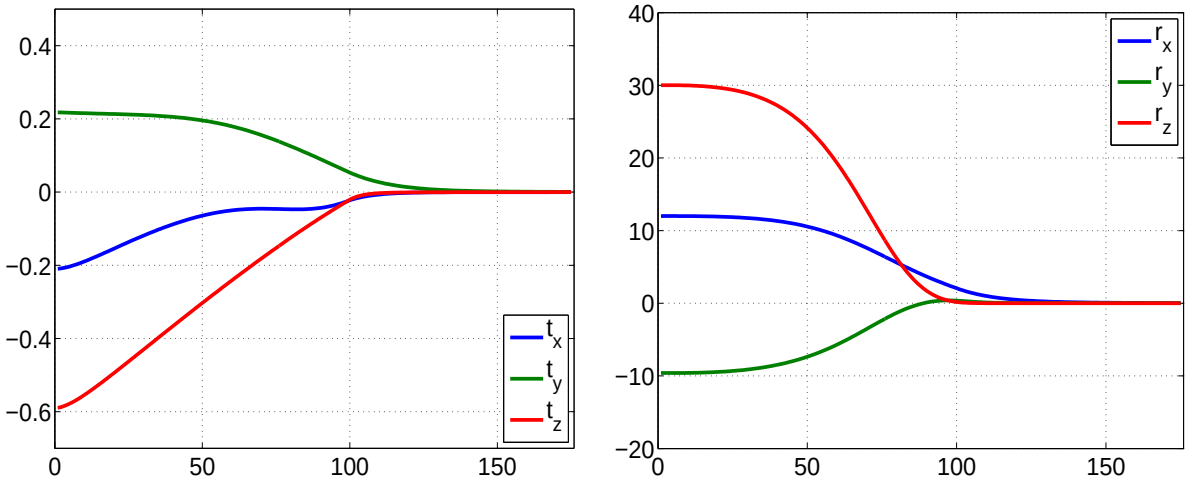


Reference image Initial image Final image



Translation velocity

Rotation velocity



Translation error

Rotation error

FIG. 4.8 – Direct visual servoing with respect to a non-planar target using an uncalibrated pinhole camera

4.2.2 Beyond standard vision-based control approaches

Standard vision-based control approaches have been widely studied and several successful applications have been performed. One may wonder if it is possible to go beyond the assumption behind these standard approaches. That is, can we achieve a task with an image-based approach despite the fact that the imaging conditions have changed ? How can we achieve a task using a monocular vision system with a position-based approach despite the fact that a metric model of the scene is unknown ? The following section describes two possible answers to these questions.

4.2.2.1 Visual servoing invariant to camera intrinsic parameters

The main assumption of the teaching-by-showing approach is that a reference image has been acquired and that the imaging conditions do not change. I proposed in [Malis 04b] to eliminate this assumption in order to enlarge the domain of application of the teaching-by-showing approach. More precisely, I showed that for pinhole cameras it is possible to define an error from image data that does not depend on camera intrinsic parameters. The error only depend on the pose of the camera with respect to the target. The invariant error can be defined for several types of visual features like points, lines, contours [Malis 01c, Malis 02e].

Consider for example that the camera observes $n > 3$ non-colinear points. The image homogeneous coordinates $\underline{\mathbf{p}}_{ij}$ depend on the camera internal parameters $\mathbf{K}_i$ and on the homogeneous normalized coordinates $\underline{\mathbf{q}}_{ij}$:

$$\underline{\mathbf{p}}_{ij} = \mathbf{K}_i \, \underline{\mathbf{q}}_{ij} \tag{4.2}$$

We can define two positive symmetric matrices $\mathbf{S}_q$ and $\mathbf{S}_p$ as follows :

$$\mathbf{S}_q = \frac{1}{n} \sum_{j=1}^{n} \underline{\mathbf{q}}_{ij} \underline{\mathbf{q}}_{ij}^\top = \mathbf{M}_q \mathbf{M}_q^\top \quad \text{and} \quad \mathbf{S}_p = \frac{1}{n} \sum_{j=1}^{n} \underline{\mathbf{p}}_{ij} \underline{\mathbf{p}}_{ij}^\top = \mathbf{M}_p \, \mathbf{M}_p^\top$$

where $\mathbf{M}_q$ and $\mathbf{M}_p$ are upper triangular matrices obtained applying the Cholesky decomposition to $\mathbf{S}_q$ and $\mathbf{S}_p$ respectively. The two matrices are related by :

$$\mathbf{M}_p = \mathbf{K}_i \mathbf{M}_q \tag{4.3}$$

Thus, it is possible to compute the invariant vectors $\mathbf{i}_{ij}$ with the following transformation :

$$\mathbf{i}_{ij} = \mathbf{M}_p^{-1} \underline{\mathbf{p}}_{ij} \tag{4.4}$$

Note that the vectors $\mathbf{i}_{ij}$ are computed only from the image coordinates. Using equations (4.3) and (4.2), we can prove that the vectors $\mathbf{i}_{ij}$ only depends on the normalized coordinates $\underline{\mathbf{q}}_{ij}$:

$$\mathbf{i}_{ij} = \mathbf{M}_p^{-1} \underline{\mathbf{p}}_{ij} = \mathbf{M}_q^{-1} \, \mathbf{K}_i^{-1} \, \mathbf{K}_i \, \underline{\mathbf{q}}_{ij} = \mathbf{M}_q^{-1} \underline{\mathbf{q}}_{ij} \tag{4.5}$$

As a consequence, the vectors $\mathbf{i}_{ij}$ are independent of the camera intrinsic parameters $\mathbf{K}_i$.

The transformation of the image points in the invariant space can be performed both for the reference image and the current image. In order to regulate the error to zero, I proposed a vision-based control technique based on the task function approach [Malis 01b, Malis 04b]. Another interesting property of the visual servoing invariant to camera intrinsic parameters is that it is possible to define trajectories in the invariant space such that the camera trajectory is a straight line in the Cartesian space [Malis 02d].

Figure 4.9 shows an example of invariant visual servoing performed with the calibrated Cartesian robot of the Lagadic research team at INRIA Rennes. The reference image is acquired with a 12 mm lens while a 6mm lens is used for servoing. The images in the last row of the figure show that the errors on the translation and the rotation of the end-effector converge to zero. If we look at the trajectory of the points in the image we can note that the final image (the green points) is obviously different from the reference image (the yellow points). On the other hand, the error on the coordinates of the points in the invariant space converge to zero.

Visual servoing with zooming cameras

An obvious extension of the invariant visual servoing approach was to show that the camera intrinsic parameters may vary during the servoing [Malis 02e]. Moreover, it is possible to define a method [Malis 02c] that unifies model-based approaches and image-based approaches. Since the error does not depend on the camera intrinsic parameters, a zooming camera can be used in order to keep the target in the camera field of view (zoom out) or to obtain a better resolution (zoom in) [Benhimane 03]. In this case it is possible to have two separate control laws for positioning the camera and for zooming. This approach allows the domain of applicability of visual servoing to be enlarged.

Indeed, being able to zoom out on the target can reduce visibility problems. However, when we zoom in to increase the resolution some features may disappear and other features may become visible. The appearance/disappearance of the visual information during the servoing produces discontinuities in the control law that perturb the performances of the system. Indeed, when one or several features appear or disappear during the servoing they are added to or deleted from the task function. The amplitude of the discontinuity depends on : i) the number of visual features, ii) the error in the image, iii) the conditioning of the interaction matrix. In [Garcia-Aracil 05a], we proposed a solution to avoid the discontinuities by giving a weight to each feature. We applied this approach to the invariant visual servoing [Garcia 04a], [Garcia-Aracil 05b], [Garcia 04b], but the approach can also be extended to standard image-based approaches.

3D Reconstruction with invariant active stereo-vision

The invariant visual servoing scheme has been used to develop an active vision method in order to improve the 3D reconstruction of natural 3D objects in the sea. We use a stereo pair mounted on a robot manipulator that lies on the ground. We constrain the displacements of the stereo pair by repeating the invariant visual servoing (the two cameras do not have the same intrinsic parameters) of the left camera on the right one

several times [Brandou 06]. Thus, the geometry of the stereo pair define the trajectory followed by the cameras. This simplifies the reconstruction process (constraint trajectory, reduction of the unknowns [Brandou 07]. We have also shown that the reconstruction process can be done in the invariant space reducing the number of unknowns of the bundle adjustment problem [Malis 04c].

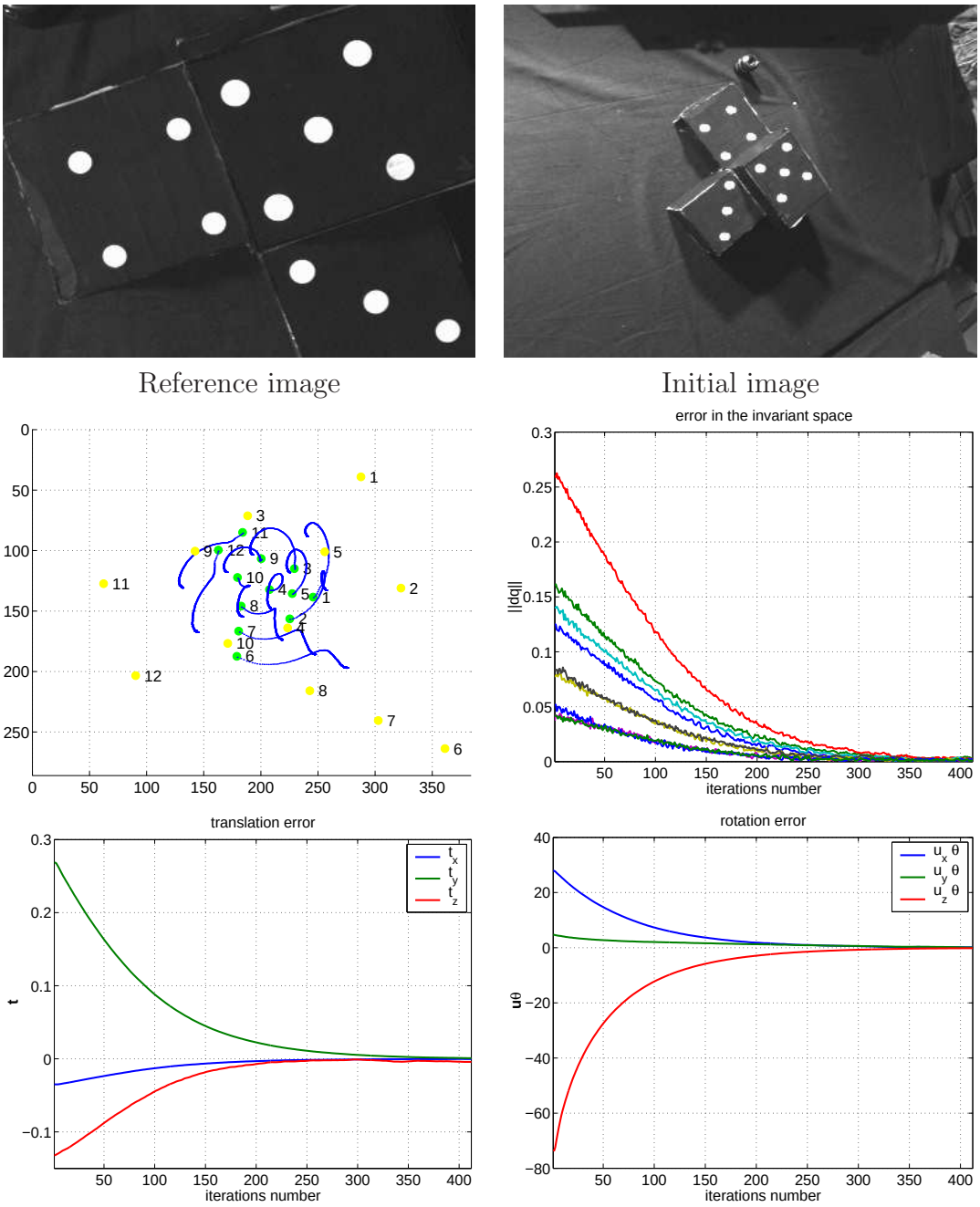


FIG. 4.9 – Intrinsics-free visual servoing using a 12mm lens for learning the reference image and a 6mm lens during the servoing.

4.2.2.2 Controlled visual simultaneous localization and mapping

In the standard "position-based" visual servoing approaches the reference pose with respect to a known target is directly given by the user [Wilson 96, Martinet 97]. Therefore, we need the model of the scene to compute the pose. We have extended this concept to visual servoing tasks when the model of the scene is not (completely) known [Silveira 06b]. Like standard "position-based" visual servoing approaches, the robot is directly controlled in the Cartesian space with a standard control law. However, the idea behind our approach is to perform a controlled visual SLAM. Indeed, the reference pose may be located very far away from the initial pose such that the same part of the scene may not always be maintained in the camera field of view as illustrated in Figure 4.10. Thus, newly observed parts of the rigid scene must be integrated into the scheme. The proposed approach is adapted to robot navigation in large environments.

