



HAL
open science

Multimodal sensor-based control for human-robot interaction

Andrea Cherubini

► **To cite this version:**

Andrea Cherubini. Multimodal sensor-based control for human-robot interaction. Robotics [cs.RO]. Université de Montpellier, 2016. tel-01419810

HAL Id: tel-01419810

<https://hal.science/tel-01419810>

Submitted on 19 Dec 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Habilitation à diriger des recherches

Délivrée par l'Université de Montpellier

Préparée au sein de l'école doctorale
I2S – Information, Structures, Systèmes

Et de l'unité de recherche
CNRS-UM LIRMM, IDH, UMR 5506

Spécialité:
SYAM – Systèmes Automatiques et Microélectroniques

Présentée par **Andrea Cherubini**
cherubini@lirmm.fr

Multimodal sensor-based control for human-robot interaction

Soutenue le 1er décembre 2016 devant le jury composé de

Joris DE SCHUTTER	PR	KU Leuven	Rapporteur
Bruno SICILIANO	PR	Università degli Studi di Napoli	Rapporteur
Olivier STASSE	DR	LAAS-CNRS	Rapporteur
François CHAUMETTE	DR	INRIA Rennes Bretagne-Atlantique	Examineur
Philippe FRAISSE	PR	LIRMM, Université de Montpellier	Examineur
Giuseppe ORIOLO	PR	Sapienza Università di Roma	Examineur



A Inès.

Acknowledgements

It would have been impossible for me to accomplish the research work presented in this manuscript without the fundamental help of friends and colleagues, first in Rennes and then in Montpellier. Many are acknowledged throughout the thesis, but some deserve a particular mention here.

At INRIA Rennes, François and Fabien shared with me the joys and sorrows of Cycab's outdoor navigation, constantly jeopardized by an explosive combination of hardware failures and Briton weather conditions. The Lagadic team spirit that François has built over the years, and that I have known from 2008 to 2011, remains unequalled among all the labs that I have visited.

Since my arrival at LIRMM, André, Abder and Philippe have mentored me and accompanied my work with authentic care and dedication. I have learned so much from them about a panoply of human and scientific qualities, required in research and teaching. Robin, an awesome person with whom it is a real pleasure to work, completes this picture with his rich scientific and technical insight! It is even more painful, in such a friendly context, to thank Fabrizio, whose legacy lives on in our daily work.

A final big "thank you!" goes to my great friend Michael Grech, for rereading this thesis and providing fundamental writing advice.

Abstract

The methods required to make humans and robots interact and collaborate are the subject of research in physical human-robot interaction (pHRI). One of the main issues, particularly when physical contact can occur between the two, is safety. A second fundamental requirement to establish pHRI is intuitive control by the human operator, in particular when s/he is non expert.

Sensor-based control can address both issues, by closing the perception-to-action loop, in a way that may well be assimilated to that of the lower motor neurons in our nervous system. Traditionally, heterogeneous sensor data was fed to fusion algorithms (e.g., Kalman or Bayesian-based), so as to provide state estimation for modeling the environment. However, since robot sensors generally measure different physical phenomena, it is preferable to use them directly in the low-level servo controller rather than to apply them to multi-sensory fusion or to design complex state machines. This idea, originally proposed in the hybrid position-force control paradigm, when extended to multiple sensors brings new challenges to the control design; challenges related to the task representation and to the sensor characteristics (synchronization, hybrid control, task compatibility, etc.).

The rationale behind my work has precisely been to use sensor-based control as a means to facilitate the physical interaction between robots and humans. In particular, I have utilized vision, proprioceptive force, touch and distance to address pHRI case studies, targeting different applications and robot platforms. My research has followed four main axes: teach-and-repeat navigation of wheeled mobile robots, collaborative industrial manipulation with safe physical interaction, force and visual control for interacting with humanoid robots, and shared robot control. Each of these axes will be presented here, before concluding with a general view of the issues at stake, and on the research projects that I plan to carry out in the upcoming years.

“ An android,” he said, “doesn’t care what happens to another android. That’s one of the indications we look for. ”

Philip K. Dick, *Do Androids Dream of Electric Sheep?*

Contents

1	Introduction	17
2	General Definitions	23
2.1	Robot Modeling	23
2.2	Task control	24
2.3	Visual servoing	24
2.4	Active force control	25
2.5	Integrating sensors at control level	27
3	Vision and distance sensing for “teach and repeat” navigation	29
3.1	Objectives and Motivation	30
3.2	State of art	31
3.3	Contribution	33
3.3.1	Defining the best features for visual navigation	33
3.3.2	Avoiding moving obstacles during navigation	35
3.3.3	Lidar-based teach and repeat	36
3.4	Conclusions	37
4	Vision and force sensing for industrial cobots	39
4.1	Objectives and Motivation	40
4.2	State of art	41
4.3	Contribution	43
4.3.1	Unified multimodal control for pHRI	44
4.3.2	Collaborative assembly cell	45
4.3.3	A safe adaptive damping controller	46
4.4	Conclusions	47
5	Force and visual control for interaction with humanoid robots	49
5.1	Objectives and Motivation	50
5.2	State of art	51
5.3	Contribution	53
5.3.1	Vision and force for autonomous car steering	53
5.3.2	Collaborative carrying using vision and force	54
5.3.3	Walking during physical collaboration	56

5.4	Conclusions	57
6	Shared human-robot control	59
6.1	Objectives and Motivation	60
6.2	State of art	61
6.3	Contribution	61
6.3.1	EMG-based robot hand grasping	62
6.3.2	BCI-based humanoid embodiment	63
6.3.3	Shared control for a humanoid robot car driver	63
6.4	Conclusions	64
7	Research perspectives	65
7.1	Sensor-based control	66
7.2	The mobile manipulator platform BAZAR	68
7.3	Envisioned collaborations	69
Appendices		
Appendix A	Activity report	73
A.1	Supervision	73
A.1.1	PhD students	73
A.1.2	Postdocs	77
A.1.3	Master students	78
A.2	Scientific activities	79
A.2.1	Project Management	79
A.2.2	Involvement in other projects	80
A.2.3	Editorial activities	80
A.3	Teaching activities	81
A.3.1	Teaching	81
A.3.2	Responsibilities	82
A.4	Dissemination of knowledge	83
A.4.1	List of publications	83
A.4.2	Invited seminars	87
A.4.3	Invitations to PhD defenses	87
A.4.4	Participation in Program Committees	88
A.4.5	National and International Expertise	89
A.4.6	Miscellaneous	89
Appendix B	Résumé en français	91
B.1	Mémoire	91
B.1.1	Activités de recherche (Chapitres 1-6)	91
B.1.2	Projet de recherche (Chapitre 7)	93
B.2	Rapport d'activités	94
B.2.1	Encadrement	94

B.2.2	Responsabilités scientifiques	96
B.2.3	Activités d'enseignement	97
B.2.4	Diffusion de travaux (rayonnement, vulgarisation)	98
Appendix C Selected journal articles		103
C.1	Visual Navigation of a Mobile Robot with Laser-based Collision Avoidance	104
C.2	Autonomous Visual navigation and Laser-based moving obstacle avoidance	121
C.3	A unified multimodal control framework for human-robot interaction	131
C.4	Collaborative manufacturing with physical human-robot interaction .	141
Bibliography		155

List of Symbols

HRI human-robot interaction.

pHRI physical human-robot interaction.

TCP tool control point.

dof degrees of freedom.

$\mathbf{x} \in \mathbb{SE}(3)$ pose of the TCP.

${}^B\mathbf{x}_A \in \mathbb{SE}(3)$ pose of frame A in frame B .

$\mathbf{v} \in \mathbb{R}^6$ kinematic screw of the TCP.

\mathbf{s} task vector.

\mathbf{J} task Jacobian.

\mathbf{q} joint angles.

X, Y image coordinates of a feature point.

$\mathbf{h} = [\mathbf{f}^\top \mathbf{n}^\top] \in \mathbb{R}^6$ contact wrench, composed of force \mathbf{f} , and torque \mathbf{n} .

BCI Brain Computer Interface.

EMG Electromyography.

Chapter 1

Introduction

Robot control is a mature field: one that is already being heavily commercialized in industry. However, the methods required to regulate interaction and collaboration between humans and robots have not been established yet. These are the subject of research in so called physical human-robot interaction (pHRI, [1, 2]). One of the main issues, particularly when physical contact can occur between the two, is *safety* [3]. Although there has been a recent push towards standardization (e.g., the ISO 13482:2014 for robots and robotic devices [4]), we are still in the initial stages. For instance, safety requirements, typically in terms of system reactivity (low bandwidth) and robustness, have not yet been attained.

A second fundamental requirement to establish pHRI is *intuitive control by the human operator*, even by non expert users. The robot should, for instance, be proactive in realizing the requested tasks [5–7]; in this regard, roboticists have drawn inspiration from behavioral and cognitive science. Finally, the reciprocal relationship, between humans and robots, is also fundamental. The robot should, for instance, be capable of *inferring the user’s intentions*, to interact more naturally from a human perspective [8, 9]. These three requirements are hampered by the unpredictability of human actions, which vary according to situations and individuals, impeding deterministic context modeling, and the use of standard feedback control.

In one looks at the literature, two major approaches for robot task execution have emerged: path planning [10] and sensor-based control [11]. The first, which is traditionally closer to the fields of computer science and artificial intelligence, breaks down the desired task into discrete subtasks that satisfy the system constraints and possibly optimize some aspect of the task/mission. Low-dimensional problems are generally solved with grid-based algorithms that overlay a grid on the robot’s configuration space. On the other hand, planning for high-dimensional systems under complex constraints is still computationally intractable (requiring excessively long time). Furthermore, this approach generally relies on a priori knowledge of the future robot and environment states over a time window; although path planning has proved its efficiency in a series of problems spanning from robot exploration to industrial manipulation and robotic surgery, it is hardly applicable to human-robot

interaction. The main reasons for this are, as pointed out above, the unpredictability of human behavior and the low bandwidth required for safety. To this end, sensor-based control offers more promising features since it closes the perception-to-action loop at a lower level, in a way that may well be assimilated to that of the lower motor neurons in our nervous system. A classic example is *image-based visual servoing* [11] which relies directly on image data to control the robot, without requiring a (computationally heavy) cognitive layer nor a precise model of the environment.

The rationale behind our work has precisely been to *use sensor-based control as a means to facilitate the interaction between robots and humans*. Although we acknowledge the need for other techniques within a complete HRI control framework (e.g., path planning as mentioned, machine learning, state observers, etc), the focus here is on direct servo control relying on data from one or multiple sensor/s.

In fact, recent technological developments on bio-inspired sensors have made these affordable and lightweight. It has therefore eased their use on robots, in particular on anthropomorphic ones (e.g., humanoids and dexterous hands). These sensors include RGB-D cameras, tactile skins, force/moment transducers, and capacitive proximity sensors. Traditionally, heterogeneous sensor data was fed to fusion algorithms (e.g., Kalman or Bayesian-based), so as to provide state estimation for modeling the environment. However, since these sensors generally measure different physical phenomena, it is preferable to use them directly in the low-level servo controller rather than to apply them to multi-sensory fusion or to design complex state machines. This idea, originally proposed in the hybrid position-force control paradigm [12], when extended to multiple sensors brings new challenges to the control design; challenges related to the task representation and to the characteristics of the sensors (synchronization, hybrid control, task compatibility, etc.).

All of the case studies addressed in our work are outlined in the scheme in Fig. 1.1. In all cases, the focus is on safety and on the dependability of robot interaction with the *Human*, hence the position of the latter at the center of the scheme. These goals are achieved with different combinations of sensing modalities, depending on the task at stake. The following robot senses have been used.

- *Vision*. This includes methods for processing and understanding images, to produce numeric or symbolic information, attempting to duplicate human sight. Although image processing is complex and computationally expensive, the richness of this sense is unique. Vision, fundamental throughout my PhD work, has maintained its relevance in the research presented here, helping to understand the environment and human intention, to react accordingly.
- *Force*. Although we group both under the term *force*, we differentiate *proprioceptive force* from *touch*, with the latter involving *direct* physical contact. Human *touch* (somatosensation), results from activation of neural receptors, mostly in the skin (but also in hair follicles, tongue, throat, and mucosa). These receptors have inspired the design of artificial tactile sensors, thoroughly

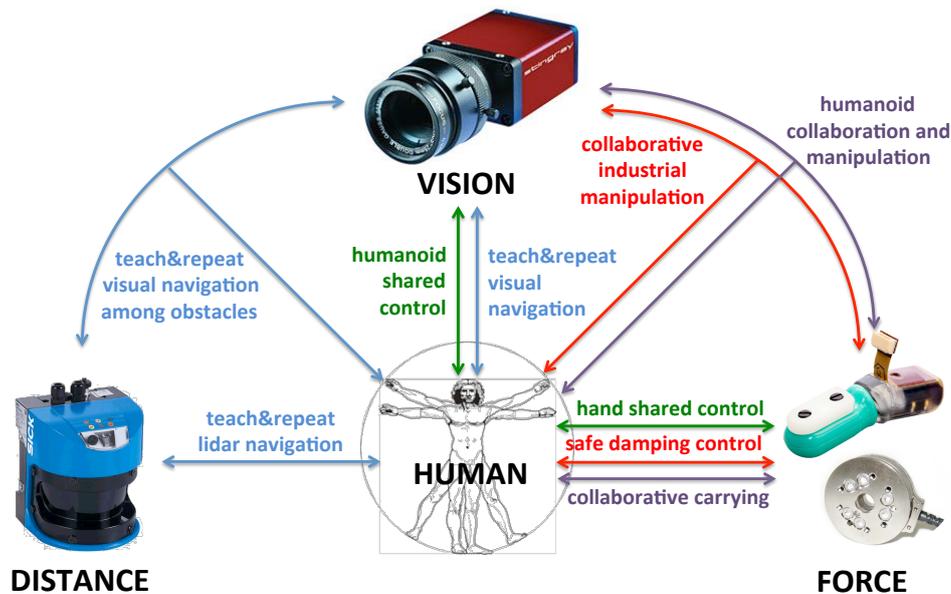


Figure 1.1: Outline of our work. The sensors shown in the figure are: SICK lidar (distance), AVT camera (vision), BioTac and ATI (force).

used for HRI. *Proprioceptive force*, on the other hand, can be seen as analogous to the sense of muscle force [13]. On robots, this can be reproduced by relying uniquely on the joint errors [14], or by inserting force/torque sensors along the kinematic chain. In our work, both were used by the robot to infer human intentions, and adapt to these, by relying on various admittance control schemes [15].

- *Distance*. Excluding stereopsis, which provides an indirect perception of depth using both eyes, this is the only type of data, among the four that is not directly measurable by the human senses. Numerous examples of this sense nevertheless exist in the mammal kingdom (e.g., among bats and whales), in the form of echolocation. In robotics, distance sensors (sonar, radar, or lidar) are thoroughly used to reconstruct either the environment (mapping problem), or the robot state (localization). The relevance of this particular “sense” in HRI is motivated by the direct relationship existing between the distance from obstacles (here, the human/s) and the danger of operation.

In our work, subsets of these sensing modes have been used to realize HRI, in various case studies. These are also indicated in Fig. 1.1. The case studies have targeted different applications and different robot platforms, specifically: mobile wheeled, humanoid, manipulator and hand robotic systems. Each of four research axes (which we outline below) is described in a different chapter. The four axes are color-coded in Fig. 1.1.

- **Chapter 3. Teach-and-repeat navigation of wheeled mobile robots** (blue in Fig. 1.1). We have exploited *visual* and *distance* sensing. The work was mainly developed during my postdoc at INRIA Rennes, in the context of the ANR CityVIP project (2008-2011). It was later extended (2013) by Christoph Sprunk, a PhD student from Albert-Ludwigs-Universitaet Freiburg (Germany). The results have been published in two Journals [16, 17] and in eight International Conference papers [18–25]. To enable autonomous, yet intuitively adaptable mobility (both for public transportation and logistics), we designed a framework, where non-expert human operators teach the path to be later replayed by the robot. Our contribution has been the integration of lidar-based obstacle avoidance to visual navigation (CityVIP) and the representation of the taught path via lidar scans instead of images (work of C. Sprunk, [25]). Collaboration with Freiburg is being pursued via the PHC Procope Project ViNavHuBot (2015-2016).
- **Chapter 4. Collaborative industrial manipulation with safe physical interaction** (red in Fig. 1.1). Here we have exploited *vision* and *force*. The work was developed at LIRMM, in the context of the French ANR Projects ICARO (2011-2014, work of postdoc Arnaud Meline), and SISCob (2014-2017, PhD thesis of Benjamin Navarro). The results have been published in two Journals [26, 27] and in two International Conference papers [28, 29]. We provide an industrial robot manipulator with two fundamental requirements for safe pHRI: human intentions should be easy to infer by the robot and the control should be intuited in relation to the human viewpoint. Our contributions are: the development of a unified controller, merging vision force and pose, with smooth transitions and weighted combinations of the sensor tasks (ANR ICARO), and the design of an ISO10218-compliant adaptive damping controller [29]. This research is ongoing in the context of projects EU H2020 Versatile, and Languedoc-Roussillon CoBot@LR, for both of which I am principal investigator.
- **Chapter 5. Force and visual control for interacting with humanoid robots** (purple in Fig. 1.1). Here we present research results of using *force* and *vision* to make real-size humanoid robots capable of interacting with both humans and machinery. The work was developed at LIRMM, in the context of the FP7 European Project RoboHow (2012-2016, PhD thesis of Don Joven Agravante) and of the CNRS PICS Project ViNCI (2013-2016, PhD thesis of Antonio Paolillo). The results have been published in five International Conference papers [30–34], and submitted to a Journal [35]. To make real-size humanoid robots capable of interacting both with humans and machinery, specific problems must be addressed. The main ones are whole-body control of highly redundant systems and walk stability. The contribution of Joven’s thesis has been the design of a complete sensor-based framework for human-humanoid collaborative carrying. Both whole-body control and walking under

sustained forces (from the human) have been addressed. During his 6 month stay at LIRMM under my supervision, Antonio developed a visio-haptic controller enabling a humanoid to manipulate sophisticated machinery (a car with steering wheel and pedal). Although interaction with humans is not explicit in this case, a human-like behavior must be designed for successful manipulation, since the machinery is human-tailored.

- **Chapter 6. Shared robot control with human in the loop** (green in Fig. 1.1). Here we present research on the use of *vision* and *force* as means of extending the senses of a human who controls/embodies robotic systems. The work was developed at LIRMM in the context of a collaboration with the INRIA Camin Team (the PhD thesis of Wafa Tigra), of the FP7 European Project VERE (2010-2015, the Phd thesis of Damien Petit), and of the post-doc of Antonio Paolillo. The results have been published in three International Conference papers [36–38], and two Journals [39, 40]. The works presented in this chapter push HRI to a higher stage, where the robot becomes an extension of the human, enabling the latter to expand his/her capabilities of interacting with the environment. Fundamental research issues to be addressed concern the shared control of the robotic system which, depending on the context and specifications, should be either autonomous, or partially or entirely controlled by the human. The case studies are: robotic hand control with electromyography, embodiment in a humanoid robot via a brain computer interface and car navigation through a humanoid robot driver. A contribution common to all works, has been the integration of senses within these shared control frameworks to aid human control when direct perception is not available.

All these aspects are detailed in each chapter of the manuscript, excluding *Chapter 2*, which gives general definitions. Each chapter also proposes avenues of future research. The final chapter, *Chapter 7* gives a more general view of the issues at stake in sensor-based control for HRI and describes the research projects that I plan to carry out in this field, in the upcoming years.

Chapter 2

General Definitions

This chapter provides the general definitions and fundamental notions that are common to the works presented in the manuscript. The reader is also invited to look at the “List of symbols” for quick reference on the notation.

2.1 Robot Modeling

We aim at controlling one or multiple **tool control points** (TCP) on an **open kinematic chain** robot, with j degrees of freedom (dof). The approach is common to the various platforms targeted in the manuscript, i.e., manipulators, wheeled systems, humanoids, and a robotic hand. For a wheeled system, we will consider as TCP, the center of rotation (e.g., the midpoint of the rear axis for a rear-drivable car). The tree kinematic structures (humanoid and hand), can be “split” in various open chains, to define multiple TCP, at convenience, on each branch.

The TCP pose in a fixed reference frame is noted:

$$\mathbf{x} = [x \ y \ z \ \phi_x \ \phi_y \ \phi_z]^\top \in \mathbb{SE}(3). \quad (2.1)$$

Its six components correspond to Cartesian coordinates and ZYX Euler angles¹.

In general, the pose of frame A in frame B , is defined as:

$${}^B\mathbf{x}_A = [{}^Bx_A \ {}^By_A \ {}^Bz_A \ {}^B\phi_{A,x} \ {}^B\phi_{A,y} \ {}^B\phi_{A,z}]^\top \in \mathbb{SE}(3). \quad (2.2)$$

Any point can be transformed from frame A to frame B by premultiplying its Cartesian coordinates by:

$${}^B\mathbf{T}_A = \begin{bmatrix} {}^B\mathbf{R}_A & {}^B\mathbf{t}_A \\ \mathbf{0} & 1 \end{bmatrix}, \quad (2.3)$$

with ${}^B\mathbf{R}_A$ the rotation matrix from A to B , and ${}^B\mathbf{t}_A = [{}^Bx_A \ {}^By_A \ {}^Bz_A]^\top$.

We also define the TCP kinematic screw (expressed in the fixed frame) as:

$$\mathbf{v} = [v_x \ v_y \ v_z \ \omega_x \ \omega_y \ \omega_z]^\top. \quad (2.4)$$

¹This choice is motivated by the fact that the inconveniences (e.g., singularities) typical of the Euler representation with respect to quaternions do not appear in the applications targeted here.

2.2 Task control

In all our works, we apply task control. We name \mathbf{s} the robot task vector, and \mathbf{u} the control inputs. The task dynamics are related to the control inputs by the **task Jacobian J**:

$$\dot{\mathbf{s}} = \mathbf{J}\mathbf{u}. \quad (2.5)$$

Then, \mathbf{s} can be regulated to a desired value \mathbf{s}^* , by applying *task control*:

$$\mathbf{u} = -\lambda\mathbf{J}^+(\mathbf{s} - \mathbf{s}^*), \quad (2.6)$$

with \mathbf{J}^+ the generalized inverse of \mathbf{J} , and λ a positive scalar gain.

When the control input coincides with the robot joint velocities $\dot{\mathbf{q}} \in \mathbb{R}^j$, and the task coincides with the TCP pose \mathbf{x} , we maintain this notation for the Jacobian and gain. In all other cases (for instance, visual servoing, see Sect. 2.3), a subscript will be added to \mathbf{J} and λ .

2.3 Visual servoing

This section is taken in part from [11]. Visual servoing refers to the use of computer vision data to control robot motion. The data may be acquired from a camera that is mounted directly on the robot, in which case motion of the robot induces camera motion, or the camera can be fixed in the workspace, so that it observes the robot motion from a stationary configuration.

Visual servoing schemes mainly differ in the way task \mathbf{s} is designed. The most common approaches are: **image-based** (\mathbf{s} is defined in the image state space) and the **position-based** (\mathbf{s} is defined in a 3D state space). Although, in this work, we have applied both, here we only briefly recall the image-based approach. The reason is that we have used position-based, by merely projecting the task from the image view to the workspace, and then applying classic task control (2.6).

In all the image-based controllers used in our work, we have used visible points to define the task:

$$\mathbf{s} = [X, Y]^T, \quad (2.7)$$

with X and Y the image point coordinates. If we use the camera kinematic screw:

$$\mathbf{v}_c = [v_{c,x} \ v_{c,y} \ v_{c,z} \ \omega_{c,x} \ \omega_{c,y} \ \omega_{c,z}]^T, \quad (2.8)$$

as control input \mathbf{u} , the Jacobian² relating $\dot{\mathbf{s}}$ and \mathbf{u} is:

$$\mathbf{J}_v = \begin{bmatrix} -\frac{1}{z} & 0 & \frac{X}{z} & -1 - X^2 & Y \\ 0 & -\frac{1}{z} & \frac{Y}{z} & -XY & -X \end{bmatrix}, \quad (2.9)$$

with z the depth of the point relative to the camera frame C .

²Named *interaction matrix* in the visual servoing literature.

The camera kinematic screw \mathbf{v}_c can be mapped to any other frame B , by applying the spatial motion transform matrix:

$${}^B\mathbf{V}_C = \begin{bmatrix} {}^B\mathbf{R}_C & [{}^B\mathbf{t}_C]_{\times}^B \mathbf{R}_C \\ \mathbf{0}_3 & {}^B\mathbf{R}_C \end{bmatrix}. \quad (2.10)$$

with $[{}^B\mathbf{t}_C]_{\times}$ the skew-symmetric matrix associated to ${}^B\mathbf{t}_C$. This is necessary whenever a control input \mathbf{u} , other than the camera screw, is utilized. Examples (treated in the manuscript) include manipulator joint velocities $\dot{\mathbf{q}}$, TCP kinematic screw \mathbf{v} , and wheeled robot translational/steering velocities, v_x, ω_z .

2.4 Active force control

This section is taken in part from [41]. In active force control (in contrast with passive force control), the compliance of the robot is ensured by the control system (instead of by the robot design and actuators). This approach usually requires the measurement of three translational force components, and three torques, which define the *wrench*:

$$\mathbf{h} = [\mathbf{f}^{\top} \ \mathbf{n}^{\top}]^{\top} = [f_x \ f_y \ f_z \ n_x \ n_y \ n_z]^{\top}. \quad (2.11)$$

Although the measure of \mathbf{h} is given in the sensor frame S , it can be transformed to any frame F , by applying ${}^F\mathbf{W}_S\mathbf{h}$, where:

$${}^F\mathbf{W}_S = \begin{bmatrix} {}^F\mathbf{R}_S & \mathbf{0}_3 \\ [{}^F\mathbf{t}_S]_{\times}^F \mathbf{R}_S & {}^F\mathbf{R}_S \end{bmatrix}. \quad (2.12)$$

The wrench \mathbf{h} is fed back to the controller and used to modify, or even generate online, the desired trajectory of the robot tool control point (TCP). Active interaction control strategies can be grouped into two categories.

- **Direct force control** offers the possibility of controlling the contact force and torque to a desired value via force feedback. This approach requires an explicit model of the interaction task. In fact, the user has to specify the desired TCP motion and the desired contact wrench (between TCP and environment or human) consistent with the constraints imposed by the environment. A widely adopted strategy is hybrid force/motion control [12], which aims at controlling the motion along the unconstrained task directions, and the wrench along the constrained task directions. This method, that will be further detailed in Sect. 2.5, allows simultaneous control of the contact wrench and of the TCP motion in two mutually independent subspaces.
- **Indirect force control**, instead, achieves force control via motion control without explicit closure of a force feedback loop. To this category belongs **impedance control** (or **admittance control**) [15, 42], where the deviation

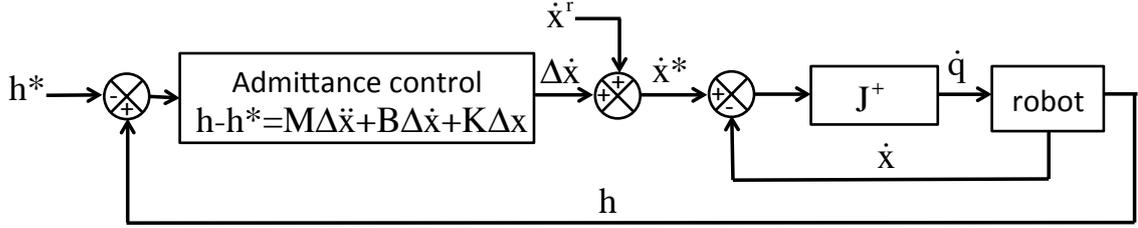


Figure 2.1: Block scheme of admittance control. The reference TCP motion $\dot{\mathbf{x}}^r$ is deformed by a virtual mass-spring-damper to yield the desired motion $\dot{\mathbf{x}}^*$ for robot kinematic control. In the figure, $\Delta\dot{\mathbf{x}} = \dot{\mathbf{x}}^* - \dot{\mathbf{x}}^r$.

of the TCP motion from the *reference* motion, due to the interaction with the environment, is related to the contact wrench \mathbf{h} through a virtual mechanical impedance/admittance with adjustable parameters. The TCP, under impedance (or admittance) control is described by an equivalent mass-spring-damper system with adjustable parameters. The most general equation is:

$$\mathbf{h} - \mathbf{h}^* = \mathbf{M}(\ddot{\mathbf{x}}^* - \ddot{\mathbf{x}}^r) + \mathbf{B}(\dot{\mathbf{x}}^* - \dot{\mathbf{x}}^r) + \mathbf{K}(\mathbf{x}^* - \mathbf{x}^r), \quad (2.13)$$

where \mathbf{h} and \mathbf{h}^* are respectively the estimated and desired TCP contact wrench. The vectors \mathbf{x}^r , $\dot{\mathbf{x}}^r$ and $\ddot{\mathbf{x}}^r$ correspond to the reference TCP pose, defined for instance by a high-level trajectory generator. Correspondingly, vectors \mathbf{x}^* , $\dot{\mathbf{x}}^*$ and $\ddot{\mathbf{x}}^*$ represent the deviated TCP pose with its first and second derivatives. Finally, matrices \mathbf{M} , \mathbf{B} and \mathbf{K} are the inertia, damping and stiffness parameters that define the desired virtual mass-spring-damper behavior. The general formulation in (2.13) can account for “different” control methods. Firstly, it can either be “impedance controlled” by controlling wrench \mathbf{h} , or “admittance controlled” by controlling position $\mathbf{x}(t)$. A typical block diagram of Admittance control is sketched in Fig. 2.1.