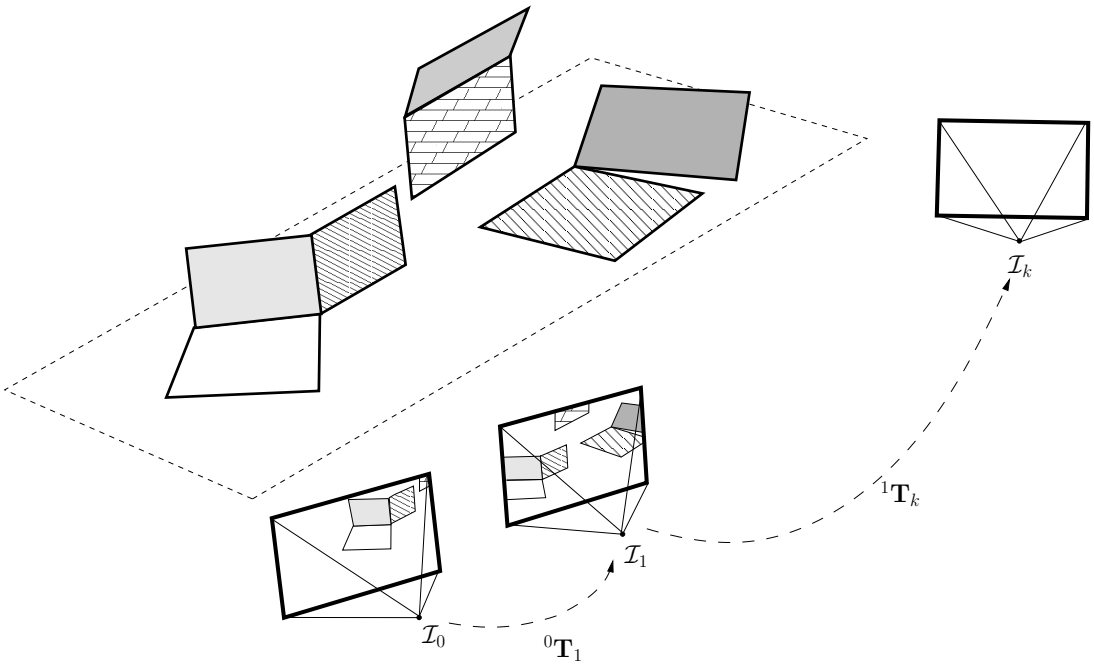


FIG. 4.10 – Controlled visual simultaneous localization and mapping.

The applicability of such an approach depends on the accuracy of the visual SLAM estimation. Since the task is uniquely defined in terms of a pose to be reached, small calibration and measurement errors may produce gross errors after long distances. The challenge is thus to obtain very precise estimations of structure and motion. We use our direct visual SLAM approach described in Section 3 to simultaneously estimate the structure and camera localization [Silveira 07c]. In order to speed up the on-line estimation we can suppose the scene is piecewise planar and we can detect the part of the scene that best fits our model. When a new part of the scene appears in the image the new planar regions are automatically detected [Silveira 06a]. Ath the time of writing, the experimental validation of the approach is in progress. However, promising simulation results can

be found in [Silveira 06b].

Conclusion and future research

When dealing with real-time robotic applications, it is important to take into account very early at the design level the respective specificities and requirements of vision-based estimation and vision-based control problems. By doing so, we have not only improved the performances during the execution of a task but also proposed methods and algorithms that perform better than approaches that consider vision-based estimation and vision-based control problems separately.

One important contribution has been to propose the Efficient Second-order approximation Method (ESM) for solving non-linear systems of equations defined on Lie Groups when additional information on the derivatives at the solution can be measured. Indeed, the ESM has been successfully applied both to vision-based estimation and to vision-based control. The theoretical results predict a higher convergence rate and a larger convergence domain with respect to standard numerical methods. These theoretical results have been confirmed by several experiments.

Another contribution has been the design of direct methods for image registration. This has permitted the accurate and robust estimation of structure and/or motion in large scale environments, despite occlusions and large changes in the illumination conditions. The proposed approach is generic, it can be used with any central catadioptric camera and it naturally extends to stereo-vision systems. The effective application of the ESM to our approach supposes that the Jacobians can be measured from image data only. We have shown when and how this can be possible using Lie groups theory. We have also experimentally shown that the ESM has very good performances even when the Jacobians are only approximated.

Various contributions have been made to the design of vision-based control schemes. We showed that standard image-based visual servoing schemes can be improved by using the ESM to design the control law. Theoretical stability and robustness analysis have allowed bounds on the "size" of the possible calibration errors to be found. Then, several contributions have been made to the design of robust vision-based control schemes that do not need an exact measure of the parameters of the vision system and that do not need any "a priori" knowledge of the structure of the observed scene. Again, a particular emphasis has been placed on the theoretical analysis of the robustness of the control laws with respect to errors on the parameters. Finally, we have proposed methods that

go beyond the teaching by showing approach. We have shown how to define an error function for pinhole cameras that is invariant to camera intrinsic parameters. This allows us to define new tasks using zooming cameras. Another new promising approach is to integrate vision-based control into simultaneous localization and mapping approaches.

Future Research

In my future research work, I intend to develop a low level vision framework as generic as possible and well adapted to robotic applications. This framework must be able to address a large number of situations frequently met : different sorts of cameras (central or not), large illumination changes, rigid/deformable objects, static/dynamic scenes, etc. The low level vision framework should also be generic enough to be coupled with other interoceptive and/or exteroceptive sensor information in order to obtain a reliable and robust localization for robotic applications. I believe that one way to achieve these objectives is to further study and develop direct methods. Within this framework, three main problems must be addressed.

The first problem is to find simple, accurate, yet general models of the plenoptic function. In particular, we need general photometric models, general camera models and general structure models. General photometric models should be able to take into account not only illumination changes but also colorimetric information. General camera models should be able to take into account non-central projection cameras. General structure models should be able to take into account not only complex rigid environments but also unknown deformable environments. These models should be realistic, but not overly complex in order to comply with computational real-time requirements of robotic applications. Finding the right compromise is a delicate and difficult issue. Indeed, the qualities of a model depend on the problem for which it is used (simulation, estimation, control), and a solid methodology (which could guide the choices in this respect) does not exist.

The second problem is the efficient estimation of the parameters associated with the models. The optimization techniques efficient enough for real-time robotic tasks have only local convergence. Thus, a fairly accurate initialization is needed (i.e. an estimate of the true unknown parameters). We can study how to efficiently initialize the local optimization schemes using fast data association algorithms or relying on external information (complementary sensor modalities). However, a more general and elegant solution would be the study of efficient global optimization methods. For example, I am currently investigating if and how the use of Interval Analysis as a global optimization tool can be compatible with real-time constraints. With respect to other global optimization techniques, the main advantage of Interval Analysis is that the solutions to the problem can be certified.

The third problem is the use of complementary sensor modalities and fusion techniques. This allows the localization and mapping tasks to be simplified and robustified

when dealing with large scale environments. For example, a typical multi-sensor system is composed of one or more vision sensors and/or a range sensor (laser or radar) with the complement of an absolute positioning sensor (e.g. an inertial navigation system or a global navigation system). To cope with multi-sensor data, I intend to relies on the direct approach for each sensor, and to study appropriate filtering techniques in order to take into account the uncertainty of each sensor.

Another objective of my future research work will be to further investigate the direct vision-based robot control approach. The design of standard vision-based control schemes rely on several assumptions : on the geometric and photometric properties of the environment, on the characteristics of the imaging device and on the characteristic of the robot. My objective is to increase the precision and the robustness of direct vision-based control algorithms despite modeling errors induced by these assumptions. On the one hand, I will investigate the design of control algorithms that are robust with respect to noise and modeling errors and analyze the associated robustness properties. On the other hand, I will study the connections between the design of robust control laws and the system modeling, with the objective of investigating if and how the assumptions on the geometric and photometric properties of the environment, on the characteristics of the imaging device and of the robot can be relaxed. A key point will be to handle the compromise between the complexity of the models, the real time aspects and the robustness of the control algorithms.

Concerning the assumptions on the vision system there are three assumptions that we should relax. The first assumption to be relaxed should be the use of central cameras. Indeed, non-central cameras can be useful in many applications since some of them have constant resolution. To my knowledge, the use of non-central cameras for direct vision-based control has not yet been investigated. A second assumption to be relaxed should be the rigidity of the environment. Thus, I will investigate whether control methods which do not involve an explicit metric reconstruction of a model of the environment and have been proposed for rigid objects, can be applied to deformable ones. The key problem is the characterization of a diffeomorphism between the sensory information and the pose of the robot. The third assumption is that the environment is Lambertian. This assumption is obviously not verified in practice and the estimation of the position of the light sources in the environment can be used to increase the accuracy and robustness. Note that this information can also be taken into account in the design of the control law in order to avoid the presence of specular reflection or shadows in the image.

Concerning the assumption on the characteristic of the robot, I have considered so far vision-based control applied to holonomic mechanisms. The control of these systems is much simpler due to the fact that instantaneous motion along any direction of the configuration space is possible. A consequence of this is that generic vision-based control laws can be designed independently of the mechanism specificity. However, this is not true in the case of critical systems like most ground, marine or aerial robots. I have proposed control schemes for holonomic robots whose performance does not depend on

parameters that cannot easily be estimated on-line (e.g. camera intrinsic parameters, pose and 3D parameters of the target). Such control laws are structurally robust with respect to estimation errors on the parameters. I plan to study whether these techniques can be generalized to critical nonlinear systems.

Appendix : Resumé en français

Introduction

La conception de systèmes robotiques complètement autonomes est une des grandes ambitions de l'humanité. De nos jours des progrès théoriques et technologiques ont permis aux ingénieurs de concevoir des systèmes robotiques complexes qui peuvent remplacer les êtres humains dans de nombreuses applications qui peuvent être répétitives ou dangereuses. Toutefois, afin d'étendre la flexibilité et le domaine d'application de ces systèmes autonomes, nous devons encore résoudre plusieurs problèmes scientifiques qui sont à la croisée de plusieurs domaines tel que l'intelligence artificielle, le traitement du signal et la théorie de la commande.

Parmi ces défis scientifiques, la perception de l'environnement et l'interaction d'un système robotique avec son environnement sont des problèmes fondamentaux dans la conception de tels systèmes autonomes. En effet, les performances d'un robot autonome dépendent non seulement de la précision, de la continuité et de la fiabilité de sa perception mais aussi de son habilité à utiliser l'information perçue dans des boucles de commande afin d'interagir de manière sûre avec son environnement malgré les inévitables erreurs de modélisation et de mesure. Pour ces raisons, la modélisation et la perception de l'environnement et la commande référencée capteur robuste sont des problèmes scientifiques majeurs en robotique.

Plusieurs capteurs extéroceptifs sont couramment utilisés en robotique : GPS, capteurs optiques, capteurs de contacts, sonars, télémètres lasers, et même des capteurs auditifs et olfactifs. Toutefois, la vision artificielle a un intérêt et une importance particulière principalement due à sa versatilité. Elle peut être utilisée à la fois pour la modélisation et la perception et pour la commande du robot. Dans ce contexte, l'étendue des recherches est très vaste et je focaliserai ce document sur l'estimation et la commande à partir d'un système de vision. L'estimation à partir de la vision concerne des méthodes dédiées à l'extraction des informations qui peuvent être utilisées à la fois pour la modélisation de l'environnement et pour la commande des robots. Dans le cas des applications robotiques, les défis majeurs sont d'augmenter l'efficacité, la précision et la robustesse de l'estimation à partir des données capteurs. La commande référencée vision concerne des méthodes dédiées à l'utilisation des informations visuelles dans des boucles de commande. Les défis sont le choix approprié de l'information et la conception de lois de commande stables et robustes aux erreurs de modélisation et de mesure.

L'objectif de ce document est de résumer, analyser et discuter mes travaux de recherche sur ces sujets pendant les derniers dix années : deux ans comme Research Associate à l'Université de Cambridge (Grande Bretagne) et huit ans comme Chargé de Recherche à l'INRIA Sophia-Antipolis (France). Afin de reconnaître les contributions de tous les collègues et les étudiants avec lesquels j'ai collaboré, j'ai essayé autant que possible de citer toutes nos publications en commun.

Activités de recherche

Mes activités de recherche se situent à la croisée de la vision par ordinateur et de la commande référencée capteurs des systèmes dynamiques. La principale problématique abordée dans mes travaux est comment utiliser les informations capteur dans des boucles de commande afin de rendre un robot le plus autonome possible dans la réalisation des tâches de type réflexe.

Pendant mon travail de post-doctorat, qui a été effectué à l'Université de Cambridge dans le cadre du projet européen VIGOR, j'ai tout d'abord généralisé une partie de mes travaux de thèse à une nouvelle classe de méthodes d'asservissement visuel. Ensuite, j'ai apporté une solution au problème de la robustesse des lois de commande lié aux erreurs d'étalonnage de la caméra. Enfin, j'ai traité le problème de la conception d'un système complet de commande référencée vision.

Une nouvelle classe de méthodes d'asservissement visuel

Une partie très importante de mon travail de post-doctorat a été de montrer que la méthode d'asservissement visuel proposée dans ma thèse est un cas particulier d'une classe plus large de schémas d'asservissement visuel [Malis 02f]. Par exemple, dans la même classe on peut concevoir un schéma d'asservissement de type "asservissement visuel 3D" en définissant comme fonction de tâche une erreur sur l'attitude de la caméra. Cette nouvelle classe possède des propriétés très intéressantes. En particulier, il est possible d'effectuer une analyse formelle de la stabilité et de la robustesse par rapport aux paramètres intrinsèques et extrinsèques de la caméra (le paramètres extrinsèques définissent l'attitude de la caméra par rapport a l'effecteur du robot) [Malis 01a]. L'importance de l'analyse de la robustesse réside dans la connaissance du taux d'erreur d'étalonnage que le système peut supporter.

Auto étalonnage des caméras

Bien que la loi de commande proposée s'avère assez robuste, si les erreurs d'étalonnage sont très grands le système peut avoir des performances réduites et il peut même devenir instable. Une solution classique est de bien étalonner la caméra préalablement. Cette étape est importante mais elle est souvent difficile à mettre en oeuvre et peu flexible. La deuxième possibilité, plus flexible, est d'auto étalonner la caméra en ligne. Dans ce

contexte, ma contribution principale a été de formuler le problème de l'auto étalonnage à partir d'objets planaires d'une manière très intuitive du point de vue géométrique [Malis 02g]. L'auto étalonnage est obtenu en imposant la contrainte que la scène observée est rigide. Donc, la normale au plan dans un repère fixe est constante quelque soit le point de vue [Malis 00b]. L'intérêt de l'auto étalonnage est que, sous certaines conditions, on peut estimer la distance focale de la caméra même si elle change au cours du temps [Malis 00c].

Conception d'un système de commande référencée vision

Un système complet d'asservissement visuel ne se réduit pas au choix de la fonction d'erreur à réguler et à la conception de la loi de commande mais doit tenir compte aussi d'un système de mesure de l'erreur [Malis 03b]. En collaboration avec G. Chesi nous avons proposé un système complet pour la commande référencée vision par rapport à des contours plans. Dans ce système, nous avons intégré l'association des données (mise en correspondance et suivi) et la commande [Chesi 00]. La mise en correspondance consiste à localiser dans deux images les projections d'un même pont de la scène. Notre contribution a été de proposer un algorithme pour la mise en correspondance de contours plans [Chesi 99]. Ce travail d'intégration est très important pour la suite de mes recherches car il a mis en évidence les limites des approches classiques de conception des systèmes de commande référencée vision (qu'on peut classifier comme "réductionnistes"). La loi de commande utilisée n'était pas assez générique car elle nécessite une connaissance sur la normale au plan. De plus, le système de mesure était spécifique à des contours plan et difficilement réutilisable dans d'autres applications.

A la suite de mon recrutement à l'INRIA en Octobre 2000 j'ai poursuivi mes travaux de recherche sur la commande référencée vision. Tout d'abord, afin d'augmenter la robustesse et la flexibilité des méthodes de type "asservissement visuel 2D", j'ai travaillé sur la conception des nouvelles méthodes d'asservissement et sur la vision active. Ensuite j'ai travaillé sur l'extension du domaine d'application de la commande référencée vision afin de passer des tâches de positionnement d'un robot manipulateur aux tâches de navigation d'un robot mobile en environnement inconnu. C'est dans ce contexte que me travaux de recherche ont peu à peu évolué vers une approche plus holistique à l'intégration des algorithmes de vision par ordinateur dans les système de commande référencée vision. Dans ce cadre, j'ai développé un nouvel axe de recherche transversal concernant des méthodes d'optimisation efficaces et robustes.

Travaux de recherche en asservissement visuel 2D

Une contribution très importante de mes travaux a été de proposer une extension de l'approche "apprentissage par démonstration" : l'asservissement visuel 2D invariant aux paramètres intrinsèques de la caméra. Mes travaux de recherche sur les méthodes d'asservissement visuel 2D ont été complété par une analyse de la robustesse des loi de commande et une étude sur les problèmes de visibilité et de continuité.

Extension de l'approche "apprentissage par démonstration" : l'asservissement visuel 2D invariant aux paramètres intrinsèques

En l'absence d'une connaissance métrique complète d'un objet par rapport auquel on veut se positionner, les méthodes d'asservissement visuel se basent sur l'apprentissage préalable d'une image de celui ci (l'approche "apprentissage par démonstration"). A partir de cette image l'on extrait des caractéristiques visuelles (point, droites, contours, ...) qui seront utilisées comme amers "référence" à atteindre dans l'image. A partir d'une position initiale différente, le mouvement du robot (et donc de la caméra embarquée) est contrôlé afin que les caractéristiques visuelles observées coïncident avec celles de "référence". Dans ce cas, la caméra se trouvera, par rapport à l'objet observé, dans la même position que celle de l'apprentissage. De manière générale, quelque soit la méthode d'asservissement visuel employée, ceci est vrai si et seulement si les paramètres intrinsèques de la caméra sont, à la convergence, les mêmes que lors de l'apprentissage de l'image de référence. En effet, il est évident que si les paramètres intrinsèques de la caméra changent pendant l'asservissement (ou si la caméra utilisée pendant l'asservissement n'est pas la même caméra utilisée pendant l'apprentissage), même si on arrive à atteindre les informations visuelles de référence (ce qui n'est généralement pas possible) la position de la caméra sera très différente de sa position à l'apprentissage. Par exemple, si on double la focale, la distance par rapport à un objet doit être réduite de moitié afin d'observer à peu près la même image. L'hypothèse selon laquelle "les paramètres intrinsèques de la caméra à la convergence doivent être les mêmes que lors de l'apprentissage de l'image de référence" est très contraignante.

J'ai donc proposé d'éliminer cette contrainte afin d'augmenter considérablement les domaines d'application de l'asservissement visuel [Malis 04b]. Plus précisément, j'ai montré qu'il est possible de définir une erreur à partir des données image qui ne dépende pas des paramètres intrinsèques de la caméra avec laquelle l'image a été prise [Malis 01c]. Cette erreur ne dépend que de la position de la caméra par rapport à l'objet observé. Afin de contrôler le mouvement de la caméra et d'asservir cette erreur à zéro, j'ai proposé une technique d'asservissement visuel 2D basée sur l'approche par fonction de tâche [Malis 01b]. L'asservissement visuel invariant aux paramètres intrinsèques de la caméra est particulièrement intéressant car il est possible de définir des trajectoires dans l'espace invariant sans aucune connaissance des paramètres intrinsèques. Par exemple, il est possible de définir une trajectoire dans l'espace invariant telle que la trajectoire de la caméra dans l'espace cartésien soit une ligne droite [Malis 02d]. La loi de commande qui permet la poursuite de trajectoire est définie directement dans l'espace invariant. Elle doit, autant que possible, présenter une grande robustesse aux incertitudes sur les paramètres du système et garantir un grand domaine de convergence. Dans le cas de notre nouvelle approche d'asservissement visuel par rapport aux incertitudes sur la calibration, nous avons montré que le domaine de robustesse est suffisamment grand pour permettre l'utilisation d'une approximation grossière des paramètres de calibration tout en conservant la convergence et la stabilité de l'asservissement.

Une extension évidente de l'approche a été de montrer que les paramètres intrinsèques de la caméra peuvent varier pendant l'asservissement [Malis 02e]. En plus, il est possible

de définir une méthode [Malis 02c] qui unifie les approches classiques proposées dans la littérature : celle basée sur la connaissance du modèle 3D de l'environnement et celle basée sur un apprentissage préalable de l'image de référence. De plus, avec cette nouvelle méthode il est possible de pallier à un certain nombre de limitations de l'asservissement visuel classique. Du fait que l'erreur ne dépende pas des paramètres intrinsèques de la caméra avec laquelle l'image a été prise, le zoom de la caméra peut alors être utilisé afin de garder un objet dans l'image (zoom arrière) ou afin d'augmenter sa taille dans l'image (zoom avant) [Benhimane 03]. Dans ce cas, il est possible de séparer l'asservissement visuel de la caméra, de la loi de commande utilisée pour contrôler le zoom. Cette approche permet d'augmenter considérablement les domaines d'application de l'asservissement visuel.

Analyse de la stabilité et de la robustesse de l'asservissement visuel 2D

Malgré ses propriété d'invariance aux paramètres intrinsèques de la caméra, l'asservissement visuel proposé appartient à la catégorie des méthodes dite d'asservissement visuel 2D. Dans ce type de méthodes la fonction de tâche peut être calculée directement à partir des caractéristiques visuelles extraites des images. Toutefois, la matrice d'interaction qui lie la dérivée de la fonction de tâche à la vitesse de la caméra dépend de l'information métrique de l'objet. Une estimation de cette matrice est nécessaire afin de concevoir une loi de commande stable. Il a été observé expérimentalement qu'une approximation grossière de l'information métrique de l'objet est suffisante pour obtenir une loi de commande stable. Toutefois, une question qui n'était pas résolue est de savoir quelle erreur peut on commettre sur l'approximation de cette structure tout en garantissant la stabilité de la loi de commande. A cause de la difficulté de l'analyse très peu de résultats théoriques ont été obtenus en littérature concernant la stabilité. Cette analyse a pu être accomplie seulement pour des simple cas, souvent considérant un modèle simplifié de caméra, et toujours en supposant que la distribution des profondeurs est estimée parfaitement. Nous avons étudié la robustesse des méthodes d'asservissement 2D par rapport aux incertitudes sur les paramètres de la structure [Malis 03a] [Malis 02a]. Nous avons prouvé théoriquement que même des erreurs relativement petites sur les paramètres 3D peuvent rendre les lois de commande instables [Malis 03c]. La preuve a été étendue à n'importe quelle caméra à centre de projection unique dans [Mezouar 04].

Visibilité et continuité de l'asservissement visuel 2D

Dans l'asservissement visuel 2D les informations extraites des images courante et de référence sont "combinées" dans une fonction de tâche qui a la même dimension que l'espace des configurations du robot (6. d.d.l. pour un robot manipulateur). Toutefois de nombreuses informations visuelles peuvent ne plus être disponibles pendant la tâche (occultation, sortie de l'image, etc..) et l'être à nouveau après un certain délai. La disparition/apparition d'informations visuelles pendant l'asservissement produit des discontinuités dans la loi de commande qui perturbent les performances du système. En effet, quand une ou plusieurs informations disparaissent ou apparaissent pendant l'asservissement elles sont enlevées ou ajoutées de la fonction de tâche. Ceci produit une discontinuité dans la loi de commande. L'amplitude de cette discontinuité dépend i) du nombre

d'amers, ii) de l'erreur dans l'image, et iii) du conditionnement de la matrice d'interaction. En collaboration avec N. Garcia, nous avons proposé une solution en [Garcia-Aracil 05a] afin d'éviter ces discontinuités en associant un poids à chaque amer. En particulier, nous avons adapté l'asservissement visuel invariant afin de prendre en compte les variations qui arrivent pendant l'asservissement [Garcia 04a] [Garcia-Aracil 05b] [Garcia 04b] .

Travaux de recherche en vision active

La vision active consiste à contrôler les paramètres intrinsèques ou extrinsèques du système de vision afin d'améliorer le processus de perception. Par exemple, afin d'améliorer la résolution de l'image observée on peut envisager soit de changer la distance focale soit d'approcher la caméra de l'objet. Dans ce contexte, j'ai étudié comment utiliser l'asservissement visuel afin d'améliorer la précision de l'estimation de la structure 3D. Cette information peut être ensuite utilisée dans la boucle de commande référencée vision.

Reconstruction affine de la structure 3D à un facteur d'échelle près

La reconstruction affine est une solution possible au problème de l'estimation de la structure 3D (à un facteur d'échelle près) car elle permet d'estimer la distribution des profondeurs d'un ensemble de points tout en se basant sur des hypothèses plus flexibles que pour la reconstruction Euclidienne. Tout d'abord, une calibration précise du robot et/ou de la caméra n'est pas requise. Ensuite seulement deux images de l'ensemble des points considérés sont suffisantes pour obtenir l'estimation. En effet, la reconstruction affine est une tâche facile quand le déplacement entre les deux images est une translation pure. La direction et l'amplitude de cette translation peuvent être arbitraires et inconnues. Une translation pure en boucle ouverte est théoriquement possible seulement si le robot est parfaitement étalonné. Au contraire, nous avons proposé une reconstruction affine active où la translation pure est réalisée en contrôlant le robot avec la commande référencée vision [Malis 03d]. Nous définissons les contraintes que les informations visuelles dans l'image de référence doivent atteindre afin que le robot réalise une translation pure entre l'image initiale et finale. La tâche d'asservissement visuel est spécifiquement conçue afin de réaliser des mouvements de la caméra qui simplifient l'estimation de la structure. Une fois que la translation pure a été obtenue la reconstruction affine à partir des images initiale et finale est triviale.