A drawback of direct control, is that it will not exploit sensor complementarity, since the range of feasible tasks is reduced to those that can be described in terms of constraint surfaces. Typically, in [43, 44], applications are limited to a single contact point on a planar surface. Considering a realistic application with unknown constraint location and complex contact geometry (typical with human-in-the-loop), it will be hard to configure in advance the controller (e.g., set the desired TCP wrench \mathbf{h}^*). Hence, for pHRI, it is more suitable to use the indirect (impedance/admittance) control framework since: i) it allows one to define a priori how the manipulator should react to unknown external force disturbances, ii) it can use, as reference trajectory $\mathbf{x}^r(t)$, the output of another sensor (e.g., vision) along any dof. For these reasons, we have utilized indirect force control, and more specifically **admittance control**, throughout this work.

2.5 Integrating sensors at control level

A first possibility would consist in combining different percepts by fusing the different information sources. This can be done with sensor fusion techniques, for example Cartesian frames are fused in [45], and fusion of proprioception and vision is done with an Extended Kalman Filter (EKF) in [46] or with task frame estimation in [47]. These approaches relate back to the merging of percepts by humans in [48, 49]. Although there is some merit to this, there is a resolvability issue [50], typically when merging force/torque and position/orientation(vision), since these are inherently different quantities. The camera and the force sensor measure different physical phenomena, while multi-sensory fusion aims at extracting a single information from disparate sensor data. In fact, motion and force are generally represented as dual vector spaces [51]. In addition, interpreting real force/torque sensor data is extremely complex, particularly in the presence of friction and when contact points locations are not known in advance. Thus, it is preferable to combine, directly at the control level, the actions simultaneously commanded by the various sensors. This has been proposed in [43] by extending hybrid position-force control [12] to account for vision.

In [43], the approaches for combining different sensors are classified into three general categories:

- **Traded:** this is the simplest strategy, since the robot is controlled by one sensor at a time, with the switches between sensors depending on the task and on the context.
- **Hybrid:** the sensors act at the same time but on different dof. This requires prior specification of a *task-frame* [44, 52, 53], i.e., a Cartesian frame divided into orthogonal directions. Then, each direction will be controlled only by one sensor. The directions are chosen using binary diagonal selection matrices, noted \mathbf{S} in this work.
- **Shared:** there is no separation in time (as for the traded approach) or in space (as for the hybrid approach). The control is shared by the various sensors throughout operation.

Although this classification was originally designed for merging vision and force, it can be extended to any number and type of sensors, and will be used throughout this manuscript to characterize our various design choices.

Chapter 3

Vision and distance sensing for “teach and repeat” navigation

This chapter presents our research on **Teach and repeat navigation of wheeled mobile robots**. The goal is to design an easy and intuitive framework, enabling a human to define the path to be followed by a mobile robot. At first, the human manually drives the robot. Then, the recorded sensed data is used by the robot, to autonomously follow the taught path. For this work, we have exploited *vision*, and *distance* as outlined in Fig. 3.1.

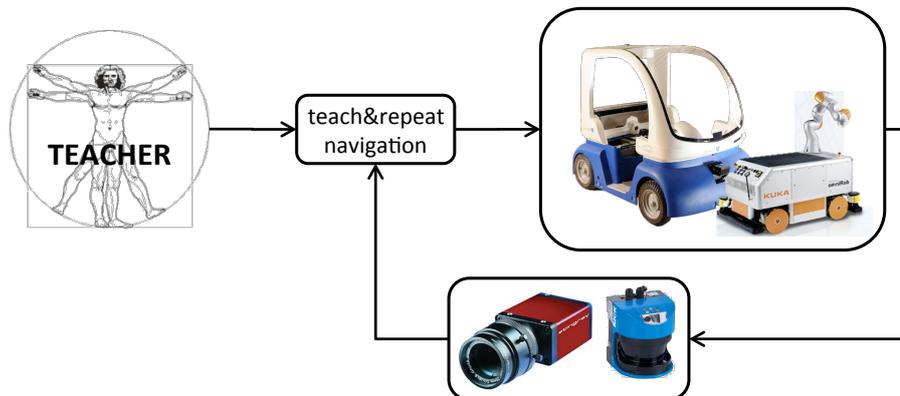


Figure 3.1: Vision and distance senses are used for teach-and-repeat navigation of wheeled robots (CyCab and omniRob). Here, HRI is asynchronous: the human acts as a *teacher*; teaching the path, to be later autonomously followed by the robot.

The research presented here has been carried out in the context of the French ANR Project CityVIP (2008-2011) and of a collaboration with the Albert-Ludwigs-Universitaet Freiburg (Germany), which then led to the French-German PHC Procope Project ViNavHuBot (2015-2016). The three main results have been:

1. Research on the choice of the best visual features to be used, in various navigation scenarios. This work was carried out during my postdoc in the context

of the **CityVIP** project and led to the publication of [18] and [19].

2. The design and validation of a framework, integrating lidar-based obstacle avoidance and visual navigation, to avoid static and moving obstacles that may appear in the repeat phase [16, 17], [20]- [24]. This work was also carried out during my postdoc, within **CityVIP**.
3. The design and validation of a teach-and-repeat framework where the taught path is represented via lidar scans instead of images. This work was carried out by **Christoph Sprunk** and **Gian Diego Tipaldi**, from the **Albert-Ludwigs-Universitaet Freiburg**, and led to the publication of [25].

The rest of the chapter is organized as follows. First, we present the motivation and objectives of the work (Sect. 3.1). Secondly, the current state of art in teach and repeat navigation frameworks (Sect. 3.2). Finally, we present the contribution of our work in Sect. 3.3, and the relevant conclusions in Sect. 3.4.

3.1 Objectives and Motivation

A great amount of robotics research focuses on vehicle guidance, with the ultimate goal of automatically reproducing the tasks usually performed by human drivers [54]. To this end, vision and lidar are fundamental in all environments (urban, or indoor, e.g., factories) where GPS localization is noisy or unreliable and when numerous perceivable natural features can aid navigation. A second fundamental aspect that is also common to all my research work on HRI is the need for intuitive frameworks allowing non-expert operators to naturally instruct and reprogram these mobile systems in changing environments.

Sensor-based navigation can be classified in model-based and appearance-based approaches. The first relies on the knowledge of a 3D model (map) of the environment, composed of geometric features (e.g., lines, planes, or points). The second, instead, operate directly in the sensor space. In appearance-based frameworks, the environment is represented as a topological graph where each node corresponds to a position and where a link between two nodes defines the possibility for the robot to move autonomously between the two positions. The nodes correspond to images (either from a camera or from a lidar) stored in a database. This paradigm is commonly realized within a teach-and-repeat framework [55–58]. During a preliminary *teach* phase, robot motion is controlled by a human operator, so as to generate the sensor database. The robot is then required to repeat the path by comparing the currently observed and the previously recorded sensor data (*repeat* phase). Most of these works rely on visual data. To design these controllers, the questions to be addressed include: What are the best visual features to be used? Can teach-and-repeat be extended to other sensed measures, e.g. distance?

Along with these questions, our research has addressed the integration of obstacle avoidance within a visual navigation framework. Obstacle avoidance consists in

either generating a collision-free trajectory to the goal or in decelerating to prevent collision when bypassing is impossible. In the past, obstacle avoidance has been integrated in sensor-based navigation by using the path geometry or the environment 3D model (e.g., including walls and doors). A reactive solution should be sought, instead, when the task is defined in the sensor space. Furthermore, the way the two tasks (navigation and obstacle avoidance) are to be realized, within a unique controller, must be carefully elaborated. Finally, the proposed solution should cope with moving obstacles which are a common feature in dynamic, real-world environments.

In summary, the motivation of the research presented in this chapter, is the study of the **best visual features for navigation**, the **integration of obstacle avoidance** and the **extension of the teach-and-repeat paradigm to lidar**.

3.2 State of art

Autonomous transportation platforms are increasingly employed for logistics applications in industrial contexts. Initially, these systems relied on guidance wires or on optical markers to drive the robot [59]. The paradigm later shifted to the use of on-board sensors for localization and navigation, while again augmenting the environment with artificial markers [60]. Impressive results have been obtained with an infrastructure-free robot that uses three cameras to navigate, with naturally occurring visual cues (mainly on the warehouse floor) learned during operation [58]. This framework is inspired by the visual teach-and-repeat approach originally presented in [55], where a particular motion (e.g., “go forward”, “turn left”) is associated to each taught image. This enables the robot to move from the current to the next image in the database. Aside logistics, teach-and-repeat approaches have also been applied to outdoor navigation (e.g., targeting public transportation, [56]).

In both indoor and outdoor contexts, an important choice concerns the database representation. In some works, three dimensional reconstruction is used, whereas other frameworks rely uniquely on image information. In [56], a 3D representation of the taught path is built from the image sequence and a path following controller is used for navigation. A similar approach, using omnidirectional cameras has been proposed in [61]. There, the path is represented by a sequence of images, used to reconstruct a local 3D map. Feature tracking is exploited, to localize the robot, and path following is then done with a sequence of homing vectors to each image. In general, 3D reconstruction is unnecessary, since moving from one key image to the next, can also be done by relying uniquely on visual servoing (see Sect. 2.3). In [62] for instance, a proportional control on the position of the feature centroid in current and database images drives the robot steering angle while the translational velocity is constant. The controller in [57] exploits angular information from features matched in panoramic images. In [63], epipolar geometry (without explicit 3D reconstruction) is used to drive a nonholonomic robot to a desired pose.

A drawback of such visual servoing schemes, is that whenever the initial and de-

sired configurations are far from each other, convergence can be difficult to ensure. As shown in [64], a possible solution is the use of a planning step joint with the servoing to limit the tracking error along the planned path. In [65], an approach that uses an image-based visual servoing system to track a desired timed feature trajectory is presented. In contrast, a time-independent solution for tracking image trajectories, based on a movement flow to determine the desired configuration from the current one is described in [66]. With these approaches, the desired configuration remains near the current one and the local stability of the image-based visual servoing is always assured. However, none of these approaches has been applied to outdoor navigation, and assessed in terms of 3D camera trajectory accuracy.

In any teach-and-repeat framework, the robot should also be capable of avoiding obstacles that have appeared on the path between the two phases. Obstacle avoidance has been integrated in many model-based navigation schemes. Simultaneous obstacle avoidance and path following are presented in [67], where the geometry of the path is perfectly known. In [68], obstacles are circumnavigated while following a path; the radius of the obstacles (assumed cylindrical) is known a priori. In practice, all these methods are based on the environment 3D model, including the geometry of the path. In contrast, in appearance-based navigation neither the environment nor the obstacle models are available. One of the most common techniques for model-free obstacle avoidance is the potential field method, originally introduced in [69]. The gap between global path planning and real-time sensor-based control has been closed in [70], where a set of trajectories (arcs of circles or “tentacles”) is evaluated for navigating. However, in [70], the trajectory computation relies on GPS way points and hence on the robot’s pose.

If the obstacles are moving, as is common in real environments (where pedestrians, bicycles and cars are present), the problem is even more challenging. In the literature, researchers have taken into account the obstacle velocities to deal with this issue. The approach presented in [71] is one of the first where static and moving obstacles are avoided, based on the objects’ current positions and velocities relative to the robot. The maneuvers are generated by selecting robot velocities outside the *velocity obstacles*, that would provoke a collision at some future time. This paradigm has been extended in [72] to take into account unpredictably moving obstacles. This has been done by using reachability sets to find matching constraints in the velocity space, called Velocity Obstacle Sets. Another pioneer method, that has inspired many others, is the Dynamic Window [73] that is derived directly from the dynamics of the robot and is especially designed to deal with constrained velocities and accelerations. The method consists of two steps: first, a valid subset of the control space is generated, then an optimal solution (driving the robot with maximum obstacle clearance) is sought within it. A generalization of the dynamic map, that accounts for moving obstacle velocities and shapes, is presented in [74], where a union of polygonal zones corresponding to the non-admissible velocities controls the robot to prevent collisions.

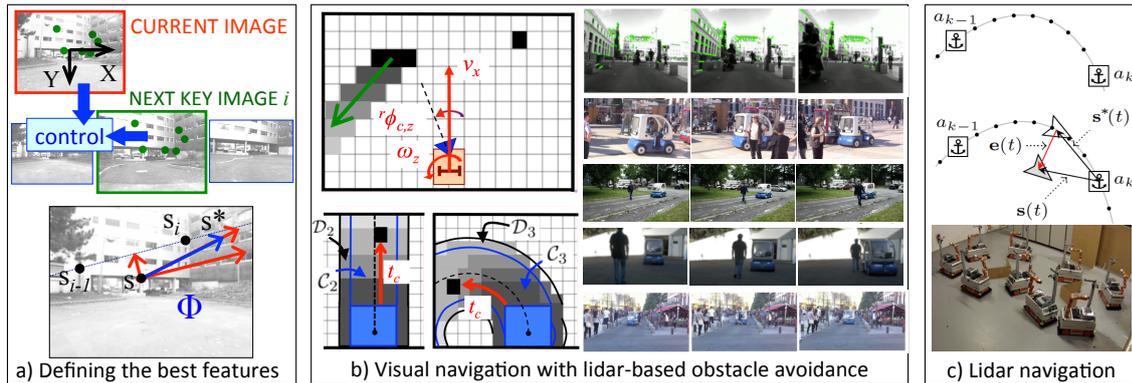


Figure 3.2: Teach and repeat navigation. a) Defining the best features. Top: teach-and-repeat paradigm. Bottom: design of the time-independent desired feature. b) Visual navigation with a lidar-based obstacle avoidance. Left: occupancy grid and tentacles. Right: experiments. c) Lidar navigation. Top: taught data. Center: control error. Bottom: nine steps of the navigation experiment.

3.3 Contribution

The research described here addresses sensor-based (*vision* and *distance*) teach-and-repeat wheeled navigation. Most of our work focused on outdoor visual navigation: a wheeled vehicle equipped with an actuated pinhole camera and with a forward-looking lidar must follow a path represented by key images, without colliding with the ground obstacles. For this (see Fig. 3.2(a),top), we rely on the image coordinates of the n points matched in the current and next key images, respectively $(X_1, Y_1) \dots (X_n, Y_n)$, and $(X_1^*, Y_1^*) \dots (X_n^*, Y_n^*)$. The most recent work, instead (Sect. 3.3.3), extends teach-and-repeat to indoor lidar-based navigation. All applications are shown in Fig. 3.2, and in videos on the internet¹. Our contributions have been: the definition of the best visual features to be used [18,19], the avoidance of obstacles (including moving ones) during navigation [16,17], [20]- [24], and the use of lidar measures for teach-and-repeat [25]. All are detailed below.

3.3.1 Defining the best features for visual navigation

We focus on appearance-based navigation, with the path defined as a topological graph (without forks), represented by a database of ordered key (taught) images. Navigation is divided in a series of subtasks, each consisting of driving the robot towards the next key image in the database. In such a framework, the choice of the features to be used in the control feedback law is fundamental. In particular, we must determine to what extent is 3D reconstruction, via structure from motion (SfM) methods necessary/useful.