Reconstruction euclidienne de la structure 3D à un facteur d'échelle près

La reconstruction euclidienne de la structure 3D à un facteur d'échelle près est une tâche plus difficile car le mouvement de la caméra n'est plus contraint à une translation pure. Nous avons alors modifié les objectifs de la tâche d'asservissement visuel afin d'effectuer un mouvement qui permette d'obtenir la structure sans trop perturber la tâche principale de positionnement [Vargas 05]. PAI PICASSO.

Reconstruction 3D avec une paire stéréo active

Dans le cadre du projet THEMIS avec IFREMER, en collaboration avec V. Brandou nous avons développé une méthode de vision active afin d'améliorer la reconstruction 3D

d'objets rigides naturels dans les fonds marins. Nous utilisons une paire stéréo embarquée sur un robot manipulateur posé sur le fond marin. Nous contraignons les déplacements de la paire stéréo en réitérant un asservissement visuel de l'image gauche sur l'image droite (les deux caméras ne sont pas les mêmes) avec une méthode d'asservissement invariante aux paramètres des caméras [Brandou 06]. De ce fait, la géométrie de la paire stéréo détermine la trajectoire suivie par les caméras. Ceci simplifie le processus de reconstruction (réduction du nombres d'inconnues, trajectoire contrainte, ...) [Brandou 07].

Travaux de recherche en optimisation efficace et robuste

Un nouvel axe de recherche transversale concerne l'optimisation efficace et robuste. En effet, le problèmes d'association des données et de commande peuvent être formulés comme des problèmes d'optimisation. Une caractéristique commune à ce type de problèmes est que souvent nous pouvons avoir une estimation des dérivées premières du vecteur d'erreur à la solution. Il est donc possible d'utiliser cette information supplémentaire afin d'améliorer les performances des technique d'optimisation. J'ai donc proposé la méthode ESM (efficient second-order method) qui permet d'obtenir une vitesse de convergence plus rapide que la méthode de Newton classique. La méthode ESM a été appliquée à la fois à la commande référencée vision et à l'association des données (plus particulièrement au suivi visuel).

Application à la commande référencée vision

Les problèmes potentiels de mouvement transitoire de l'asservissement visuel 2D sont dûs au fait que l'on utilise, pour des tâches de positionnement commençant très loin du point d'équilibre, des lois de commande qui sont valables seulement localement. Dans ce cas, des mouvements transitoires non souhaitables du robot peuvent se produire (comme par exemple une translation très grande alors qu'une simple rotation suffit pour réaliser la tâche). J'ai montré qu'une partie des problèmes de l'asservissement visuel 2D peut être résolue en utilisant une loi de commande inspirée d'algorithmes d'optimisation ESM [Malis 04a]. En effet, les méthodes de commande référencée vision peuvent être classifiées en suivant une analogie avec les méthodes classiques de minimisation. Les méthodes de commande sont généralement basée sur des approximations au premier ordre comme des simple méthode de descente de gradient. Théoriquement, les performances des lois de commande peuvent être améliorées en utilisant des approches possédant les mêmes propriétés que la méthode de Newton qui implique une approximation au second ordre de la fonction de coût. Malheureusement l'utilisation des dérivées secondes (les matrices Hessiennes) peut être mal conditionnée, provoquant ainsi des problèmes de convergence. Au contraire, le schéma de commande basé sur la technique d'optimisation ESM, possède des meilleures propriétés de convergence ainsi qu'un plus grand domaine de convergence.

Application à l'association des données (suivi visuel)

En collaboration avec S. Benhimane nous avons appliqué la méthode ESM au problème du suivi visuel [Benhimane 04]. Un algorithme de suivi temps-réel est une partie fonda-

mentale d'un système complet de commande référencée vision. Son rôle est de mettre en correspondance les informations visuelles extraites à partir de deux images successives en supposant que le déplacement dans l'image n'est pas très grand. Au cours de mes recherches j'ai utilisé plusieurs algorithmes de suivi de caractéristiques visuelles : suivi de points d'intérêts [Espiau 02], de lignes [Mei 06b], de contours [Malis 03b]. Malheureusement, les méthodes de suivi d'amers visuels sont sensibles aux échecs de détection et aux erreurs de mesure. En plus, le choix d'un algorithme adéquat de suivi dépend du type d'amer qui est visible dans la scène. Par exemple, un algorithme conçu pour suivre des cercles dans l'image ne peut pas être utilisé pour suivre des objets qui ont seulement des points d'intérêt. Au contraire, les algorithmes directs ne dépendent pas de caractéristiques images. Pour cette raison ces méthodes sont mieux adaptées à l'asservissement visuel. Nous avons alors proposé une méthode d'alignement d'images directe pour des objets plans ayant un texture quelconque. Au lieu d'extraire des amers visuel comme des points, lignes, ou contours, l'homographie qui caractérise le mouvement 3D et la géométrie du plan est estimée directement à partir des intensités de la région d'intérêt. L'optimisation ESM robuste s'avère particulièrement efficace afin d'éviter la défaillance de la minimisation en présence de mesures aberrantes [Malis 06]. Cette nouvelle technique d'alignement d'images basée sur l'optimisation ESM permet d'expliquer les excellentes performance d'une technique d'alignement d'images proposée dans le domaine de l'imagerie médicale [Vercauteren 07].

Les travaux de recherche sur l'optimisation efficace ont ouvert des nouvelles perspectives pour la conception des approches d'asservissement visuel. Plus en particulier, une approche plus holistique est apparue comme la meilleur solution au problème de l'intégration des algorithmes de perception visuelle de de commande.

Une approche holistique à la commande référencée vision

L'approche standard pour la conception des approches d'asservissement visuel est d'assembler des méthodes de vision par ordinateur et de commande qui ont été conçues séparément. Avec cette approche, l'intégration d'un système complet peut être difficile vu le grand nombre de techniques différentes pour le suivi et l'asservissement. Mais aussi elle peut s'avérer peu flexible. Au lieu de considérer les systèmes de vision et de commande séparément nous proposons d'unifier autant que possible des travaux en vision par ordinateur et asservissement visuel dans un cadre unifié [Malis 05]. Mon objectif est de construire un système générique, robuste et flexible qui peut être utilisé pour une très grande variété d'applications robotiques comme par exemple la commande référencée vision de robot manipulateur [Malis 04d] ou celle de robots mobiles [Benhimane 05]. L'idée principale est d'utiliser les méthodes de perception directes afin d'extraire des données images la mesure de la fonction d'erreur à réguler à zéro. Au même temps, il faut concevoir des lois de commande référencée vision qui ne nécessitent que des informations estimées par le suivi visuel.

Association des données (suivi visuel)

Le travail de recherche sur le suivi visuel direct, initié par S. Benhimane (voir travaux de recherche en optimisation efficace), à été poursuivi en collaboration avec C. Mei et G. Silveira. Nous avons tout d'abord étendu l'algorithme de suivi visuel à tous les capteurs à centre de projection unique [Mei 06a]. Le capteurs à projection centrale unique incluent les caméras omnidirectionnelles qui sont particulièrement utiles pour plusieurs applications robotiques grâce à leur champ de vue a 360 degrés. Ensuite, nous avons propose une extension de la méthode afin de prendre en compte les changements d'éclairage quelconques [Silveira 07b]. Avec une information supplémentaire sur le facteur d'échelle, l'algorithme de suivi visuel ESM peut être appliqué à l'estimation précise du mouvement d'un robot mobile a partir des l'observation de plusieurs objets plans par une caméra perspective [Benhimane 06b] ou omnidirectionnelle [Mei 06c]. Une approximation planaire par morceaux de la scène pourrait ne pas être appropriée pour certaines applications. Pour cette raison, nous avons commencé à étudier comment étendre le suivi visuel ESM à des objets 3D génériques.

Asservissement visuel

L'intégration des algorithmes de suivi visuel dans les schémas de commande devient très facile si le schéma d'asservissement visuel peut utiliser directement et uniquement la sortie du suivi. En collaboration avec S. Benhimane nous avons étudié comment concevoir une lois de commande qui ne dépendent pas des mesures sur la structure de la cible [Benhimane 07]. Comme dans le cas du suivi visuel, il s'agit d'une approche de commande générique qui peut être applique aussi bien a des caméras perspectives [Benhimane 06a] que à des caméras omnidirectionnelles [Benhimane 06c]. La loi de commande proposée est stable localement et, contrairement à toutes les méthodes existantes, elle ne nécessite pas de mesure du modèle de l'objet par rapport auquel la commande est effectuée. Seules des informations issues des images de référence et courante suffisent pour calculer la loi de commande.

Le passage des tâches de positionnement aux tâches de navigation en environnement inconnu demandent des algorithmes de perception capables de reconstruire en ligne l'environnement et de localiser le robot par rapport a un repère de référence. C'est pourquoi ces dernières années mes travaux de recherche se sont oriente vers la reconstruction en temps-réel de la structure et du mouvement (odomètrie visuelle et SLAM). Ainsi l'extension de l'approche "apprentissage par localisation" classique à de tâches de navigation vers des lieux jamais explorés devient possible.

Reconstruction de la structure et du mouvement

Odomètrie Visuelle Stéréo

Dans le cadre du projet MOBIVIP et en collaboration avec A. Comport nous avons proposé une technique d'odomètrie visuelle pour localiser une paire stéréo par rapport à un repère fixe attaché à l'environnement [Comport 07]. Nous utilisons une approche de

minimisation dense qui utilise toute l'information disponible de la paire stéréo (ou d'une région) qui produit des résultats précis et robustes. Les contraintes de rigidité sur la structure 3D sont imposées en reprojetant de manière consistante les images de la paire stéréo afin de générer une nouvelle couple d'images. Une estimation non-linéaire de la trajectoire est formulée sur la base de la relation quadrifocale entre les intensités des images courantes et de référence de la paire stéréo. Nous utilisons la technique d'optimisation ESM robuste (avec des M-estimateurs) afin de rejeter les mesures aberrantes correspondant à des objets en mouvement dans la scène ou d'autres mesures aberrantes comme des occultations, changement d'illumination et erreurs de mise en correspondance. La technique est générique et elle a été appliquée aussi bien à l'estimation de trajectoire d'un véhicule dans un environnement urbain qu'à l'estimation de trajectoire d'un dirigeable évoluant en 3D.

Localisation et reconstruction simultanée à l'aide d'un système de vision

Le SLAM (Simultaneous Localisation and Mapping) consiste à estimer l'attitude de la caméra tout en imposant une cohérence temporelle sur la rigidité de la scène. Dans le cadre du projet AURORA et en collaboration avec G. Silveira, nous avons proposé une méthode efficace qui calcule directement le déplacement 3D de la caméra et la structure de la scène [Silveira 07c]. Il est important de noter que la méthode étant basée sur l'utilisation directe des valeurs d'intensités de tous les pixels dans les régions d'intérêt. Les paramètres de mouvement et de la structure sont simultanément estimés par une technique d'optimisation ESM pour un traitement des données plus rapide et permettant d'éviter des minimum locaux non significatifs. Par ailleurs, la rigidité de la scène et le fait qu'elle soit située face à la caméra sont imposés en tant que contraintes dans l'optimisation. Tous ces facteurs contribuent à obtenir des résultats très précis.

Extension de l'approche "apprentissage par localisation"

Dans le cadre du projet AURORA et en collaboration avec G. Silveira, nous avons proposé une nouvelle méthode d'asservissement visuel pour positionner une caméra dans le cadre de l'approche "apprentissage par localisation". Notre approche d'asservissement visuel ne nécessite ni une image de référence ni une mesure métrique a priori du modèle de la scène [Silveira 06b]. La tâche est définie uniquement en terme d'une attitude à atteindre. L'attitude courante est obtenue par "odomètrie visuelle". Afin d'améliorer la précision la stabilité et la vitesse de convergence de l'estimation nous modélisons la scène comme un ensemble de régions planes. Le schéma d'asservissement visuel est basé sur deux techniques. Premièrement, il est basé sur un détecteur de régions planaires [Silveira 06a] de telle manière à détecter des nouvelle régions d'intérêt quand celles déjà utilises sortent du champ de vue de la caméra. Deuxièmement, le suivi visuel ESM est utilisé afin d'estimer en ligne la structure et le mouvement de la caméra simultanément. Cette approche est bien adaptée à des applications temps-réel comme la navigation visuelle dans des environnements très grands.

Conclusion et perspectives de recherche

Quand l'on considère des applications robotiques en temps-réel, il est important de prendre en compte très tôt, au niveau de la conception, les spécificités et les caractéristiques respective des problèmes d'estimation et de commande. Ce faisant, non seulement nous avons pu améliorer les tâches classiques d'asservissement visuel mais aussi nous avons pu proposer des méthodes qui sont plus performantes des méthodes qui considèrent l'estimation et la commande comme deux problèmes séparés.

Une contribution importante a été de proposer la méthode dite Efficient Second-order approximation Method (ESM) pour la solution de systèmes d'équations non linéaires, définis sur des groupes de Lie, quand l'on peut mesurer une information supplémentaire sur les dérivée à la solution. En effet, la méthode ESM a été appliquée avec succès à la fois pour l'estimation et la commande à partir des données visuelles. L'analyse théorique prévoit une vitesse de convergence plus rapide et un domaine de convergence potentiellement plus grand par rapport aux méthodes classiques. Ces résultats théoriques ont été confirmes par des nombreuses expériences.

Une autre contribution a été la conception de méthodes directe pour l'alignement d'images. Ceci a permis une estimation précise et robuste de la structure et/ou de la localisation dans des environnement à large échelle, malgré des occultations et des grands changements des conditions d'éclairage. L'approche que nous avons propose est générique, elle peut être utilise avec n'importe quelle caméra à centre de projection unique et elle s'étend naturellement à des systèmes de stéréo vision. L'application de l'ESM à notre approche suppose que les Jacobiens peuvent être mesurés seulement à partir des donnes image. Nous avons montre quand et comment ceci est possible en utilisant la théorie des groupes de Lie. Nous avons aussi montré que l'ESM a des performance très satisfaisantes même quand les Jacobiens ne peuvent qu'être grossièrement estimés.

Plusieurs contribution ont été apportées à la conception de schémas d'asservissement visuel. Nous avons montré que les schémas d'asservissement visuel classiques de type "image-based" peuvent être améliorés en concevant une loi de commande basée sur l'ESM. L'analyse théorique de la stabilité et de la robustesse a permis de trouver des bornes sur la "taille" des erreurs de calibration que le système peut tolérer avant de devenir in stable.

Ensuite, nous avons conçu des méthodes d'asservissement visuel robustes qui ne nécessitent pas une mesure exacte des paramètres du système de vision et qui ne nécessitent pas d'aucune connaissance "a priori" sur la structure de la scène observée. Encore une fois, une attention particulière a été donnée a l'analyse théorique de la stabilité et de la robustesse des lois de commande par rapport aux erreurs d'estimation des paramètres du système.

Enfin, nous avons proposé des méthodes qui dépassent le cadre d'application des méthodes classique d'asservissement visuel. Nous avons montré, pour une caméra perspective, comment il est possible de définir une mesure qui est invariantes aux paramètres intrinsèques de la caméra. Ceci permet, entre autre, de définir des nouvelles taches d'asservissement visuel en utilisant des objectifs a focale variable. Une autre nouvelle approche concerne l'intégration des méthodes d'asservissement visuel dans les approche de localisation et de reconstruction. Ceci permet, de définir des nouvelles tâches d'asservissement

visuel sans mesurer au préalable une image de référence ou un modèle de la scène.

Perspectives de recherche

Dans mon travail de recherche futur, j'ai l'intention de développer un système de perception le plus générique possible et bien adapté aux applications robotiques (c'est-à-dire qui puisse tenir compte de la grande variabilité des environnements et de la grande diversité des taches robotiques). Il s'agit d'un objectif à (très) long terme, vu le nombre très diffèrent de fonctionnalités et de modalités de perception possibles. A plus court terme je vais donc considérer comme prioritaire la fonction de localisation, qui est une fonctionnalité primordiale en robotique. Dans ce cadre, je considère que la vision est une modalité sensorielle primordiale (mais pas l'unique) pour accéder à une certaine autonomie des systèmes robotisées interagissant avec des environnements dynamiques complexes. Toutefois, malgré les énormes progrès accomplis ces dernières années dans le domaine de la vision robotique la flexibilité, la robustesse et la fiabilité de la perception ne sont pas encore suffisantes pour des applications dans des environnements non contrôlés ou impliquant une interaction avec l'homme. Un système robuste et flexible basé sur la vision doit considérer des types différents de caméras (par exemple des caméras à projection centrale unique ou pas), doit être robuste à des très grand changements d'éclairage, doit considérer des objets rigides ou déformable, des scènes statiques ou dynamiques, etc. Ce système doit être assez générique afin d'être facilement couplé avec d'autres capteurs proprioceptifs et/ou extéroceptifs afin d'obtenir une localisation précise et robuste pour les applications robotiques. Je suis convaincu qu'une façon d'obtenir ces objectifs est d'étudier et développer davantage les méthodes directes. Deux problèmes scientifiques majeurs doivent êtres résolus.

Le premier problème concerne la modélisation de l'environnement et du système de perception. La difficulté majeure provient de la grande variabilité des types d'environnements, qui sont souvent des systèmes dynamiques ouverts et complexes. Les différents capteurs utilisés en robotique ne sont pas capables individuellement de donner une information robuste et fiable dans toutes les situations. Il est donc nécessaire d'utiliser des modalités sensorielles complémentaires. Par exemple, un système de perception multi capteur typique est composé par une ou plusieurs caméras, un télémètre laser et un système de positionnement absolu (par exemple un GPS). La modélisation de l'incertitude de chaque capteur est nécessaire si l'on veut fusionner de manière appropriée les différentes informations. Dans ce contexte, un problème scientifique important concerne la conception d'observateurs non-linéaire pour le filtrage des mesures. Il s'agit d'une direction de recherche pour moi nouvelle et récente que je commence à aborder par l'étude du filtrage de l'état d'un système non-linéaire défini sur un groupe de Lie. Une bonne modélisation de l'environnement est aussi primordiale. Par exemple, si l'on considère des capteurs optiques, il est nécessaire d'intégrer des modèles photométriques et géométriques génériques avec des modèles d'acquisition d'images appropriées. Les modèles photométriques doivent pouvoir tenir compte des possibles changements d'éclairage qui modifient non seulement l'intensité lumineuse mais aussi la couleur des objets. Les modèles géométriques doivent considérer non seulement des environnements rigides complexes (géométrie discontinues

ou fractale) mais aussi des environnements déformables. Tous ces modèles doivent reproduire assez fidèlement les phénomènes physiques et naturels mais sans être trop complexes afin d'être utilisés dans des applications robotiques temps-réel. En effet, la qualité d'un modèle dépend du problème pour lequel il est utilisé (simulation, mesure, commande, ...). Trouver un bon compromis entre complexité et qualité est un vrai défi scientifique car il n'existe pas une méthodologie solide pour nous guider dans ce choix.

Un deuxième problème concerne l'estimation des paramètres, associés aux modèles considérés, à partir des données acquises par des capteurs. Il s'agit d'un problème difficile car souvent de multiples acquisitions sont nécessaires afin de rendre le problème observable. Dans ce cas, il est nécessaire d'effectuer une association de données entre les différentes acquisitions. Une classe de méthodes qui a été largement étudiée en littérature simplifie le problème en séparant cette phase d'association de données de la solution du système. Une autre façon de résoudre le problème est d'effectuer l'association de données simultanément, en formalisant le problème d'estimation des paramètres comme un problème d'optimisation. Dans ce cas, il est nécessaire d'étudier des méthodes efficaces d'analyse numérique pour la solution de systèmes d'équations non-linéaires et pour l'optimisation. Les technique d'optimisation qui sont assez efficaces pour être utilisées dans des applications temps-réel robotiques ont généralement une convergence locale. Donc, une initialisation assez précise est nécessaire (c'est-à-dire une approximation initiale assez précise des paramètres inconnus). Une première direction de recherche est donc d'étudier comment initialiser de manière efficace l'optimisation par exemple en utilisant des algorithmes d'association des données très rapides mais fournissant seulement une approximation grossière de la solution, ou en utilisant des modalités sensorielles complémentaires. Toutefois, une solution plus élégante et générique serait d'étudier des méthodes efficaces d'optimisation globales. Un effort de recherche considérable sera nécessaire car ces méthodes sont généralement très coûteuses en temps de calcul et mal adaptées aux applications temps-réel.

Un autre objectif de mon travail de recherche futur est l'intégration des informations du système de perception dans des boucles de commande pour des systèmes robotiques complexes. La conception des lois de commande référencée vision se base sur une série d'hypothèses simplificatrices sur les propriétés géométriques et physiques de l'environnement, sur les caractéristiques des capteurs et sur celles des systèmes robotiques considérés. Mon objectif est d'augmenter la précision et la robustesse des lois de commande malgré les erreurs de modélisation induits par ces hypothèses. Tout d'abord, j'ai l'intention de concevoir des algorithmes de commande qui sont robustes par rapport aux erreurs de modélisation et au bruit de mesure et d'analyser leur propriétés de robustesse. Ensuite, j'étudierai le système dans sa globalité afin de voir si et comment il est possible d'assouplir les hypothèses généralement admises.

Concernant le système de vision il y a trois hypothèses qu'il faudrait lever.

La première hypothèse concerne le type de capteur utilisé. Actuellement presque uniquement des caméras à projection centrale unique sont utilisés pour la commande référencée vision. Je propose d'étudier l'utilisation de caméras à projection non-centrales. En effet, ce type de caméras peuvent être très utiles dans beaucoup d'applications car certaines ont un très grand champ de vue et une résolution constante. A ma connais-

sance l'utilisation de caméras à projection non centrales n'a pas encore été étudiée dans la commande référencée vision.

Une deuxième hypothèse concerne la rigidité de l'environnement. Je propose d'étudier si les méthodes de commande, qui ne nécessitent pas une reconstruction explicite de la structure, que j'ai proposés pour des objets rigides peuvent être appliqués aux objets déformables. Le problème clef est la détermination d'un diffeomorphisme entre l'information capteur et l'état Cartésien du robot. Il faut remarquer qu'il ne s'agit pas d'une question purement académique et que des applications très importantes existent en robotique médicale (par exemple la chirurgie non invasive à coeur battant).

La troisième hypothèse est que l'environnement est Lambertian. Cette hypothèse n'est bien sûr pas vérifiée en pratique. Il sera donc nécessaire d'introduire des modelés photométriques plus génériques mais aussi d'analyser théoriquement l'influence des erreurs de modélisation sur la stabilité des lois de commande.

Concernant les hypothèses sur les caractéristiques du robot, la plupart des lois de commande référencées vision sont conçues en supposant que les robots sont holonômes. Ces robots sont globalement linéarisables par retour d'état statique (permettant l'utilisation de techniques classiques de commande des systèmes linéaires). Pour des systèmes robotiques plus complexes (comme par exemple les robots non-holonômes) le modèle linéarisé autour de configurations fixes n'est pas commandable. Des nouvelles techniques de commande ont dû être élaborées pour ces systèmes. J'ai l'intention d'étudier si et comment les techniques de commande référencée vision que j'ai proposés pour les robots holonômes peuvent être généralisées ou adaptées à une large classe de systèmes robotiques complexes.

Bibliography

[Adelson 91] E. Adelson & J. Bergen. *The Plenoptic Function and the Elements of Early Vision*. Computational Models of Visual Processing, 1991. (page 7)

[Allen 93] P. K. Allen, A. Timcenko, Yoshimi B. & P Michelman. *Automated Tracking and Grasping of a Moving Object with a Robotic Hand-Eye System*. IEEE Transactions on Robotics and Automation, vol. 9, no. 2, pages 152–165, April 1993. (page 71)

[Ansar 03] A. Ansar & K. Daniilidis. *Linear Pose Estimation from Points or Lines*. IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 25, no. 5, pages 578–589, May 2003. (page 49)

[Baker 99] S. Baker & K. Nayar. *A theory of single-viewpoint catadioptric image formation*. International Journal of Computer Vision, vol. 35, no. 2, pages 1–22, 1999. (page 9)

[Baker 01] S. Baker & I. Matthews. *Equivalence and Efficiency of Image Alignment Algorithms*. In IEEE Conference on Computer Vision and Pattern Recognition, volume 1, pages 1090–1097, Los Alamitos, CA, USA, December 2001. (page 54)

[Baker 04] S. Baker & I. Matthews. *Lucas-Kanade 20 years on : a unifying framework*. International Journal of Computer Vision, vol. 56, no. 3, pages 221–255, February 2004. (page 25)

[Barreto 02] J.P. Barreto & H. Araujo. *Geometric Properties of Central Catadioptric Line Images*. In European Conf. on Computer Vision, pages 237–251, 2002. (page 9)

[Bartoli 04] A. Bartoli & A. Zisserman. *Direct estimation of non-rigid registrations*. In British machine vision conference, volume 2, pages 899–908, 2004. (page 54)

[Basri 99] R. Basri, E. Rivlin & I. Shimshoni. *Visual homing : surfing on the epipole*. International Journal of Computer Vision, vol. 33, no. 2, pages 22–39, 1999. (page 76)

[Beaton 74] A. E. Beaton & J. W. Tukey. *The fitting of power series, meaning polynomials, illustrated on band-spectroscopic data*. Technometrics, vol. , no. 16, pages 147–185, 1974. (pages 41, 42)