¹<https://www.youtube.com/watch?v=FYuqlQDj3yg>

<https://team.inria.fr/lagadic/demo/demo-cycab-vis-navigation/vis-navigation>

The main contribution of [18] is the quantitative comparison between six controllers for nonholonomic appearance-based navigation using monocular vision. We set the forward velocity v_x to a constant positive value, and design a visual servo controller (using the image jacobian, see Sect. 2.3) for the angular velocity ω_z , with four alternative feature vectors:

$$\mathbf{s} = [X_1 Y_1 \dots X_n Y_n]^\top, \quad \mathbf{s} = [\bar{X} \bar{Y}]^\top, \quad s = \bar{X}, \quad s = \phi_z. \quad (3.1)$$

These are respectively: the image coordinates of all matched points, those of their centroid, the image abscissa of the centroid, and the robot heading error (obtained with SfM). For the first two feature vectors, we also compare the behavior of a controller where the point depths in the image jacobian are precisely estimated with SfM, with one where all depths are approximated by a pre-tuned, *uniform* value. The third feature, \bar{X} , is used in a simple proportional controller without image Jacobian, as in [62]. We have compared the accuracy of all six controllers, both in the image and in the 3D pose state space (since both are fundamental for precise unmanned navigation). The experiments showed that the 4 controllers, which combine both image data and feature depth outperform the 2 which utilize only 3D data (ϕ_z), or only image data (\bar{X}). Besides, although 3 controllers require SfM, for those using the image jacobian, a large 3D reconstruction error (e.g., due to coarse camera calibration) can be tolerated, without jeopardizing performance. Since SfM introduces computational delay at run time and increases sensitivity to image noise, a valid alternative is to use the uniform depth controller. Given these results, in all our following works navigation was image, instead of position-based, as in [56, 61].

The results of that work inspired [19], where we designed a time-independent varying desired visual feature \mathbf{s}^* , to replace the next key image feature. The goal is to maintain the task error norm $\|\mathbf{s} - \mathbf{s}^*\|$ constant, while tracking each key image, to guarantee target visibility, even when the current and key camera configurations are far away from each other. Ideally, this can be done by computing the trajectory of the features between pairs of consecutive key images, using SfM. Still we aim at avoiding SfM, and timed trajectory tracking as much as possible. Our varying feature is designed using a vector field $\Phi(\mathbf{s})$, derived from the previous and next key feature values, \mathbf{s}_{i-1} and \mathbf{s}_i :

$$\mathbf{s}^* = \mathbf{s} + \|\mathbf{s}_i - \mathbf{s}_{i-1}\| \frac{\Phi(\mathbf{s})}{\|\Phi(\mathbf{s})\|}. \quad (3.2)$$

Considering points as features, Φ is determined by summing two fields, one tangential (Φ_t), and one normal (Φ_n) to the line connecting \mathbf{s}_{i-1} and \mathbf{s}_i . This is outlined in Fig. 3.2(a),bottom. The experiments show the advantages of the varying desired feature with respect to a fixed one, in the image as well as in the 3D state space. Besides, the performances are close to those obtained if the database contained *all* images recorded during teaching (not only the key ones), while requiring less disk space.

3.3.2 Avoiding moving obstacles during navigation

Based on these results, we focused on obstacle avoidance, within the same framework: while following the taught path, the robot must avoid obstacles which were not present during teaching and which are sensed by an on-board lidar. To this end, we designed a novel control scheme guaranteeing that obstacle avoidance and navigation are achieved simultaneously. In the presence of obstacles, the camera pan angle ${}^r\phi_{c,z}$ is velocity controlled, to maintain scene visibility, while the robot circumnavigates the obstacle. We also deal with unavoidable obstacles, which make the robot reduce its translational velocity v_x or stop. The situation risk (represented by scalar function $S \in [0, 1]$) and the consequent behavior are determined via a tentacle approach that differs from the original [70] by not requiring the robot pose.

Our tentacle-based approach is based on an occupancy grid linked to the robot (see Fig. 3.2(b), left). Any grid cell is considered to be currently occupied (black) if an obstacle has been sensed there. The cells' linear velocities (green in the figure) are estimated using a Kalman-based observer [17, 24] to predict the obstacle positions (hence, possible future collisions). Each tentacle j is a semicircle (drivable path), of curvature κ_j , tangent to the robot sagittal plane at the robot center. It is characterized by two classification areas (*dangerous* \mathcal{D}_j and *collision* $\mathcal{C}_j \subset \mathcal{D}_j$), which are respectively used to determine the safest tentacle curvature κ_b , and the (eventual) time to collision t_c .

In contrast with [75], redundancy is not necessary in our controller since we design the two tasks (in the *safe* and *unsafe* contexts) to be independent. Specifically, we control the velocities of the robot (v_x, ω_z) and camera (${}^r\dot{\phi}_{c,z}$) through:

$$\begin{bmatrix} v_x \\ \omega_z \\ {}^r\dot{\phi}_{c,z} \end{bmatrix} = (1 - S) \mathbf{J}_s^{-1} \begin{bmatrix} v_s \\ -\lambda_x (\bar{X} - \bar{X}^*) \\ -\lambda_\phi {}^r\phi_{c,z} \end{bmatrix} + S \mathbf{J}_u^{-1} \begin{bmatrix} v_u \\ -\lambda_x (\bar{X} - \bar{X}^*) \\ \kappa_b v_u \end{bmatrix}. \quad (3.3)$$

The variables in the equation are:

- S the risk function on the best tentacle, null iff the best tentacle is clear,
- $\mathbf{J}_{s/u}$ the Jacobians relating safe/unsafe tasks to the control inputs²,
- $v_s > 0$ the translational velocity in the *safe context* (i.e., when $S = 0$), maximal on straight path portions, and smoothly decreasing at sharp turns,
- $v_u \in [0, v_s]$ the translational velocity in the *unsafe context* ($S = 1$), decreasing (eventually to zero) as time to collision t_c decreases,
- $\lambda_x > 0$ and $\lambda_\phi > 0$ empirical gains,
- \bar{X} and \bar{X}^* the points' centroid abscissas in the current and next key image,

²In [16], we provide the sufficient conditions for inverting \mathbf{J}_s and \mathbf{J}_u .

- κ_b the curvature of the best tentacle, determined with a rule, detailed in [16].

In the *safe* context ($S = 0$), the robot acts as in the teaching phase, i.e. following the taught path with the camera looking forward and translational velocity set to v_s . If the current pan angle ${}^r\phi_{c,z}$ is non-null (i.e., if the robot just avoided an obstacle), it is servoed back to 0. Instead, in the *unsafe* context ($S = 1$) the robot circumnavigates the obstacles, by following the best tentacle at velocity v_u . Since this heading variation drives the robot away from the taught path, the camera pan angle must be actuated to maintain visibility of the database features. Referring to the definitions of Sect. 2.5, our approach can be classified as *hybrid* when $S = 0$ or $S = 1$ (i.e., vision and distance operate on independent components of the task vector), and as *shared* in all intermediate situations $S \in (0, 1)$.

Controller (3.3) is compact, leading to a smooth behaviour. We guarantee asymptotic stability of the visual task in the presence of non occluding obstacles and prove that (3.3) does not present any local minima. Our approach is merely appearance-based, hence simple and flexible: it requires the database of key images and the lidar data, but no model of the environment. Hence, there is no need for sensor data fusion or planning, both of which can be computationally costly and require precise calibration of the camera/scanner pair.

Our approach was validated in a series of real outdoor experiments with the CyCab robot (see right of Fig. 3.2(b)). This was the first time that obstacle avoidance and visual navigation, merged directly at the control level (without the need for planning) were validated in real-life, unpredictable urban experiments.

3.3.3 Lidar-based teach and repeat

In a fruitful collaboration with the University of Freiburg [25], teach-and-repeat navigation was extended to lidar data, in the context of indoor logistic applications. Current autonomous transportation systems need globally consistent metric maps, to allow flexible route planning. This however requires an expert user to redesign the map and docking positions at each change of the environment or transportation routes. Instead, to make automation profitable for frequently changing production processes, systems should be intuitively usable by non-expert workers. To this end, we proposed a natural approach where the user just needs to demonstrate a new trajectory that is then reproduced with high accuracy and precision.

Our scheme relies on localizing the robot along the taught trajectory represented by a sequence of odometry-based *anchor poses* \mathbf{a}_k (shown in Fig. 3.2(c), top), with associated recorded lidar data. Scan matching provides the current pose of the robot with respect to the next anchor point, i.e., task vector $\mathbf{s}(t)$. By subtracting the desired task $\mathbf{s}^*(t)$, from $\mathbf{s}(t)$, we derive the task error $\mathbf{e}(t)$ (Fig. 3.2(c), center), that is then input to a standard feedback controller.

Our approach has two advantages: it does not require globally consistent metric mapping, and through the direct use of raw sensor data, it avoids errors that may

be introduced by grid map approximation. It differs from standard appearance-based navigation, in the sensor modality (lidar scans vs. camera images) and in the state space (configuration vs. sensor space). Real-world experiments (Fig. 3.2(c) bottom) with both a holonomic (Kuka³ omniRob) and a differential drive platform, demonstrate that the accuracy of our scheme outperforms that of standard Monte Carlo localization.

3.4 Conclusions

This chapter outlined our main research results in teach-and-repeat navigation for mobile robots. Human interaction is not physical, nor synchronous, but is present, since the operator teaches the path beforehand. All works rely on vision and/or distance as sensing modalities, and both are merged in the controller, to circumnavigate obstacles, in Sect. 3.3.2.

The main contribution is in the integration of collision avoidance. In fact, prior to our works [16, 17], [20]- [24], appearance-based navigation frameworks had never been extended to account for obstacles (neither static nor in motion). Other – minor – contributions are the design of a time-independent varying desired image feature, to improve pose accuracy, and the extension of teach-and-repeat to lidar data.

In future work, we plan to extend the tentacle-based approach to robots with other kinematic models, e.g., the pseudo-omnidirectional Neobotix MPO700 platform present at LIRMM. On highly redundant systems, such as the humanoid HRP-4 (also at LIRMM) perspective work could also include automatic prevention of the visual occlusions provoked by obstacles.

³<http://www.kuka-robotics.com>

Chapter 4

Vision and force sensing for industrial cobots

This chapter presents our research on **Collaborative industrial manipulation with safe physical interaction**. The goal is to provide an industrial robot manipulator (arm and possibly gripper or hand) with the three fundamental requirements of pHRI: the human intention should be easy to infer by the robot, the control should be intuitive from the human viewpoint and the designed controller should be safe for both human and robot. To this end, we have exploited *vision* and *force*, as outlined in Fig. 4.1.

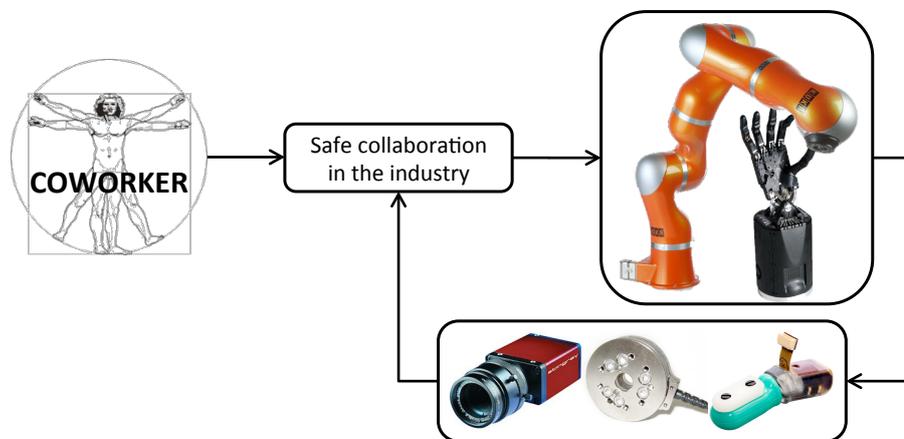


Figure 4.1: Vision and force sensing for collaborative robots (*cobots*) Kuka LWR and Shadow Hand.

The research was carried out in the framework of the French ANR Projects ICARO (2011-2014), and SISCob (2014-2017), and has lead to three main results:

1. A unified controller for collaborative interaction, merging vision and force with smooth transitions and weighted combinations of the sensor tasks. This work

has been published in [26,28], in collaboration with postdoc **Arnaud Meline**, and in the context of the **ICARO** Project.

2. The design and validation of a robot manufacturing cell, for homokinetic joint assembly [27]. This work was also carried out in the context of **ICARO**.
3. The design and validation, on a hand-arm robot, of an adaptive damping controller that fulfills the ISO10218 safety standard. This work has been published in [29], by PhD student **Benjamin Navarro**, and in the context of the **ANR SISCob** Project.

The rest of the chapter is organized as follows: first we present the motivation and objectives (Sect. 4.1), and the current state of art in collaborative industrial robotics (Sect. 4.2). Then, we present the contribution of our work in Sect. 4.3, and we conclude in Sect. 4.4.

4.1 Objectives and Motivation

The concept of *cobots*, i.e., robots collaborating with human workers in manufacturing assembly lines, dates back to the pioneer work [76]. In fact, cobots - designed for the assembly line worker – can reduce ergonomic concerns that arise due to on-the-job physical and cognitive loading, while improving safety, quality and productivity. This is a key issue, since according to statistics of the Occupational Safety and Health Department of the US Department of Labour, more than 30% of European manufacturing workers are affected by lower back pain, leading to enormous social and economic costs. Thorough surveys on human-machine cooperation in manufacturing lines are provided in [77,78]. Both point to the absence of high level human-robot collaboration (if one excludes “Intelligent Lift Assistants”) and to the need for more advanced collaboration: although humans remain indispensable in many assembly operations, ergonomic tools assisting their duties are fundamental. Although some automotive manufacturers are gradually introducing robots in their production line [79,80], a crucial question persists: how should a collaborative robotic cell be designed? The ultimate goal would be to have the adaptability of humans merged with the high performance of robots in terms of precision, speed and payload [81].

Furthermore, robots must behave safely, especially when operators are present in their workspace. Higher safety levels need to be attained when physical contact occurs between the two. This makes it indispensable to define safety and dependability metrics [82–84]. These can contribute to the definition of standards, such as the recent ISO 10218 “Safety requirements for industrial robots” [85], that imposes velocity, power and contact force bounds to the robot tool control point, in the presence of a human.

To guarantee safety, particularly during physical contact phases, human-robot interaction has largely relied on the use of force/torque control [41]. Even when there

is no direct contact, the robot should reactively adapt to sudden changes in the environment, especially because of unpredictable human behaviour [86]. This can be done through vision, which is also useful to infer/guess the underlying intention behind motion, as done in [87], to track the human hand during hand-over. Hence, vision and force should be used concurrently, since the information they provide is complementary. The integration of the two however requires operating different modes and managing sudden signal changes from heterogeneous sensor data. Important research problems include: What information would be helpful? How can this information be reliably obtained in the context of the task and platform? How/where should this information be used?