[Benhimane 03] S. Benhimane & E. Malis. *Vision-based control with respect to planar and non-planar objects using a zooming camera*. In IEEE International

Conference on Advanced Robotics, pages 991–996, Coimbra, Portugal, July 2003. (pages 87, 99)

[Benhimane 04] S. Benhimane & E. Malis. *Real-time image-based tracking of planes using efficient second-order minimization.* In IEEE/RSJ International Conference on Intelligent Robots Systems, volume 1, pages 943–948, Sendai, Japan, September-October 2004. (pages 25, 54, 56, 81, 101)

[Benhimane 05] S. Benhimane, E. Malis, P. Rives & J. R. Azinheira. *Vision-based Control for Car Platooning using Homography Decomposition.* In IEEE International Conference on Robotics and Automation, pages 2173–2178, Barcelona, Spain, April 2005. (pages 78, 102)

[Benhimane 06a] S. Benhimane & E. Malis. *Homography-based 2D Visual Servoing.* In IEEE International Conference on Robotics and Automation, pages 2397–2402, Orlando, USA, May 2006. (pages 80, 81, 103)

[Benhimane 06b] S. Benhimane & E. Malis. *Integration of Euclidean constraints in template-based visual tracking of piecewise-planar scenes.* In IEEE/RSJ International Conference on Intelligent Robots Systems, pages 1218–1223, Beijing, China, October 2006. (pages 64, 103)

[Benhimane 06c] S. Benhimane & E. Malis. *A new approach to vision-based robot control with omni-directional cameras.* In IEEE International Conference on Robotics and Automation, pages 526–531, Orlando, USA, May 2006. (page 103)

[Benhimane 07] S. Benhimane & E. Malis. *Homography-based 2D Visual Tracking and Servoing.* International Journal of Robotic Research, vol. 26, no. 7, pages 661–676, July 2007. (page 103)

[Benhimane 08] S. Benhimane, H. Najafi, M. Grundmann, Y. Genc, N. Navab & E. Malis. *Real-time object detection and tracking for industrial applications.* In International Conference on Computer Vision Theory and Applications, Funchal, Portugal, January 2008. (page 52)

[Bergen 92] J. Bergen, P. Anandan, K. Hanna & R. Hingorani. *Hierarchical Model-Based Motion Estimation.* In European Conference on Computer Vision, pages 237–252, 1992. (page 54)

[Black 98] M. Black & A. Jepson. *Eigen-tracking : robust matching and tracking of articulated objects using a view-based representation.* International Journal of Computer Vision, vol. 26, no. 1, pages 63–84, 1998. (page 53)

[Blinn 77] J. F. Blinn. *Models of Light Reflection for Computer Synthesized Pictures.* In SIGGRAPH, pages 192–198, 1977. (page 8)

[Boufama 95] B. Boufama & R. Mohr. *Epipole and fundamental matrix estimation using the virtual parallax property.* In IEEE International Conference on Computer Vision, pages 1030–1036, Cambridge, USA, 1995. (page 47)

[Brandou 06] V. Brandou, E. Malis, P. Rives, Allais A.-G. & M. Perrier. *Active stereovision using invariant visual servoing.* In IEEE/RSJ International Conference on Intelligent Robots Systems, pages 2326–2331, Beijing, China, October 2006. (pages 88, 101)

[Brandou 07] V. Brandou, A. Allais, M. Perrier, E. Malis, P. Rives, J. Sarrazin & P. Sarradin. *3D reconstruction of natural underwater scenes using the stereovision system IRIS.* In OCEANS, Aberdeen, Scotland, June 2007. (pages 88, 101)

[Brown 92] Lisa Gottesfeld Brown. *A Survey of Image Registration Techniques.* ACM Computing Surveys, vol. 24, no. 4, pages 325–376, 1992. (page 53)

[Canny 86] J. Canny. *A Computational Approach to Edge Detection.* IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 8, no. 6, pages 679–698, November 1986. (page 44)

[Carr 97] J. Carr, W. Fright & R. Beatson. *Surface interpolation with radial basis functions for medical imaging.* IEEE Transactions Med. Imag., vol. 16, no. 1, February 1997. (pages 55, 59)

[Cervera 03] E. Cervera, F. Berry, P. Martinet & A. Del Pobil. *Improving Image-Based Visual Servoing with 3D Features.* International Journal of Robotics Research, 2003. (page 74)

[Chaumette 96] F. Chaumette, S. Boukir, P. Boutemy & D. Juvin. *Structure from controlled motion.* IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 18, no. 5, pages 492–504, March 1996. (page 49)

[Chaumette 98] F. Chaumette. *Potential problems of stability and convergence in image-based and position-based visual servoing.* In D. Kriegman, G. Hager & A. Morse, editeurs, The confluence of vision and control, volume 237 of *LNCIS Series*, pages 66–78. Springer Verlag, 1998. (page 74)

[Chaumette 06] F. Chaumette & S. Hutchinson. *Visual Servo Control, Part I : Basic Approaches.* IEEE Robotics and Automation Magazine, vol. 13, no. 4, pages 82–90, December 2006. (pages 14, 71)

[Chaumette 07] F. Chaumette & S. Hutchinson. *Visual Servo Control, Part II : Advanced Approaches.* IEEE Robotics and Automation Magazine, vol. 14, no. 1, pages 109–118, March 2007. (pages 14, 71)

[Cheah 98] C. C. Cheah, S. Kawamura & S. Arimoto. *Feedback control for Robotic Manipulator with uncertain kinematics and dynamics.* In IEEE Int. Conf. on Robotics and Automation, volume 4, pages 3607–3612, Leuven, Belgium, May 1998. (page 75)

[Chesi 99] G. Chesi, E. Malis & R. Cipolla. *Collineation estimation from two unmatched views of an unknown planar contour for visual servoing.* In British Machine Vision Conference, volume 1, pages 224–233, Nottingham, United Kingdom, September 1999. (pages 77, 97)

[Chesi 00] G. Chesi, E. Malis & R. Cipolla. *Automatic segmentation and matching of planar contours for visual servoing.* In IEEE International Conference on Robotics and Automation, volume 3, pages 2753–2758, S. Francisco, CA, United States, April 2000. (pages 77, 97)

[Comport 07] A. Comport, E. Malis & P. Rives. *Accurate Quadrifocal Tracking for Robust 3D Visual Odometry.* In IEEE International Conference on Robotics and Automation, Rome, Italy, April 2007. (pages 56, 65, 66, 68, 103)

[Cook 82] R.L. Cook & K.E. Torrance. *A Reflectance Model for Computer Graphics*. ACM Trans. on Graphics 1, pages 7–24, 1982. (page 8)

[Cootes 98] T. F. Cootes, G. J. Edwards & C. J. Taylor. *Active appearence models*. In European Conference on Computer Vision, volume 2, pages 484–498, 1998. (page 53)

[Courbon 07] J. Courbon, Y. Mezouar, L. Eckt & P. Martinet. *A generic fisheye camera model for robotic applications*. In IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 1683–1688, San Diego, USA, 2007. (page 9)

[Davison 03] A.J. Davison. *Real-Time Simultaneous Localisation and Mapping with a Single Camera*. In International Conference on Computer Vision, October 2003. (page 47)

[De Luca 07a] A. De Luca, G. Oriolo & P. Giordano. *Image-based visual servoing schemes for nonholonomic mobile manipulators*. Robotica, vol. 25, no. 2, pages 131–145, March 2007. (page 71)

[De Luca 07b] A. De Luca, G. Oriolo & P. Giordano. *On-Line Estimation of Feature Depth for Image-Based Visual Servoing Schemes*. In IEEE International Conference on Robotics and Automation, pages 2823–2828, April 2007. (page 49)

[Dementhon 95] D Dementhon & L. S. Davis. *Model-Based Object Pose in 25 Lines of Code*. International Journal of Computer Vision, vol. 15, no. 1/2, pages 123–141, June 1995. (page 49)

[Deng 02] Lingfeng Deng, F. Janabi-Sharifi & W.J. Wilson. *Stability and robustness of visual servoing methods*. In icra, volume 2, pages 1604–1609, Washington, DC, USA, May 2002. (page 75)

[Dennis 83] J.E. Dennis & Schnabel R. B. Numerical methods for unconstrained optimization and nonlinear equations. Classics in applied mathematics. SIAM, 1983. (pages 17, 36)

[Deriche 90] R. Deriche & G. Giraudon. *Accurate corner detection : an analytical study*. In International Conference on Computer Vision, pages 66–70, 1990. (page 44)

[Dhome 89] M. Dhome, M. Richetin & J. Lapreste. *Determination of the Attitude of 3D Objects from a Single Perspective View*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, no. 12, pages 1265–1278, December 1989. (page 49)

[Drummond 99] T. Drummond & R. Cipolla. *Visual Tracking and Control Using Lie Algebras*. In IEEE International Conference on Computer Vision and Pattern Recognition, volume 2, pages 652–657, Fort Collins, Colorado, USA, June 1999. (page 77)

[Duda 72] R. O. Duda & P. E. Hart. *Use of the Hough Transformation to Detect Lines and Curves in Pictures*. Comm. ACM, vol. 15, pages 11–15, January 1972. (page 46)

[Espiau 92] B. Espiau, F. Chaumette & P. Rives. *A New Approach to Visual Servoing in Robotics*. IEEE Transactions on Robotics and Automation, vol. 8, no. 3, pages 313–326, June 1992. (pages 25, 73, 74)

[Espiau 93] B. Espiau. *Effect of Camera Calibration Errors on Visual Servoing in Robotics*. In 3rd International Symposium on Experimental Robotics, Kyoto, Japan, October 1993. (page 75)

[Espiau 02] F.-X. Espiau, E. Malis & P. Rives. *Robust features tracking for robotic applications : towards 2 1/2 D visual servoing with natural images*. In IEEE International Conference on Robotics and Automation, volume 1, pages 574–579, Washington, D.C., USA, May 2002. (pages 48, 102)

[Fabrizio 02] J. Fabrizio, J.-P. Tarel & R. Benosman. *Calibration of Panoramic Catadioptric Sensors Made Easier*. In IEEE Workshop on Omnidirectional Vision (Omnivis'02), 2002. (page 46)

[Fang 05] Y. Fang, W. Dixon, D. Dawson & P. Chawda. *Homography-Based Visual Servoing of Wheeled Mobile Robots*. IEEE Transactions on Systems, Man, and Cybernetics - Part B, vol. 35, no. 5, pages 1041–1050, 2005. (page 71)

[Faugeras 87] O.D. Faugeras & G. Toscani. *Camera calibration for 3D computer vision*. In International Workshop on Machine Vision and Machine Intelligence, pages 240–247, February 1987. (page 46)

[Faugeras 88] O. Faugeras & F. Lustman. *Motion and Structure From Motion in a Piecewise Planar Environment*. International Journal of Pattern Recognition and Artificial Intelligence, vol. 2, no. 3, pages 485–508, 1988. (page 48)

[Faugeras 93] O. Faugeras. Three-dimensionnal computer vision : a geometric viewpoint. MIT Press, Cambridge, MA, 1993. (pages 11, 43, 47, 50)

[Faugeras 01] O. Faugeras, Q. Luong & T. Papadopoulo. The geometry of multiple images : The laws that govern the formation of multiple images of a scene and some of their applications. MIT Press, 2001. (pages 11, 43, 47, 47)

[Fiore 01] P. Fiore. *Efficient linear solution of exterior orientation*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 2, pages 140–148, February 2001. (page 49)

[Fischler 81] M. A. Fischler & R. C. Bolles. *Random Sample Consensus : A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography*. Comm. of the ACM, vol. 24, pages 381–395, 1981. (page 46)

[Fleet 00] David J. Fleet, Michael J. Black, Yaser Yacoob & Allan D. Jepson. *Design and Use of Linear Models for Image Motion Analysis*. Int. J. Comput. Vision, vol. 36, no. 3, pages 171–193, 2000. (page 53)

[Fruchard 06] M. Fruchard, M. Morin & C. Samson. *A Framework for the Control of Nonholonomic Mobile Manipulators*. International Journal of Robotics Research, vol. 25, no. 8, pages 745–780, August 2006. (page 71)

[Garcia-Aracil 05a] N. Garcia-Aracil, E. Malis, R. Aracil-Santonja & C. Perez-Vidal. *Continuous Visual Servoing Despite the Changes of Visibility in Image Features*. IEEE Transaction on Robotics, vol. 21, no. 6, pages 1214–1220, December 2005. (pages 87, 100)

[Garcia-Aracil 05b] N. Garcia-Aracil, O. Reinoso, E. Malis & R. Aracil. *Parameters Selection and Stability Analysis of Invariant Visual Servoing with Weighted Features*. In IEEE International Conference on Robotics and Automation, pages 3492–3497, Barcelona, Spain, April 2005. (pages 87, 100)

[Garcia 04a] N. Garcia & E. Malis. *Preserving the continuity of visual servoing despite changing image features*. In IEEE/RSJ International Conference on Intelligent Robots Systems, volume 2, pages 1383–1388, Sendai, Japan, September-October 2004. (pages 87, 100)