In summary, the motivation behind the research presented here is the design of **collaborative industrial robot cells**, with **safety** as a fundamental requirement, using **vision-force control**, as means for attaining this.

4.2 State of art

In the field of collaborative robotics, many solutions for realizing safe collaborative tasks have been explored in recent years. Although few of these solutions have been transferred to the industry, we hereby list some of the most relevant theoretical works. In [88], a deformation-tracking impedance control strategy is designed to enable robot interaction with environments of unknown geometrical and mechanical properties. For successful interaction with unknown environments and operators, the robot should behave in a human-like manner. This is the target of the research in [89] and [90]: a human-like learning controller is designed, to minimize motion error and effort during interaction tasks. Simulations show that this controller is a good model of human-motor adaptation, even in the absence of direct force sensing. A robust controller for a collaborative robot in the automotive industry, is extended in [91], to manage not only the interaction between an industrial robot and a stiff environment, but also human-robot-environment and human-robot-human-environment interactions.

Other researchers have focused on industrial applications. For example, an industrial robot controller, incorporating compliance of the joints with the environment, is presented in [92]. The desired pose of the tool center point is computed from the force error. Parallel control considers a reference trajectory, while allowing feedforward in force-controlled directions. Although the method is designed for industrial assembly tasks, it does not take into account the presence of humans in the loop. In contrast, Erden and colleagues [9,93,94] have thoroughly studied an industrial task that directly involves a human operator: manual welding. In [9], a physically interacting controller is developed for a manipulator robot arm: the human applies forces on the robot, to make it behave as s/he likes. The assistant robot is then designed in [93]: as the human controls the welding direction and speed, the robot suppresses involuntary vibrations (e.g., caused by novice welders). The results show a consid-

erable improvement in the welders performance when they are assisted. Finally, [94] presents a study of end-point impedance measurement at human hand, with professional and novice welders. The results support the hypothesis that impedance measurements could be used as a skill level indicator, to differentiate the welding performance levels. Similar works are presented in [95] and [96]. In [95], an operator teaches tasks to a robotic manipulator, by manually guiding its TCP. For this, the authors design a virtual tool, whose dynamics the operator should feel when interacting with the robot. An admittance controller driven by the measurements of a force/torque sensor is designed to ensure the desired virtual dynamic behaviour. The second paper [96] addresses the problem of controlling a robot arm, executing a cooperative task with a human, who guides it physically. This problem is tackled by allowing the TCP to comply according to an impedance controller [15] defined in the Cartesian space. Redundancy ensures the stability of the coupled human-robot system through inertial decoupling at the TCP.

As already mentioned, safety is a crucial element in the design of all these collaborative industrial robotic cells. To our knowledge, present-day collaborative robot manufacturers [97] fulfill the ISO10218 standard by saturating the velocity, stopping the robot, or using expensive hardware solutions. Although novel safe actuation systems have been recently proposed in the literature [98], [99], [100], these are not always easily affordable or adapted for any robotic system. An alternative comes from control, although, to the best of our knowledge, the only work that tackles the ISO1028-2011 is [101], where only the force limitation is considered. As for most of the works cited above, a solution comes from impedance control [15] and its modified versions for force tracking [102], force limitation [103], adaptive damping [104] or exploiting redundancy [105].

In general, having a human being as a physical collaborator requires revisiting some aspects, such as the choice of the impedance parameters. For instance, variable impedance control is used for human-robot collaboration in [106,107], with parameters obtained from human-human experiments. In fact, mechanical impedance was shown to provide a good model of the human being in [108]. A variable damping controller is defined in [109] using the derivative of the interaction force. A method for improving impedance control consists in utilizing an estimate of the human intended motion [110]. An example is [6], where machine learning is used to obtain a model of the task, which is then utilized within an adaptive impedance framework.

Finally, let us briefly review work where vision and force are merged - directly at the control level - for HRI. One such work is [111], where force- and vision-based control are used to avoid collisions, while tracking human motion during interaction. Force sensing, along with minimum jerk based estimation of the human motion, is used by Maeda et al. [5] within a virtual compliance framework for cooperative manipulation. The authors of [112] present a system (including a wearable suit with inertial motion capture sensors) for precise localization of a human operator in industrial environments. If the robot is realizing a task and a human enters the safe area, the robot will pause until the human leaves.

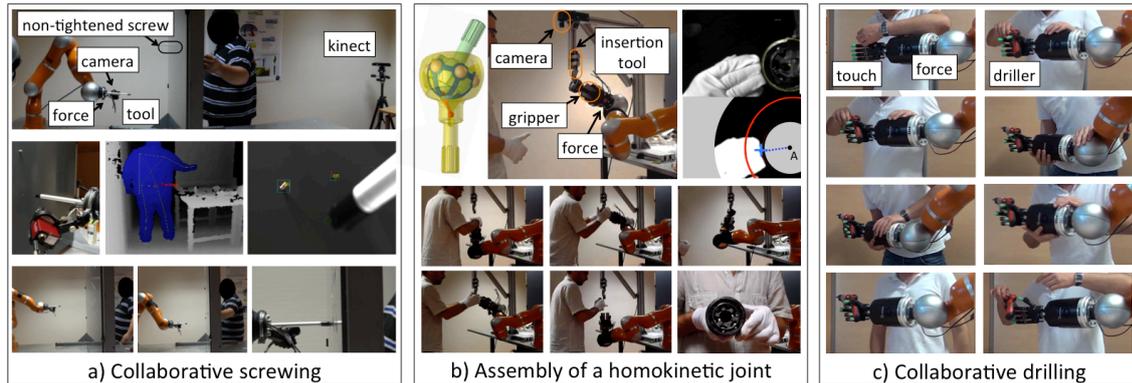


Figure 4.2: Collaborative robot applications. a) Collaborative screwing. Top: experimental setup. Center: on-board camera, kinect view and on-board camera view. Bottom: three phases of the operation. b) Assembly of a homokinetic joint. Top: Rzeppa joint, setup and raw/processed images. Bottom: six phases of the assembly. c) Collaborative drilling: eight phases of the operation.

4.3 Contribution

The research described in this chapter was driven by industrial requirements. It targeted three applications: collaborative screwing (case study proposed by AIRBUS [26, 28]), collaborative assembly of a homokinetic joint (proposed by PSA, Peugeot Citroën [27]) and collaborative drilling, complying with the ISO10218 standard [29]. These applications are shown in Fig. 4.2, and in videos on the IDH LIRMM youtube channel¹. For the first two scenarios, we only utilized a Kuka LWR arm, whereas in the third one a Shadow robotic hand² was also mounted on the arm. In the first two scenarios, the control relied on *force proprioception* (external wrench estimated, via the FRI Interface³.) and *vision* (including a kinect, in the first one), whereas in the third scenario, we used *force proprioception* and *touch*, measured by the BioTac⁴ on the robot hand. The contributions of the three works have respectively been: a unified multimodal control framework for pHRI, the design of a collaborative cell with the robot alternating proactive and compliant behaviors, and an adaptive damping controller guaranteeing safety. These are detailed hereby.

¹<https://www.youtube.com/watch?v=P8wfQ5tOa5E>

<https://www.youtube.com/watch?v=3KWduKKSyy8>

<https://www.youtube.com/watch?v=iOuhFKp31xY>

²<http://www.shadowrobot.com/>

³<http://cs.stanford.edu/people/tkr/fri/html/>

⁴<http://syntouchllc.com/>

4.3.1 Unified multimodal control for pHRI

In the first application (Fig. 4.2(a)), the robot aids a human operator in a screwing operation. The two operate on opposite sides of a flank, where a series of screws must be inserted. The required operations are: the human inserts the screws in the holes; the robot touches the screws with its TCP properly aligned (bolt tightening is out of scope). In such an application (which resembles “peg-and-hole”, but adding the human-in-the-loop), it seems natural to exploit vision and force complementarity. Indeed, while the robot is far from any physical constraints, image-based control is useful for nearing the parts to be mated, but as the robot tip approaches the environment, unpredicted contacts can occur. Then, a force controller can intervene on some dof to provide compliance and guide the manipulator to the desired pose. Given these specifications, we have drawn inspiration from *hybrid* schemes (see Sect. 2.5), in particular from our previous work on vision and distance-based navigation (Sect. 3.3.2).

Inspired by inverse kinematic control [113, 114], we have designed a unified task formalism. The contributions, with regards to classic hybrid vision-force-position control (typically to [115]), is that we can guarantee global asymptotic stability, while: enabling smooth transitions (homotopies) and weighted combinations (even on the same dof) of the different sensor tasks.

We hereby recall the formulation of our approach (further details are in [26]). Let $k \leq 6$ be the dimension of the operational space associated to the TCP. Consider n senses and, for each sense, the task vector $\mathbf{s}_m \in \mathbb{R}^k$, with $m = 1, \dots, n$ (e.g., if vision and force are used, $n = 2$). A combination of tasks defined by different senses is realizable as long as its size is also k . The tasks are selected via n positive definite square diagonal selection matrices, denoted \mathbf{S}_m (generalizing the scalar activation function S used in Sect. 3.3.2). The k -dimensional hybrid task \mathbf{s} to be realized is:

$$\dot{\mathbf{s}} = \mathbf{S}\dot{\bar{\mathbf{s}}}, \quad \text{with } \mathbf{S} = [\mathbf{S}_1 \dots \mathbf{S}_n] \in \mathbb{R}^k \times \mathbb{R}^{kn}, \quad \text{and } \bar{\mathbf{s}} = [\mathbf{s}_1^\top \dots \mathbf{s}_n^\top]^\top \in \mathbb{R}^{kn}. \quad (4.1)$$

If the m -th sensor provides less than k measures, the missing components can be deselected by zeroing the corresponding rows in \mathbf{S}_m . Matrix \mathbf{S} can also be used to weigh/combine outputs from different sensors in a single task. Each task is related to the Cartesian velocity of the TCP $\mathbf{v} \in \mathbb{R}^k$ by the $k \times k$ Jacobian \mathbf{J}_m :

$$\dot{\mathbf{s}}_m = \mathbf{J}_m \mathbf{v}. \quad (4.2)$$

We prove that the optimal controller⁵ ensuring convergence of \mathbf{s} to \mathbf{s}^* is⁶:

$$\mathbf{v} = (\mathbf{S}\mathbf{J})^{-1} (\mathbf{s}^* - \mathbf{s}), \quad \text{with } \mathbf{J} = [\mathbf{J}_1 \dots \mathbf{J}_n]^\top \in \mathbb{R}^{kn} \times \mathbb{R}^k. \quad (4.3)$$

On the other hand, *classic* hybrid control (see Sect. 2.5) consists in assigning each sensor to a Cartesian direction in the operational space and then summing up the

⁵By optimal, here, we mean it minimizes the task error 2-norm.

⁶Assuming matrix $\mathbf{S}\mathbf{J}$ is invertible.

velocities associated with the selected sensors:

$$\mathbf{v} = \mathbf{S} \begin{bmatrix} \mathbf{J}_1^{-1} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \mathbf{J}_n^{-1} \end{bmatrix} (\bar{\mathbf{s}}^* - \bar{\mathbf{s}}). \quad (4.4)$$

Controller (4.4) is optimal under two strong assumptions: all the sensor tasks \mathbf{s}_m must be expressed in the same reference frame and only one sensor can be used to control each direction of \mathbf{v} . These assumptions limit its use in practical applications. For instance, merging image-based visual servoing, which defines the visual task in the image frame with force control usually implemented at the contact point, would infringe the first assumption. On the other hand, (4.3) only requires that $\mathbf{S}\mathbf{J}$ is invertible (a weaker assumption, always true under the two assumptions above).

Furthermore, the use of homotopies (differentiable time-varying expressions for the selection matrices \mathbf{S}_m) smoothens the transitions between sensor tasks, to guarantee safer operation. We show experimentally, in the mentioned collaborative screwing setup, that vision and force tasks can be realized either exclusively or simultaneously with our controller. To this end, we utilize a fixed Kinect and a black and white camera mounted on the robot, so as to respectively track the human hand position (using OpenNI⁷) and newly inserted screws on the flank. To properly align tool and screw, the external wrench on the TCP is estimated via the FRI Interface. The robot infers the - unpredictable - human intentions using only low-cost sensors, without the need for structuring neither the environment nor the operator (in contrast, e.g., with [112]).

4.3.2 Collaborative assembly cell

The second target application (Fig. 4.2(b)) is the assembly of an Rzeppa homokinetic joint. In particular, we focus on the insertion of six steel balls in the joint grooves. This is currently done manually by the PSA operators, using an *insertion tool* and a *gripper* to incline the joint cage and insert the balls. The cage opening should be automated, to alleviate the worker from musculoskeletal disorders, while ball insertion requires very high precision and adaptability skills, not attainable by present-day industrial robots.

To fulfill these requirements, in [27] we propose a novel design of the Rzeppa assembly cell. The lower part of the joint is held by the robot, while the insertion tool is fixed to a rigid support. Hence, most of the required movements are carried out by the robot, with the human intervening only to position the balls.

By relying on force and vision, we successfully manage direct physical contact between robot and human, and between robot and environment. In fact, vision stops robot operation in case of danger for the operator hand (as in the images

⁷<https://github.com/OpenNI/OpenNI><https://github.com/OpenNI/OpenNI>

in Fig. 4.2(b), top right), while the external forces are used (via admittance controller (2.13)) to deform the robot nominal trajectory for collision/blockage avoidance. Referring to Sect. 2.5, our approach can be classified as *shared*, since vision and force operate concurrently. However vision intervenes merely as a trigger, blocking or releasing the robot motion.

Although the applications targeted by most of the works cited in Sect. 4.2 also fall in the *shared workplace* paradigm evoked in [77], they differ from the one treated here, since the robot motion is driven only by the human worker. Instead, in our work the robot alternates active (i.e., autonomous) and passive (compliant) behaviors, to lighten the burden on the operator in the first case and to follow his/her needs in the latter.

Furthermore, in contrast with most similar works (e.g., [96]), our approach can be applied to standard position (and not torque) controlled robots, common in the industry. From the end user's (PSA) viewpoint, two aspects are particularly noteworthy. First, since the operator load is reduced by approximately 60%, the proposed assembly cell can be reclassified in the PSA ergonomics scale. Second, a complete risk analysis by PSA indicates that the proposed setup is compatible with the safety standards and can be certified.

4.3.3 A safe adaptive damping controller

Following that work, we pursued research on manually guided collaboration, on the one hand focusing on the requirements of the ISO10218 safety standard, on the other, exploring the features of direct force measurement via *touch* [29]. This research has been carried out on a hand-arm robotic system, in a mock-up drilling application, shown in Fig. 4.2(c). Here, as for the Rzeppa assembly, the robot operates as an enhanced weight compensator, by alternating active and passive modes. Touch (via tactile sensing) provides an intuitive interface for the operator, enabling it to easily switch between the various modes.