[Garcia 04b] N. Garcia, E. Malis, O. Reinoso & R. Aracil. *Visual Servoing Techniques for Continuous Navigation of a Mobile Robot*. In International Conference on Informatics in Control, Automation and Robotics, Setubal, Portugal, October 2004. (pages 87, 100)

[Geyer 99] C. Geyer & K. Daniilidis. *Catadioptric Camera Calibration*. In International Conference on Computer Vision, volume 1, pages 398–405, 1999. (page 46)

[Geyer 00] C. Geyer & K. Daniilidis. *A Unifying Theory for Central Panoramic Systems and Practical Applications*. In European Conf. on Computer Vision, volume 2, pages 445–461, 2000. (page 9)

[Gleicher 97] M. Gleicher. *Projective registration with difference decomposition*. In IEEE International Conference on Computer Vision and Pattern Recognition, pages 331–337, 1997. (page 53)

[Hager 95] G. Hager, W. Chang & A. Morse. *Robot hand-eye coordination based on stereo vision : Towards Calibration-Free Hand-Eye Coordination*. IEEE Control Systems Magazine, vol. 15, no. 1, pages 30–39, February 1995. (page 71)

[Hager 98] G. D. Hager & P. N. Belhumeur. *Efficient region tracking with parametric models of geometry and illumination*. IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 20, no. 10, pages 1025–1039, 1998. (page 54)

[Hall 03] B. Hall. Lie groups, lie algebras, and representations : an elementary introduction. Springer, 2003. (pages 4, 5, 6, 32)

[Hamel 04] T. Hamel & R. Mahony. *Pure 2D visual servo control for a class of under-actuated dynamic systems*. In IEEE International Conference on Robotics and Automation, volume 3, pages 2229– 2235, April 2004. (page 71)

[Harris 88] C. Harris & M. Stephens. *A combined corner and Edge Detector*. In Proceedings of the 4th Alvey Vision Conference, pages 147–151, 1988. (page 44)

[Hartley 97] R. Hartley. *In Defense of the Eight-Point Algorithm*. IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 19, no. 6, pages 580–593, June 1997. (page 47)

[Hartley 00] R. I. Hartley & A. Zisserman. Multiple view geometry in computer vision. Cambridge University Press, ISBN : 0521623049, 2000. (pages 11, 43, 47, 50)

[Hashimoto 93] Hashimoto, editeur. Visual servoing : Real time control of robot manipulators based on visual sensory feedback, volume 7 of *World Scientific Series in Robotics and Automated Systems*. World Scientific Press, Singapore, 1993. (pages 14, 71, 73)

[Horaud 97] R. Horaud, F. Dornaika, B. Lamiroy & S. Christy. *Object Pose : The Link between Weak Perspective, Paraperspective, and Full Perspective*. International Journal of Computer Vision, vol. 22, no. 2, pages 173–189, 1997. (page 49)

[Horaud 98] R. Horaud, F. Dornaika & B. Espiau. *Visually guided object grasping*. IEEE Transaction on Robotics and Automation, vol. 14, no. 4, pages 525–532, August 1998. (page 71)

[Horn 81] B Horn & B. Schunk. *Determining Optical Flow*. Artificial Intelligence, vol. 17, pages 185–203, 1981. (pages 43, 53)

[Horn 88] B. Horn & E. Weldon. *Direct methods for recovering motion*. International Journal of Computer Vision, vol. 2, pages 51–76, 1988. (pages 43, 53)

[Huber 81] P. J. Huber. Robust statistics. Wiley, New York, 1981. (pages 41, 42, 46)

[Hutchinson 96] S. Hutchinson, G. D. Hager & P. I. Corke. *A tutorial on Visual Servo Control*. IEEE Transactions on Robotics and Automation, vol. 12, no. 5, pages 651–670, October 1996. (pages 14, 71)

[Irani 00] M. Irani & P. Anandan. *All about direct methods*. In B. Triggs, A. Zisserman & R. Szeliski, editeurs, Vision Algorithms : Theory and practice, numéro 1883 in Lecture Notes in Computer Science, pages 267–277. Springer-Verlag, September 2000. (pages 43, 52)

[Irani 02] M. Irani & P. Anandan. *Direct recovery of planar-parallax from multiple frames*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 11, pages 1528–1534, November 2002. (page 54)

[Isaacson 66] E. Isaacson & H. Keller. Analysis of numerical methods. John Wiley & Sons, 1966. (pages 17, 18, 19, 29)

[Jin 03] H. Jin, P. Favaro & S. Soatto. *A semi-direct approach to structure from motion*. The Visual Computer, vol. 19, no. 6, pages 377–394, 2003. (page 65)

[Kumar 94a] R. Kumar, P. Anandan & K. Hanna. *Direct Recovery Of Shape From Multiple Views : A Parallax Based Approach*. In ICPR, pages 685–688, 1994. (page 54)

[Kumar 94b] R. Kumar & A. Hanson. *Robust Methods for Estimating Pose and a Sensitivity Analysis*. Computer Vision, Graphics, and Image Processing : Image Understanding, vol. 60, no. 3, pages 313–342, 1994. (page 49)

[La Cascia 00] M. La Cascia, S. Sclaroff & V. Athitsos. *Fast, reliable head tracking under varying illumination : an approach based on registration of textured-mapped 3D models*. IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 22, no. 4, pages 322–336, 2000. (page 53)

[Longuet-Higgins 81] H. C. Longuet-Higgins. *A computer algorithm for reconstructing a scene from two projections*. Nature, vol. 293, pages 133–135, September 1981. (page 47)

[Longuet-Higgins 84] H.C. Longuet-Higgins. *The reconstruction of a scene from two projections : configurations that defeat the 8-point algorithm*. In Conference on Artificial intelligence applications, pages 395–397, Denver, 1984. (page 47)

[Lowe 91] D.G. Lowe. *Fitting Parameterized Three-Dimensional Models to Images*. IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 13, no. 5, pages 441–450, May 1991. (page 49)

[Lowe 04] David G. Lowe. *Distinctive image features from scale-invariant keypoints*. International Journal of Computer Vision, vol. 60, no. 2, pages 91–110, 2004. (pages 44, 45)

[Lu 00] C. Lu, G. Hager & E. Mjolsness. *Fast and Globally Convergent Pose Estimation from Video Images*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 6, pages 610–622, 2000. (page 49)

[Lucas 81] B. Lucas & T. Kanade. *An iterative image registration technique with application to stereo vision*. In International Joint Conference on Artificial Intelligence, pages 674–679, 1981. (page 54)

[Ma 03] Y. Ma, S. Soatto, J. Kosecka & S. Sastry. An invitation to 3-d vision : From images to geometric models. Springer-Verlag, 2003. (pages 11, 43, 47, 47)

[Mahony 01] R. Mahony & T. Hamel. *Visual servoing using linear features for underactuated rigid bodydynamics*. In IEEE/RSJ International Conference on Intelligent Robots and Systems, volume 2, pages 1153–1158, 2001. (page 71)

[Mahony 02] R. Mahony & J. Manton. *The Geometry of the Newton Method on Non-Compact Lie Groups*. Journal of Global Optimization, vol. 23, no. 3-4, pages 309–327, 2002. (page 32)

[Malis 99] E. Malis, F. Chaumette & S. Boudet. *2 1/2 D Visual Servoing*. IEEE Transactions on Robotics and Automation, vol. 15, no. 2, pages 234–246, April 1999. (pages 48, 49, 76, 77)

[Malis 00a] E. Malis, F. Chaumette & S. Boudet. *2 1/2 D Visual Servoing with Respect to Unknown Objects Through a New Estimation Scheme of Camera Displacement*. Intenational Journal of Computer Vision, vol. 37, no. 1, pages 79–97, June 2000. (page 48)

[Malis 00b] E. Malis & R. Cipolla. *Multi-view constraints between collineations : application to self-calibration from unknown planar structures*. In European Conference on Computer Vision, volume 2, pages 610–624, Dublin, EIRE, June 2000. (pages 51, 97)

[Malis 00c] E. Malis & R. Cipolla. *Self-calibration of zooming cameras observing an unknown planar structure.* In Int. Conf. on Pattern Recognition, volume 1, pages 85–88, Barcelona, Spain, September 2000. (pages 51, 97)

[Malis 01a] E. Malis. *Hybrid vision-based robot control robust to large calibration errors on both intrinsic and extrinsic camera parameters.* In European Control Conference, pages 2898–2903, Porto, Portugal, September 2001. (pages 48, 77, 96)

[Malis 01b] E. Malis. *Vision-based control using different cameras for learning the reference image and for servoing.* In IEEE/RSJ International Conference on Intelligent Robots Systems, volume 3, pages 1428–1433, Maui, Hawaii, November 2001. (pages 87, 98)

[Malis 01c] E. Malis. *Visual servoing invariant to changes in camera intrinsic parameters.* In International Conference on Computer Vision, volume 1, pages 704–709, Vancouver, Canada, July 2001. (pages 86, 98)

[Malis 02a] E. Malis. *Stability Analysis of Invariant Visual Servoing and Robustness to Parametric Uncertainties.* In Joint CSS/RAS International Workshop on Control Problems in Robotics and Automation, Las Vegas, Nevada, USA, December 2002. (pages 75, 99)

[Malis 02b] E. Malis. *Survey of vision-based robot control.* In European Naval Ship Design, Captain Computer IV Forum, Brest, France, 2002. (pages 14, 71)

[Malis 02c] E. Malis. *A Unified Approach to Model-based and Model-free Visual Servoing.* In European Conference on Computer Vision, volume 4, pages 433–447, Copenhagen, Denmark, May 2002. (pages 87, 99)

[Malis 02d] E. Malis. *Vision-based control invariant to camera intrinsic parameters : stability analysis and path tracking.* In IEEE International Conference on Robotics and Automation, volume 1, pages 217–222, Washington, D.C., USA, May 2002. (pages 87, 98)

[Malis 02e] E. Malis, J.-J. Borelly & P. Rives. *Intrinsics-free visual servoing with respect to straight lines.* In IEEE/RSJ International Conference on Intelligent Robots Systems, volume 1, pages 384–389, Lausanne, Switzerland, October 2002. (pages 86, 87, 98)

[Malis 02f] E. Malis & F. Chaumette. *Theoretical improvements in the stability analysis of a new class of model-free visual servoing methods.* IEEE Transaction on Robotics and Automation, vol. 18, no. 2, pages 176–186, April 2002. (pages 48, 76, 77, 96)

[Malis 02g] E. Malis & R. Cipolla. *Camera self-calibration from unknown planar structures enforcing the multi-view constraints between collineations.* IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 24, no. 9, pages 1268–1272, September 2002. (pages 51, 97)

[Malis 03a] E. Malis. Control problems in robotics, volume 4 of *Springer Tracts in Advanced Robotics*, chapitre Stability Analysis of Invariant Visual Servoing and Robustness to Parametric Uncertainties, pages 265–277. Springer, 2003. (pages 75, 99)

[Malis 03b] E. Malis, G. Chesi & R. Cipolla. *2 1/2 D visual servoing with respect to planar contours having complex and unknown shapes.* International Journal of Robotic Research, vol. 22, no. 10-11, pages 841–853, October-November 2003. Special Issue on Visual Servoing. (pages 97, 102)

[Malis 03c] E. Malis & P. Rives. *Robustness of Image-Based Visual Servoing with Respect to Depth Distribution Errors.* In IEEE International Conference on Robotics and Automation, volume 1, pages 1056–1061, Taipei, Taiwan, October 2003. (pages 75, 99)

[Malis 03d] E. Malis & P. Rives. *Uncalibrated Active Affine Reconstruction closing the loop by Visual Servoing.* In IEEE/RSJ International Conference on Intelligent Robots Systems, volume 1, pages 498–503, Las Vegas, USA, October 2003. (pages 49, 100)

[Malis 04a] E. Malis. *Improving vision-based control using efficient second-order minimization techniques.* In IEEE International Conference on Robotics and Automation, volume 2, pages 1843–1848, New Orleans, USA, April 2004. (pages 25, 72, 74, 101)

[Malis 04b] E. Malis. *Visual servoing invariant to changes in camera intrinsic parameters.* IEEE Transaction on Robotics and Automation, vol. 20, no. 1, pages 72–81, February 2004. (pages 86, 87, 98)

[Malis 04c] E. Malis & A. Bartoli. *Euclidean Reconstruction Independent on Camera Intrinsic Parameters.* In IEEE/RSJ International Conference on Intelligent Robots Systems, volume 3, pages 2313–2318, Sendai, Japan, September-October 2004. (page 88)

[Malis 04d] E. Malis & Benhimane. *A Unified Framework for Real-time Visual Tracking and Servoing.* In Workshop on Advances in Robot Vision, IEEE/RSJ International Conference on Intelligent Robots Systems, Sendai, Japan, September-October 2004. (pages 77, 102)