The ISO10218 standard specifies that in presence of a human being, any robot must respect contact force, velocity and power limits at the TCP:

$$|\mathbf{f}| \leq F_M, \quad |\mathbf{v}| \leq V_M, \quad \langle {}^B\mathbf{h}, \dot{\mathbf{x}} \rangle \leq P_M. \quad (4.5)$$

Positive scalars F_M , V_M and P_M are respectively the maximum external force, velocity and transmitted power allowed (all three specified by the end-user). Since in pHRI it is very difficult to model/predict the evolution of exchanged force and wrench over time, we assume the norm of their derivatives to be bounded by known \dot{F}_M and \dot{H}_M respectively. Then, considering the robot as a delayed first order system, we can determine the maximum increment of the force and power norm that could occur as the system responds:

$$\Delta F_M = \frac{3\dot{F}_M}{\lambda}, \quad \Delta H_{M\%} = \frac{3\dot{H}_M}{\lambda|\mathbf{h}|}, \quad (4.6)$$

with $\lambda > 0$ the scalar gain of the inverse kinematic controller (see Sect. 2.2). Under these assumptions, we have designed and experimentally validated an adaptive damping controller that limits online (and only when needed) $|\mathbf{f}|$, $|\mathbf{v}|$ and $\langle {}^B\mathbf{h}, \dot{\mathbf{x}} \rangle$, to fulfill (4.5). We apply (2.13), with $\mathbf{h}^* = \mathbf{M} = \mathbf{K} = \mathbf{0}$, $\mathbf{B} = K^{-1}\mathbf{B}_t$, and \mathbf{B}_t a constant, pre-tuned, damping matrix. Our controller generates the desired velocity $\dot{\mathbf{x}}^*$, by adapting the scalar $K \leq 1$:

$$\dot{\mathbf{x}}^* = K\mathbf{B}_t^{-1}\mathbf{h} + \dot{\mathbf{x}}^r, \text{ with } K = \begin{cases} \min \{K_v, K_p, 1\} & \text{if } |\mathbf{f}| \leq F_M - \Delta F_M \\ \min \{K_v, K_p\} & \text{otherwise.} \end{cases} \quad (4.7)$$

Parameters $K_v < 1$ and $K_p < 1$, which respectively guarantee the velocity and power limitation, are function of the task, configuration, and measured wrench. Note that in all cases K is chosen conservatively to guarantee that neither the velocity (since $K \leq K_v$) nor the power (since $K \leq K_p$) constraints are infringed. Moreover, if the force is below the security threshold $F_M - \Delta F_M$, the damping will stay above its pre-tuned values (because $K \leq 1$), whereas otherwise, it could go below these values and move the robot away from collision.

Although we did not apply our control framework directly to the robot hand, we did guarantee safety of its use by exploiting tactile data in two ways. First, we designed a simple grasp strategy that is driven by tactile measures. In fact, we use them to modify online the desired articular configuration of each finger to regulate the contact pressure between finger and object. Second, we exploit the thumb BioTac as an intuitive interface for the operator. This BioTac is used as a button, to trigger some events (e.g., to start grasping a tool). The implemented switch is based on a comparator with hysteresis. With this system, the operator can interact with the robot without an external sophisticated interface. This solution improves both the ergonomy, and the time required to perform the task.

4.4 Conclusions

This chapter has outlined our main research results in the field of collaborative industrial robotics. The three case studies were application-driven, and have required us to address two crucial specifications from the industry: safety and robustness. We have shown that *vision* and *force* are fundamental to achieve these, and have started exploring the potential of *touch* in pHRI.

On a more theoretical viewpoint, the development of our unified multimodal controller, has opened numerous research avenues. These include: studying its robustness with regards to inaccurate estimation of the task Jacobian matrices \mathbf{J}_m , guaranteeing the boundary conditions at sensor task switches, and considering the use of parsimonious control [116]. The second and third aspect can be addressed with non-linear optimization, a powerful tool, which we has been successfully deployed on humanoid robots, as will be shown in Chapter 5.

Chapter 5

Force and visual control for interaction with humanoid robots

This chapter presents our research on applying **force** and **visual control** on real-size **humanoid robots**. The objective is to make humanoids capable of carrying objects with humans and of operating human-tailored devices. Humanoid platforms are clearly less application-oriented than the industrial manipulators targeted in Chapter 4. Nevertheless, their operation requires solving specific fundamental research problems, typically problems related to whole-body control of highly redundant systems, and to walking stability. These problems become even more challenging when closing the feedback loop on sensed information. In our work we have exploited *force*, and in two of the three case studies, *vision*.

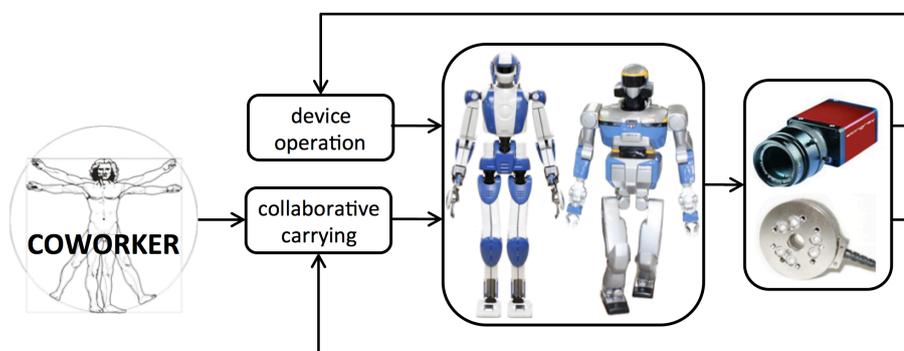


Figure 5.1: Force and visual control of humanoids HRP-2 and HRP-4 (Kawada Ind.), for operating devices and for collaborative carrying with a human co-worker.

The research presented here has been carried out in the framework of the FP7 European Project RoboHow (2012-2016), and of the CNRS PICS Project ViNCI (2013-2016), and has led to three main results:

1. The design of a framework combining vision and force control for operating the steering wheel of a simulated car, enabling to drive the car along a road.

This work has been published in [33], by PhD student **Antonio Paolillo**, in the context of the **ViNCI** Project.

2. The design and validation of a framework combining vision and force control in human-humanoid collaborative carrying. This work has been published in [30–32], in collaboration with the PhD student **Don Joven Agravante**, in the context of the **RoboHow** Project.
3. The design and validation of a humanoid walking pattern generator, accounting for external forces, e.g., those applied by a human in collaborative carrying. This work was also carried out in **RoboHow** by **Don Joven Agravante**, and led to the publication of [34], and to the submission of [35].

The rest of the chapter is structured in the following way: first we present the motivation and objectives (Sect. 5.1), and the current state of art in humanoid robot interaction (Sect. 5.2). Then, we present the contribution of our work in Sect. 5.3, and we conclude in Sect. 5.4.

5.1 Objectives and Motivation

Humanoid robotics focuses on the design of robots directly inspired by human motion and sensing capabilities. This design gives many advantages in operating human-tailored devices and in working in cooperation with humans.

The operation of human-tailored tools by humanoid robots presents clear advantages for society; for instance, in disaster response scenarios, as well as in the domains of service, industrial and entertainment robotics. In disaster situations, for instance, the robot should be capable of driving a utility vehicle to reach the intervention zone, operating cranks or other tools attached to the vehicle [117, 118], and executing other tasks, e.g., turning a valve, drilling, etc. In fact, it is unlikely to have such devices instrumented for autonomous or teleoperated control. If the robot is not a humanoid robot, these operations can hardly be achieved by a unique platform.

All the cited applications require the solution of an unprecedented “humanoid-in-the-loop” problem, since the robot must autonomously control another dynamic system. In fact, it should handle the physical contact with the tool, while controlling its motion. To this end, force and vision provide, once more, the most adapted sensing modalities. Here, we focus on the development of a robot driver, a case study of the more general problem of operating human-tailored devices and machinery (including, e.g., levers, pedals, buttons).

A second, important application of humanoid robotics is collaborative object carrying with human beings in the construction, service and rescue scenarios. In fact, humans have extensive experience in physically cooperating with each another. A humanoid with a similar range of motion and sensing, can provide an intuitive interface, reducing the level of expertise required by the operator.

As shown in Chapter 4, physical human-robot collaboration generally relies on control of the exchanged interaction wrench. Vision can also be fundamental in collaborative carrying, for two main reasons. First, a robot that is able to “see” the user’s non-verbal and non-haptic cues to infer his/her intentions, will be able to interact more intuitively from the human perspective. Second, wrench alone cannot give the robot a complete knowledge of the environment (e.g., a table may have to be moved carefully, to avoid objects from falling off). Both information sources may then be needed to enable a humanoid to perform complicated tasks that are similar to those that humans can perform.

An important challenge, specific to humanoid robots, is biped walk stability. In the case of collaborative carrying, this should take into account the external forces, applied by the human operator. Disregarding the legged aspect, various works have been carried out on wheeled mobile manipulators [119–121]. The trade-off is having an easier control over more limited mobility. Although the locomotion is different, these works reveal one of the main problems: coordinating the motion of the mobile/floating base, with that of the upper body, and with human intentions (i.e., the exchanged force). However, balance is not addressed since, in contrast with humanoids, these robots have a low Center of Mass.

To summarize, the motivation behind the research presented in this chapter is the use of **humanoid robots to operate human-tailored devices and to walk while carrying objects in collaboration** with humans.

5.2 State of art

Since in both applications (device operation and collaborative carrying), robot and environment/human are permanently in contact, within the classification of Sect. 2.5, a *shared* approach is preferable. We hereby review two works that have inspired us: [122] and [123], which rely on vision/force shared control. The control schemes proposed in these two papers are shown respectively at the top and bottom of Fig. 5.2. In [122], the visual loop runs outside the force loop. The reference trajectory $\dot{\mathbf{x}}^r$ output by visual servoing, is deformed in the presence of contact by the admittance controller. Although the formulation relies on the task function approach, the effects of conflicts between force and visual tasks on stability are not discussed. In [123], vision is used to first guide a robot hand towards a grasp position and then, to perform the manipulation by taking into account the contact forces. Here, the admittance controller runs outside the vision loop to deform the reference visual task \mathbf{s}^r . Hence, as is common in external control [124], conflicts between force and position/vision are avoided. The dynamics of the task error however, are not explicitly derived. A stability proof is therefore impossible. Furthermore, the cited works, as well as others using similar approaches such as [125], present experiments on manipulator robots but not on humanoids.

Collaborative carrying has been addressed using small scale humanoids in [126–

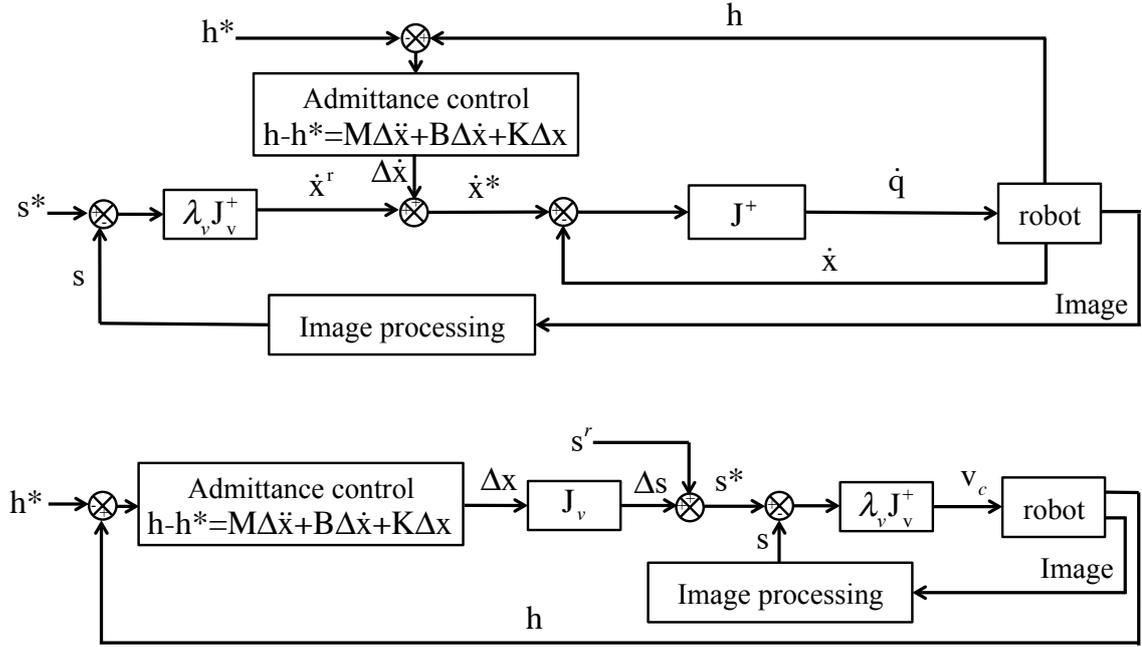


Figure 5.2: Two schemes for *shared* vision-force control. Top: visual loop outside force loop [122] (with \dot{x}^r corresponding to the v_c). Bottom: force loop outside vision loop [123]. In both schemes, $\Delta x = x^* - x^r$, and $\Delta s = s^* - s^r$.

128]. The main focus of [126] was on the use of internal sensors in place of the hand force/torque sensors commonly used in pHRI. The same platform (NAO) is used in [127], where the capture point [129, 130], estimated by the robot IMU, guides the robot walk. Another closely related demonstration can be found in [128], where Darwin robots carry a stretcher. Here however, the human element is removed. In these works, where only robots are used, the interest is turned to synchronization and communication between machines.

Moving to real-size humanoids, early work was done in the Humanoid Robotics Project (HRP), where the HRP-2P humanoid helps a human transport a panel [131]. This type of application has two important aspects: robot and human are jointly doing the same task, and a haptic interaction exists among the two. The research is focused on regulating the interaction forces for safety, and on making the robot proactive in helping humans [5, 6]. Recent advances are presented in [7, 132], where the HRP-2 carries a beam collaboratively with a human. Initially, studies were conducted to understand how a human-human dyad cooperates for such a task [7]. These observations were then exploited to achieve a proactive behavior, enabling the robot to be either a leader or a follower [132]. Both works are achieved using only the force/torque measurements, hence the interest of adding vision, because of the complementary information that it provides.

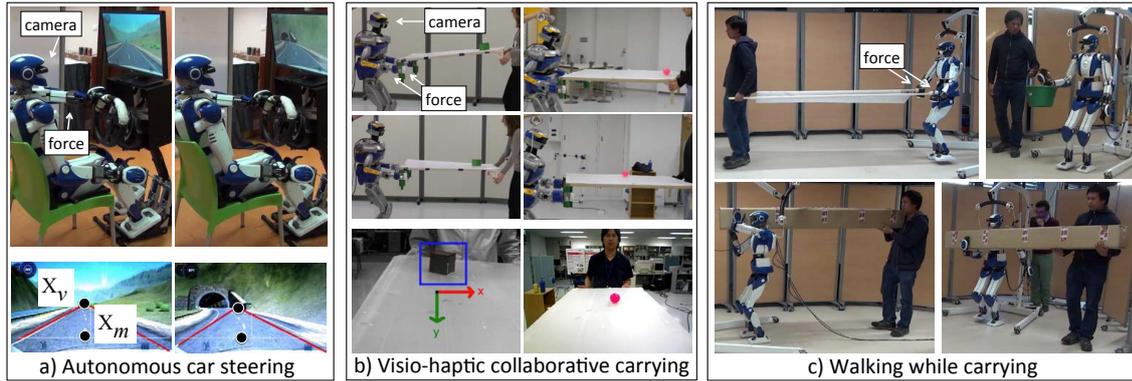


Figure 5.3: Humanoid robot applications. a) Autonomous car steering. Top: experimental setup. Bottom: robot camera views. b) Visual-force collaborative carrying. Top: experiments. Bottom: robot camera views. c) Walking while carrying different objects in different postures.

5.3 Contribution

The research described in this chapter targeted two applications: robot car driving (in the context of the DARPA Robotics Challenge) and human-humanoid collaborative carrying. These applications are shown in Fig. 5.3 and in videos on the IDH LIRMM youtube channel¹. We have utilized both HRP-2 and HRP-4 from Kawada Industries equipped with ATI Mini45 force/torque sensors at each wrist, and with an Asus Xtion RGB-D sensor in the head, which is used as a monocular camera. The contributions have respectively been: a shared vision/force controller enabling a humanoid robot to operate a steering wheel for car driving, a shared vision/force controller for collaborative carrying, and a humanoid walking pattern generator, which accounts for external forces. These are detailed hereby.

5.3.1 Vision and force for autonomous car steering

The first application (shown in Fig. 5.3(a)) consisted in making a humanoid robot operate sophisticated devices. Specifically, we made it turn a car steering wheel and drive the vehicle along a road. Although HRI is not explicit here, since the device is human-tailored, a human-like behavior must be designed.

First, a vision-based controller used two image features of the road to provide the reference angle α for the steering wheel. These features were: the image abscissa of the intersection of the road borders, X_v , and the abscissa of the midpoint of the segment linking the borders along the image plane abscissa axis, X_m . Subsequently, an admittance controller allowed the humanoid to safely rotate the steering wheel

¹https://www.youtube.com/watch?v=WzacruLkN_g
<https://www.youtube.com/watch?v=-1BcC3aEuZM>
<https://www.youtube.com/watch?v=btAws9RDcSA>

with its hand, so as to perform the desired steering command. To obtain such behavior, we draw inspiration from [122] (Fig. 5.2, top), and design a shared vision-force controller with vision in the outer loop. The reason for this choice is that the force sensor frequency is much higher than that of vision.

The main contribution of this work has been a proof of concept, showing the possibility of autonomous operation of complex devices, with a humanoid robot, relying only on standard on-board sensors. With respect to recent works on tool operation [133], a great deal of effort was made to elaborate autonomous behavior and avoid human teleoperation. A second, important contribution has been in the simultaneous control of two sophisticated and interrelated dynamic systems: robot and car. Both kinematic models were taken into account in the design of the framework to guarantee success of the driving task.

We made the following assumptions: that the car driving speed v is always strictly positive and assumed as approximately known, that the robot is already in driving posture with its hand grasping the wheel and that the road is locally flat with feasible turns. Under these assumptions, we considered a rear-wheel kinematic model of the car and showed that the road is followed when the steering wheel is turned by angle:

$$\alpha = \frac{k_\alpha}{k_1 + X_m X_v} \left(k_2 X_v - \lambda_p \frac{X_m}{v} \right), \quad (5.1)$$

with k_α a positive scalar characteristic of the steer, $k_{1,2}$ constants related to the camera parameters, and λ_p a hand-tuned control gain. Angle α was converted to the hand *reference* position required for turning the wheel, \mathbf{x}^r . To soften the contact of the hand with the wheel, we utilized admittance controller (2.13) to yield the desired hand position, \mathbf{x}^* . For this application, our target was to have good tracking of α and safe interaction with the wheel. To this end, the admittance system (applied only on forces, not torques) was designed to be fast and stiff along the wheel tangent, and slow and compliant along the two other axes. Finally, \mathbf{x}^* was sent to the kinematic controller embedded in the stack of tasks framework [114], to move the robot joints accordingly.

The controller was validated (even with respect to inaccurate estimates of v) via a car driving experience by the humanoid robot HRP-4, within a video game setup. This work was preliminary to our research on real car driving (outdoor, and including gas pedal control, [40]). This is presented in Chapter 6.

5.3.2 Collaborative carrying using vision and force

A similar framework can be used to combine vision and wrench in human-robot joint actions, i.e. in collaborative tasks requiring both parties to physically interact. In such scenarios, the robot must move safely, regulating exchanged force and sharing control with a human-in-the-loop, using only its on-board sensors. Vision brings new information that cannot be obtained from force/torque sensors (e.g, perception of object motion and gesture recognition for anticipating human intention).

However, having a human as a physical collaborator implies that some aspects of the shared vision-force framework presented in Sect. 5.3.1 need to be revisited, i.e.: what information would be helpful? how can this information be obtained in the given context? and how/where should this information be used while accounting for the human safety/intention?

We considered the task of jointly carrying a flat surface (e.g., a table) while keeping it horizontal, to prevent any object on top from falling off (see Fig. 5.3(b)). The strategy consisted in using vision to observe/servo the table inclination, while regulating the forces exchanged with the human to follow his/her intention. As in Sect. 5.3.1, we used a *shared* approach, with visual servoing in the outer loop of an admittance controller (see Fig. 5.2, top). The shared approach was preferred over the others, since it allows for compliance in all dof. However, in contrast with Sect. 5.3.1, the framework here was also *hybrid*; controlling some dof only with admittance and adding vision for the remaining ones. In [30], vision was used with force to stabilize only the height of the table. Later, in [32], to avoid a moving object (ball) from falling off the table, a second dof was controlled with vision. Another contribution was the possibility for the robot to be either leader or follower within the same framework.

The approach in [32] consisted in moving the robot's hands (TCP) to drive the table to a reference pose in the robot frame R . This pose, defined (omitting the left superscript R for simplicity) as:

$$\mathbf{x}^r = [x^r, y^r, z^r, \phi_x^r, \phi_y^r, \phi_z^r]^\top, \quad (5.2)$$

was fed to the admittance controller (2.13), to guarantee safety. A fundamental issue of our work was the design of the 6 reference dof. To avoid the ball from falling, visual servoing regulated the table height z^r and the roll angle ϕ_x^r . The three dof characterizing planar displacement of the table (x^r, y^r, ϕ_z^r) were defined (from our group's earlier work [7, 132]), to make the robot either leader or follower. Finally, for human comfort, the table tilt angle was made compliant: $\phi_y^r = 0$. For visual control of z^r and ϕ_x^r , we modeled the ball dynamics using a simple sliding model. Asymptotic convergence of the ball to the table center is guaranteed by a Proportional Derivative feedback controller on the ball position in the table frame, ${}^t x_b, {}^t y_b$:

$$\begin{cases} z^r = -\lambda_1 {}^t x_b - \lambda_2 {}^t \dot{x}_b \\ \phi_x^r = \lambda_3 {}^t y_b + \lambda_4 {}^t \dot{y}_b \end{cases} \quad (5.3)$$

with λ_{1-4} positive scalar gains, ${}^t x_b, {}^t y_b$ derived using the robot on-board Asus Xtion and ${}^t \dot{x}_b$ and ${}^t \dot{y}_b$ obtained by numeric differentiation.

Having now set \mathbf{x}^r , the output of admittance control (2.13) \mathbf{x}^* was then sent to the stack of tasks, so as to control the robot joints. The results on HRP-2 showed that the proposed vision-force controller could be successfully applied to human-robot joint actions.

5.3.3 Walking during physical collaboration

Although in Sect. 5.3.2 we successfully integrated vision and force in a setup where both were necessary, that work highlighted an important limitation of the framework at hand. In fact, postures (e.g., with extended arms or leaning chest) requiring the Center of Mass (CoM) to move away from the waist would make the Walking Pattern Generator (WPG) fail. Indeed, the WPG [134] used there was not designed for physical interaction. Hence, we worked on the design of humanoid walking pattern generators to be used for physical collaboration.

The use case addressed here is that of a humanoid robot helping a human to carry large and/or heavy objects (see Fig. 5.3(c)). In contrast with the two previously described works, here only force measured at the robot wrists is used for control. Our contribution is the design of two WPG for *walking under sustained forces*, one for the leader and the other for the follower robot.

To this end, we constructed a reduced model that takes into account physical interaction. The external wrench (excluding the ground contact forces) expressed in a frame placed on the CoM is noted: $\mathbf{h} = [\mathbf{f}^\top \mathbf{n}]^\top \in \mathbb{R}^6$. The WPG aims at keeping the Zero Moment Point (ZMP) in the support polygon. Assuming horizontal flat ground, constant angular momentum and constant CoM height z_{com} , the dynamic equations yield:

$$\begin{cases} x_{zmp} = x_{com} + \frac{z_{com}f_x + n_y}{mg - f_z} - \frac{z_{com}}{g - f_z/m} \ddot{x}_{com} \\ y_{zmp} = y_{com} + \frac{z_{com}f_x - n_x}{mg - f_z} - \frac{z_{com}}{g - f_z/m} \ddot{y}_{com} \end{cases}, \quad (5.4)$$

with m the robot mass and g the gravity. In absence of the external wrench, this becomes the standard expression found in the literature:

$$\begin{cases} x_{zmp} = x_{com} - \frac{z_{com}}{g} \ddot{x}_{com} \\ y_{zmp} = y_{com} - \frac{z_{com}}{g} \ddot{y}_{com} \end{cases}. \quad (5.5)$$

Applying *model predictive control* to this system, with CoM jerk as input, the future states and outputs can be written in function of vector:

$$\mathbf{p} = \begin{bmatrix} \mathbf{u} \\ \mathbf{r} \end{bmatrix}. \quad (5.6)$$

Vectors \mathbf{u} and \mathbf{r} are the concatenation, over the preview horizon, respectively of CoM jerk components $[\ddot{x}_{com}, \ddot{y}_{com}]^\top$ and of future foot landing positions expressed in the preceding foot frame. Using (5.6), we design two WPG that are suited for *leader* and *follower* modalities. Each is obtained by minimizing an appropriate cost function, using constrained quadratic optimization with \mathbf{p} as argument.

The approach is validated both on simulation and on the HRP-4 humanoid robot. In the follow-up, submitted in [35], the WPG is integrated in a whole-body optimization control framework that also accounts for object grasping, holding, and releasing.

5.4 Conclusions

This chapter has outlined our main research results on sensor-based control of humanoid robots. The case studies targeted both operation of human-tailored devices, and collaborative carrying with a human. The first work has later led to the development of the complete car driving framework that will be described in Chapter 6. The research on collaborative carrying has shown the importance of the human role, for task success (typically, the ball carrying experiment would fail without human collaboration). It should also be noted that these type of applications would greatly profit from an extended sense of touch (e.g., via robot skins).

From a methodological viewpoint, the last work on walking has emphasized the enormous potential of constrained quadratic optimization for whole body kinematic/dynamic control. This approach has in fact gradually replaced the stack of tasks in our research group. One of the main research axes that I plan to follow, will consist in profiting from the experiences gained in controlling such highly redundant systems to provide similar solutions in industrial mobile manipulator control. This will bridge the results presented in this Chapter with those from Chapter 4.

Chapter 6

Shared human-robot control

There is a growing interest in using anthropomorphic robots for many applications, to enrich control and sensing capabilities of human beings. In this chapter, we present our research on applying **vision** and **force** (specifically, **touch**) to extend the senses of a human as s/he controls/embodies robotic systems. This research falls in the broad class of **shared control**. The reader should here note that the term *shared* has a different meaning from how it is used in Sect. 2.5, where it refers to a technique that exploits different senses within a unique control scheme. Instead, in this Chapter, the term indicates that the robot depending on the context can be autonomous, or either partially or entirely controlled by the human operator.

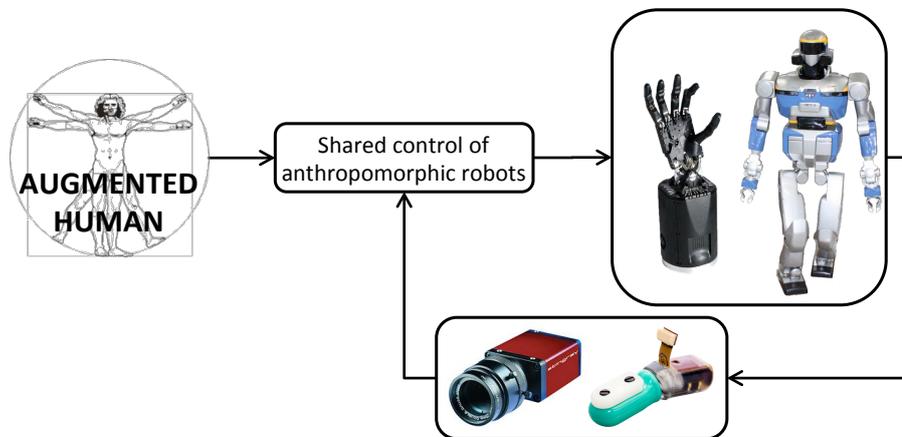


Figure 6.1: Touch and vision-based shared control of humanoid HRP-2 and Shadow Robotic Hand, for “augmenting” human capabilities.

The research presented here has been driven by three application scenarios; two targeting assistance to disabled persons and one for post-disaster intervention. The three main results that were obtained are:

1. The control of a robotic hand with electromyography (EMG) and tactile feedback. This work has been published in the IEEE Trans. on Neural Systems &

Rehabilitation [39], in collaboration with the **INRIA Demar team** (work of PhD students **Wafa Tigra** and **Benjamin Navarro**).

2. The design of a complete visual navigation framework for embodiment in a humanoid robot, via a brain-computer interface (BCI). This work has been published in [36–38] by PhD student **Damien Petit** in the context of the FP7 European Project **VERE**.
3. A vision-based shared control framework for operating a humanoid robot to drive a car in post-disaster scenarios. This work has been submitted to the Journal of Field Robotics [40] by postdoc **Antonio Paolillo**.

The rest of the chapter is organized as follows: first we present the motivation and objectives (Sect. 6.1), and the current state of art in the various research areas (Sect. 6.2). We then present the contribution of our work in Sect. 6.3. Finally, we conclude in Sect. 6.4.

6.1 Objectives and Motivation

Although complete spinal cord injury is devastating, patients can move paralyzed limbs by sending an electric stimulation sufficient to excite the cells responsible for muscle contraction [135]. Nevertheless, precise control of the stimulation device is still problematic, since the choice of voluntary movements (hence, of available media) remains limited. Poor ergonomics of the piloting modes are another cause of the low usage of orthotics and functional electrical stimulation (FES) for restoring hand movements. While avoiding invasive surgery, a dexterous robot hand allows investigation on the use of EMG (which result from the activation of muscles) to control grasping. In fact, its dimensions and dof are close to those of the user hand, allowing him/her to get (via visual feedback) an intuitive representation of the movement that s/he could control with FES-based hand movement restoration.