[Malis 05] E. Malis & S. Benhimane. *A unified approach to visual tracking and servoing.* Robotics and Autonomous Systems, vol. 52, no. 1, pages 39–52, July 2005. (pages 77, 102)

[Malis 06] E. Malis & E. Marchand. *Experiments with robust techniques in real-time robotic vision.* In IEEE/RSJ International Conference on Intelligent Robots Systems, pages 223–228, Beijing, China, October 2006. (pages 41, 102)

[Malis 07a] E. Malis. *An efficient unified approach to direct visual tracking of rigid and deformable surfaces.* In IEEE/RSJ International Conference on Intelligent Robots Systems, San Diego, USA, October 2007. (pages 54, 55, 60, 81)

[Malis 07b] E. Malis & M. Vargas. *Deeper understanding of the homography decomposition for vision-based control.* Research Report 6303, INRIA, September 2007. (pages 48, 80)

[Martinet 97] P. Martinet, N. Daucher, J. Gallice & M. Dhome. *Robot control using monocular pose estimation.* In Workshop on New Trends In Image-Based Robot Servoing, pages 1–12, Grenoble, France, September 1997. (page 89)

[Matthies 89] L. Matthies, T. Kanade & R. Szeliski. *Kalman Filter-based Algorithms for Estimating Depth from Image Sequences.* International Journal of Computer Vision, no. 3, pages 209–236, 1989. (page 49)

[Maya 06] M. Maya, P. Morin & C. Samson. *Control of a Nonholonomic Mobile Robot Via Sensor-based Target Tracking and Pose Estimation.* In IEEE International Conference on Intelligent Robots and Systems, pages 5612–5618, 2006. (page 71)

[Mei 06a] C. Mei, S. Benhimane, E. Malis & P. Rives. *Homography-based Tracking for Central Catadioptric Cameras.* In IEEE/RSJ International Conference on Intelligent Robots Systems, pages 669–674, Beijing, China, October 2006. (pages 64, 67, 103)

[Mei 06b] C. Mei & E. Malis. *Fast central catadioptric line extraction, estimation, tracking and structure from motion.* In IEEE/RSJ International Conference on Intelligent Robots Systems, pages 4774–4779, Beijing, China, October 2006. (page 102)

[Mei 06c] C. Mei, E. Malis, S. Benhimane & P. Rives. *Constrained Multiple Planar Template Tracking for Central Catadioptric Cameras.* In British Machine Vision Conference, Edinburgh, United Kingdom, September 2006. (pages 64, 67, 103)

[Mei 07] C. Mei & P. Rives. *Single View Point Omnidirectional Camera Calibration from Planar Grids.* In IEEE International Conference on Robotics and Automation, April 2007. (page 46)

[Mendonça 99] P. Mendonça & R. Cipolla. *A Simple Techinique for Self-Calibration.* In IEEE International Conference on Computer Vision and Pattern Recognition, volume 1, pages 500–505, 1999. (page 50)

[Mezouar 04] Y. Mezouar & E. Malis. *Robustness of Central Catadioptric Image-Based Visual Servoing to Uncertainties on 3D Parameters.* In IEEE/RSJ International Conference on Intelligent Robots Systems, volume 2, pages 1389–1394, Sendai, Japan, September-October 2004. (pages 75, 75, 99)

[Mikolajczyk 04] K. Mikolajczyk & C. Schmid. *Scale and affine invariant interest point detectors.* International Journal of Computer Vision, vol. 60, no. 1, pages 63–86, 2004. (pages 44, 45)

[Molton 04] N. D. Molton, A. J. Davison & I. D. Reid. *Locally Planar Patch Features for Real-Time Structure from Motion.* In Proc. BMVC, 2004. (page 65)

[Moons 96] T. Moons, L. Van Gool, M. Proesmans & E. Pauwels. *Affine reconstruction from perspective image pairs with a relative object-camera translation in between.* IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 18, no. 1, pages 77–83, January 1996. (page 49)

[Moreno 07] F. Moreno, V. Lepetit & P. Fua. *Accurate Non-Iterative O(n) Solution to the PnP Problem.* In International Conference on Computer Vision, pages 1–8, October 2007. (page 49)

[Morin 04] P. Morin. Stabilisation de systèmes non linéaires critiques et application à la commande de véhicules. Habilitation à Diriger des Recherches. Université de Nice-Sophia Antipolis, 2004. (page 7)

[Nakamura 86] Y. Nakamura & H. Hanafusa. *Inverse kinematics solutions with singularity robustness for robot manipulator control.* Transaction ASME Journal of Dynamic Systems, Measurement and Control, vol. 108, pages 163–171, September 1986. (page 73)

[Nister 04] D. Nister. *An efficient solution to the five-point relative pose problem.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 6, pages 756–770, 2004. (page 47)

[Nister 06] D. Nister, O. Naroditsky & J. Bergen. *Visual Odometry for Ground Vehicle Applications.* Journal of Field Robotics, vol. 23, no. 1, 2006. (page 47)

[Pennec 99] X. Pennec, P. Cachier & N. Ayache. *Understanding the "Demon's Algorithm" : 3D Non-Rigid registration by Gradient Descent.* In Int. Conf. on Medical Image Computing and Computer-Assisted Intervention (MICCAI'99), volume 1679 of *LNCS*, pages 597–605, Cambridge, UK, September 1999. Springer Verlag. (page 54)

[Pollefeys 99] M. Pollefeys, R. Koch & L. VanGool. *Self-Calibration and Metric Reconstruction Inspite of Varying and Unknown Intrinsic Camera Parameters.* ijcv, vol. 32, no. 1, pages 7–25, August 1999. (page 50)

[Quarteroni 00] A. Quarteroni, R. Sacco & F. Salieri. Numerical mathematics. Springer-Verlag, 2000. (pages 17, 22)

[Rogel 06] P. Rogel & S. Kova. *Symmetric image registration.* Medical Image Analysis, vol. 10, no. 3, pages 484–493, 2006. (page 54)

[Royer 07] E. Royer, M. Lhuillier, M. Dhome & J. Lavest. *Monocular vision for mobile robot localization and autonomous navigation.* International Journal of Computer Vision, vol. 74, no. 3, pages 237–260, 2007. (page 47)

[Se 05] S. Se, D. Lowe & J. Little. *Vision-based global localization and Mapping for Mobile Robots.* IEEE Transactions on Robotics and Automation, vol. 21, no. 3, pages 364–375, 2005. (page 47)

[Silveira 06a] G. Silveira, E. Malis & P. Rives. *Real-time Robust Detection of Planar Regions in a Pair of Images.* In IEEE/RSJ International Conference on Intelligent Robots Systems, pages 49–54, Beijing, China, October 2006. (pages 68, 89, 104)

[Silveira 06b] G. Silveira, E. Malis & P. Rives. *Visual Servoing over Unknown, Unstructured, Large-scale Scenes.* In IEEE International Conference on Robotics and Automation, pages 4142–4147, Orlando, USA, May 2006. (pages 89, 90, 104)

[Silveira 07a] G. Silveira & E. Malis. *Direct Visual Servoing with respect to Rigid Objects.* In IEEE Computer Vision and Pattern Recognition, San Diego, USA, October 2007. (pages 80, 81)

[Silveira 07b] G. Silveira & E. Malis. *Real-time Visual Tracking under Arbitrary Illumination Changes.* In IEEE Computer Vision and Pattern Recognition, Minneapolis, USA, June 2007. (pages 54, 59, 60, 103)

[Silveira 07c] G. Silveira, E. Malis & P. Rives. *An Efficient Direct Method for Improving visual SLAM.* In IEEE International Conference on Robotics and Automation, Rome, Italy, April 2007. (pages 65, 68, 89, 104)

[Smith 94] C. Smith & N. Papanikolopoulos. *Computation of shape through controlled active exploration.* In IEEE International Conference on Robotics and Automation, volume 3, pages 2516–2521, San Diego, CA, USA, May 1994. (page 49)

[Stein 00] G. Stein & A. Shashua. *Model-Based Brightness Constraints : On Direct Estimation of Structure and Motion.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 9, pages 992–1015, September 2000. (page 43)

[Szeliski 95] R. Szeliski & S. Kang. *Direct Method for Visual Scene Reconstruction.* In IEEE Workshop on Representation of Visual Scenes, 1995. (page 54)

[Thirion 98] J. Thirion. *Image matching as a diffusion process : an analogy with Maxwell's demons.* Medical Image Analysis, vol. 2, no. 3, pages 243–260, 1998. (page 54)

[Torr 98] P. Torr, A. W. Fitzgibbon & A. Zisserman. *Maintaining Multiple Motion Model Hypotheses Over Many Views to Recover Matching and Structure.* In IEEE International Conference on Computer Vision, pages 485–491, Bombay, India, January 1998. (page 47)

[Torr 00] P. H. S. Torr & A Zisserman. *Feature Based Methods for Structure and Motion Estimation.* In B. Triggs, A. Zisserman & R. Szeliski, editeurs, Vision Algorithms : Theory and practice, numéro 1883 in Lecture Notes in Computer Science, pages 279–295. Springer-Verlag, September 2000. (page 43)

[Triggs 97] B. Triggs. *Autocalibration and the Absolute Quadric.* In IEEE International Conference on Computer Vision and Pattern Recognition, pages 609–614, 1997. (page 50)

[Triggs 98] B. Triggs. *Autocalibration from planar scenes.* In European Conference of Computer Vision, pages 89–105, 1998. (page 51)

[Triggs 00] B. Triggs, P. McLauchlan, R. Hartley & A. Fitzgibbon. *Bundle Adjustment – A Modern Synthesis.* In B. Triggs, A. Zisserman & R. Szeliski, editeurs, Vision Algorithms : Theory and Practice, volume 1883 of *Lecture Notes in Computer Science*, pages 298–372. Springer-Verlag, 2000. (pages 47, 50)

[Tsai 87] R.Y. Tsai. *A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses.* IEEE Journal of Robotics and Automation, vol. 3, no. 4, pages 323–344, August 1987. (page 46)

[Tuytelaars 04] T. Tuytelaars & L. Van Gool. *Matching widely separated views based on affine invariant regions.* International Journal of Computer Vision, vol. 59, no. 1, pages 61–85, 2004. (pages 44, 45)

[Varadarajan 74] V. Varadarajan. Lie groups, lie algebras, and their representation, volume 102 of *Graduate Texts in Mathematics.* Prentice-Hall Inc., 1974. (pages 4, 5, 6, 32)

[Vargas 05] M. Vargas & E. Malis. *Visual servoing based on an analytical homography decomposition.* In IEEE Conf. on Decision and Control and European Control Conf., pages 5379–5384, Sevilla, Spain, December 2005. (pages 48, 80, 100)

[Vercauteren 07] T. Vercauteren, X. Pennec, E. Malis, A. Perchant & N. Ayache. *Insight Into Efficient Image Registration Techniques and the Demons Algorithm.* In Information Processing in Medical Imaging, Kerkrade, Netherlands, June 2007. (pages 54, 60, 102)

[Wampler 86] C. W. Wampler. *Manipulator inverse kinematic solution based on vector formulations and damped least squares methods.* IEEE Transactions on System, Man, Cybernetics, vol. 16, no. 1, pages 93–101, 1986. (page 73)

[Wang 05] H. Wang, L. Dong, J. O'Daniel, R. Mohan, A. Garden, K Ang, D. Kuban, M. Bonnen, J. Chang & R. Cheung. *Validation of an accelerated 'demons' algorithm for deformable image registration in radiation therapy.* Physics in Medicine and Biology, vol. 50, no. 12, pages 2887–2905, June 2005. (page 54)

[Warner 71] Frank W. Warner. Foundations of differentiable manifolds and lie groups. Springer-Verlag, 1971. (pages 4, 5, 6, 32)

[Weiss 87] L. E. Weiss, A. C. Sanderson & C. P. Neuman. *Dynamic Sensor-Based Control of robots with Visual Feedback.* IEEE Journal of Robotics and Automation, vol. 3, no. 5, pages 404–417, October 1987. (page 72)

[Weng 92] J. Weng, P. Cohen & M. Herniou. *Camera calibration with distortion models and accuracy evaluation.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 10, pages 965–980, October 1992. (pages 9, 46)

[Wilson 96] W. J. Wilson, C. C. W. Hulls & G. S. Bell. *Relative End-Effector Control Using Cartesian Position-Based Visual Servoing.* IEEE Transactions on Robotics and Automation, vol. 12, no. 5, pages 684–696, October 1996. (page 89)

[Zhang 96] Z. Zhang & A. R. Hanson. *3D Reconstruction Based on Homography Mapping.* In ARPA Image Understanding Workshop, Palm Springs, CA, 1996. (page 48)

[Zhang 00] Z. Zhang. *A flexible new technique for camera calibration.* IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 22, no. 11, pages 1330–1334, 2000. (page 46)

[Zitova 03] B. Zitova & J. Flusser. *Image registration methods : a survey.* Image and Vision Computing, vol. 21, no. 11, pages 977–1000, October 2003. (page 53)