A similar paradigm can be applied to the whole body of a humanoid robot, to extend not only the control but also the mobility capabilities of elder or physically disabled people. As the world population is aging rapidly, the increase in health care demand may lead to deterioration in the quality of life of senior citizens. This has spurred interest in research on assistive robotics, i.e, on robots aiding in domestic tasks. The human-tailored environment and tools, that are targeted, motivate the choice of humanoids in such scenarios.

The recent DARPA Robotics Challenge (DRC) has shown that disaster scenarios also require humanoids capable of operating tools and machinery originally designed for humans. For these complex tasks, the robot should be aided, at least partially, by a human being, leading to shared control architectures, where (depending on the situation at hand) the system autonomy can be tuned. The DRC utility car driving task is a good illustration of this.

To summarize, the motivation behind the research presented in this chapter is the design of **shared control frameworks** aiding humans when direct perception and/or control are not available. This pushes pHRI to a higher stage, where the robot becomes an **extension of the human being**, enabling him/her to expand his/her capabilities of interacting with the environment.

6.2 State of art

Recently, robotic hands have been used as prosthesis for amputees. These are controlled either by BCI [136] or by EMG [137, 138]. These systems are preferred to conventional grippers, since they allow manipulators or humanoids to handle complex shaped objects originally designed for humans [139]. However, to the best of our knowledge, EMG signals (in contrast with neural signals [140]) have never been used by individuals with tetraplegia (i.e., who are not amputees) to pilot robot hands (in [141], EMG pilots the patient's own hand, via FES). Besides, in most works, a single motor is used to open/close a finger; a design constraint that impedes precise hand grasps.

Electroencephalography-based BCI is preferable for patients with severe motor disability or whenever EMG is unavailable. Given the complexity of decoding motor intention in BCI, researchers generally use high-level intention recognition to perform goal-oriented control [142]. Many assistive applications have thus been developed using BCI [143]. These include control of a wheelchair [144], of a humanoid [145], or of a quadruped robot [146–148]. All works focus either on navigation or on interaction with objects, but never address both issues within the same framework.

Another application where shared control is fundamental is disaster response. In such scenarios where wireless communication may be unavailable, the robot should be capable of some autonomous behavior while still guaranteeing the possibility for manual teleoperation. Again, a layered goal-oriented architecture should be considered. A typical case study is the DRC task that requires a humanoid robot to drive a car at the center of an unknown road at a desired speed. Some researchers have achieved this goal by relying on teleoperation [149, 150], while others have proposed fully autonomous solutions [151, 152]. However, no one has designed shared control frameworks merging the two approaches.

6.3 Contribution

The research described in this chapter addressed shared control of antropomorphic robots. All applications are shown in Fig. 6.2, and in videos on the IDH youtube channel¹. Our contributions have been: the design of an EMG-based robotic grasping interface for individuals with quadriplegia [39], a BCI framework for humanoid

¹<https://www.youtube.com/watch?v=ekP1oxD6Vj0>
<https://www.youtube.com/watch?v=SYHI2JmJ-lk>

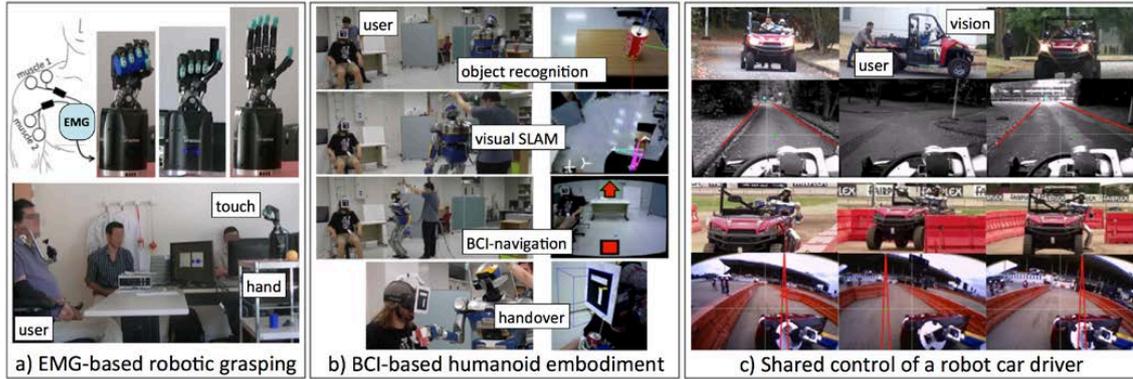


Figure 6.2: Shared control experiments. a) EMG-based robotic grasping. Top: EMG outline, and three grasp states. Bottom: experiments. b) BCI-based humanoid embodiment: four snapshots and robot views of an experiment, including object grasping, carrying, and handover. c) Shared control for humanoid robot car driving: snapshots of the experiments and robot view. Top: experiment alternating autonomous and teleoperated modes. Bottom: assistive mode experiment.

embodiment [36–38] and a shared control scheme for humanoid robot car driving [40]. These are detailed hereby.

6.3.1 EMG-based robot hand grasping

In [39] we investigated the feasibility of processing online the muscle responses of quadriplegic subjects to pilot a robotic hand. A fundamental advantage, is that visual feedback (from observing the hand) can speed up the training phase. The ability to grasp with a Shadow robotic hand equipped with BioTac sensors on each fingertip, was assessed in two male spinal cord injured subjects, at the Propara Neurological Rehabilitation Center, in Montpellier, France (see Fig. 6.2(a)).

A pair of surface-recording electrodes was positioned on each of two chosen muscles. The choice was based on the subject preference, and on the EMG signal quality, which was then low-pass filtered. Subjects did not receive any pre-training, and were only instructed on the contraction movements.

The hand was controlled in five alternative modes, each corresponding to a different finite state machine, with the transitions between states triggered by muscular contractions and relaxations (using hysteresis). The triggering thresholds on the EMG were determined via a calibration phase. Three hand states were used: palmar grasp, palmar pinch, and open hand (left to right in the top of Fig. 6.2(a)). Each state is characterized by the five finger target joint values, \mathbf{q}_i^* ($i = 1, \dots, 5$), and an online trajectory generator² ensures smooth motion to \mathbf{q}_i^* . For the two closing states, the motion of each finger i stops when contact with the object is detected,

²<http://www.reflexxes.ws>

via the norm of the force measured on the corresponding BioTac, $\|\mathbf{f}\|_i$.

Two objects with different sizes were placed near the robot hand, and the subjects had to select the most appropriate muscle contraction (and corresponding hand grasp) for the given object. In one of the five modes, where closure is proportional to the contraction intensity, the patients succeeded in modulating the EMG by using visual feedback to compensate for the lack of haptic sensitivity. This confirms that our setup can be useful for rehabilitation or for training FES control. Another interesting feature is that even a weak muscle can produce a proper EMG signal, hence an effective robotic movement. Finally, *force control* via the BioTacs compensates the lack of tactile feedback and ensures proper object grasping.

6.3.2 BCI-based humanoid embodiment

Being able to control a robotic avatar and to perceive the world through its senses covers issues related to *embodiment*; a topic that we studied within the FP7 European Project VERE. In that context, we designed a framework for assisting a disabled user to control a humanoid robot with a BCI. In contrast with most related works, our framework provides the robot with both navigation and interaction capabilities. The low frequency and accuracy of the BCI commands are compensated by vision algorithms, including object recognition, and simultaneous localization and mapping (SLAM). In the resulting shared control scheme, only high-level goals are specified by the human being, while the robot *vision-based controller* takes care of achieving them.

We use the HRP-2 robot; a robot which is remotely controlled by a user, equipped with a BCI cap, and with a head mounted display (HMD) enhancing immersion (the user sees through the robot “eyes”). The images acquired by the robot are processed online to localize the robot in the environment (using SLAM from [153]) and to recognize/localize relevant objects in the scene. Both the robot camera view, and the recognized object models are projected in the HMD. The objects and navigation commands (directional arrows, or target poses) are displayed in the form of stimuli, flickering at different frequencies and related to tasks to be realized by the robot. When the user focuses his/her attention on a flickering stimulus, a peak is observed in the BCI signal at the corresponding frequency. The user intention, output by the BCI, determines the tasks (navigation or interaction) to be realized by the robot. Generality of this framework is demonstrated in a complex experiment, where the user controls the robot to serve himself a drink (Fig. 6.2(b)).

6.3.3 Shared control for a humanoid robot car driver

As a follow-up to the work presented in Sect. 5.3.1, we have designed a sensor-based framework enabling a humanoid robot to drive a car along a road. To make the approach vehicle-independent, we use only sensors commonly on-board humanoids, i.e. a camera in the head and an Inertial Measurement Unit (IMU) in the chest.

The robot sits in the car with its camera pointing the road. The acquired images and IMU data are fed to two branches of the framework running in parallel: car steering, and velocity control. The first maintains the car at the center of the road, with the approach described in Sect. 5.3.1. The latter makes the car progress at a desired speed: a Kalman Filter fuses optical flow and IMU data to estimate the current car speed, so that the gas pedal is pushed accordingly by the robot's foot.

The framework described above allows autonomous robot car driving. Under certain working conditions, however, this mode may fail. Hence, humans should supervise the driving operation, and control the car if required. To this end, we implemented the three driving modes below.

- *Autonomous*: as indicated above, car steering and velocity control are both autonomous, so that the robot does not need any human aid.
- *Assisted*: the human takes care of *road detection* (by manually selecting the road borders), and *car velocity control* (by teleoperating the robot ankle). The borders are still fed to the robot controller, that autonomously computes the hand command.
- *Teleoperated*: for steering wheel and gas pedal operation, both the robot hand and foot are teleoperated. The reference signals are sent to the robot kinematic controller through a keyboard or joystick. The human being uses the robot camera images as visual feedback for driving.

The human user/supervisor can intervene at any moment during the driving, to select a driving mode, by pushing proper joystick (or keyboard) buttons.

We experimentally validated the framework, with a real car and the humanoid robot HRP-2. In the intermediate phase of the experiment shown at the top of Fig. 6.2(c), the human switches to a teleoperated mode, while the visual features are not considered by the controller anymore. Nevertheless, the transitions are smooth. The bottom of Fig. 6.2(c) shows assisted driving: by clicking, the user selects artificial road borders (red in the figure) while teleoperating the robot ankle.

6.4 Conclusions

In this chapter we presented our main research in shared human-robot control. Here, pHRI is stronger than in the other chapters, since the robot *augments* of the human capabilities. In fact, quadriplegics can grasp via a robotic hand, mobility and interaction are enhanced by a humanoid robot, and a sophisticated task (car driving) can be executed remotely by relying on a humanoid pilot.

A key issue is decoding human intentions, and using this to control the robot. Since the human signal frequency is, generally, lower than that of the robot controller, goal-oriented task architectures appear as the best solution. Nevertheless, their use including tunable levels of autonomy should be transparent for the user. To this end, the sensors can be used to close an inner feedback loop directly on the robot.

Chapter 7

Research perspectives

In spite of the enormous progress in terms of both research and commercialization that robotics has achieved in recent years, much work is still to be done. My short, but now decennial, experience in the field, leads me to believe that there has been a growing gap between the various technologies, particularly in terms of their maturity for transfer (the so called Technology Readiness Level, TRL).

For instance, researchers have solved most of the problems (SLAM, exploration, navigation) related to the control of mobile unmanned vehicles, as testified by the picturesque invasion of such systems in our everyday life. Robotic vacuum cleaners, as well as assisting car driving systems, not to mention drones, and automatic flight and train controllers, are now present everywhere. Even the advent of urban self-driving vehicles seems much closer than it was at the time of my research in that field (Chapter 3). This progress has clearly been accompanied and motivated by the concurrent breakthrough of related technologies: vision (including real-time tracking and structure-from-motion), wireless communication, and speech recognition, just to mention a few. Generally, the digital revolution originally predicted in “Moore’s law”, has made complex algorithms executable in real time and on small, embedded processors.

On the other hand, issues such as energy autonomy and safe operation in the presence of humans, are still far from being solved. The latter problem has motivated all my research, since arriving at LIRMM within the IDH group in 2011 (Chapters 4-6). Since physical Human-Robot Interaction is at the core of IDH’s work, I was required a slight thematic change from my previous research on sensor-based navigation. Five years into this change, I am even more strongly persuaded that studying pHRI is not only indispensable for societal acceptance of robots; it also opens a series of unprecedented research questions. Aside from the delicate issue of ethics, most of these questions have been raised in the previous pages. These concern: safety, human intention recognition, control sharing and role assignment. In the upcoming years, I plan to pursue my research in this direction, contributing to pHRI by exploiting my experience in sensor-based feedback control, on numerous platforms. In the following sections, I outline my planned avenues of research

on advancing *methodologies for sensor-based control*, and on merging the experiences drawn from all the platforms on the novel BAZAR *mobile manipulator*. I will conclude this work with a short overview of *perspective collaborations*.

7.1 Sensor-based control

As explained throughout this manuscript, the integration of different sensor modalities for direct control has been at the core of my recent research. Although some important results have been obtained, I consider this research as still exploratory, since it has opened tracks in various directions, promising to “keep us busy” for the upcoming years.

To get a clear picture of the current state of art, I started, in 2015, a workshop on “*Multimodal sensor-based robot control for HRI and soft manipulation*”. This workshop, organized with Youcef Mezouar, David Navarro-Alarcon, Mario Prats and Juan Antonio Corrales Ramon, took place at IROS in Hamburg, on September 28th, 2015. It included invited talks and a session of posters, selected after a peer review process. Although the workshop focused mainly on vision- and force- based control, the poster session showcased the growing need for tactile and proximity sensing in present-day robotics. An important goal of the workshop was to bridge the gap between communities, by sharing knowledge on signal processing, modeling and task control aspects common to the different sensors. The workshop success (over 50 participants) has motivated the publication of a Special Issue on the “*Robotics and Autonomous Systems*” journal for which I will be in charge as Guest Editor. I plan to pursue my activity in *shaping a community* of researchers interested in these research themes, possibly including the foundation of a dedicated RAS Technical Committee.

The second edition of the workshop at IROS 2016 emphasized the importance of *auditive* sensing. This sense has proved its technological maturity in numerous applications, including speech and voice recognition for telephony, language learning, blind people education, and other functions. However, in spite of its unavoidable role for successful HRI, audio has somewhat been neglected by the robotics research community, probably (and paradoxically) because of its maturity. It is, however, my belief that aside from providing an indisputable interface, the use of audio recognition, in combination with other robot senses, will be of great research interest.

In terms of senses and their correlation, the attentive reader may have noticed that in the manuscript introductory scheme (Fig. 1.1), the only missing link is the one between *distance and force*. In our opinion, this is not merely an aesthetic lack. All applications requiring the robot to gradually come in contact with the human being or environment are prone to need these senses for smoothly transitioning between the free and constrained spaces. Besides control, the need can also come from perception itself. Typically, in our initial works with the BioTacs, we have noticed the measurements non-repeatability which motivates the need for continuous online

calibration. In such scenarios, anticipating the contact via distance measurements, while helpful, requires proper sensor fusion tools. Within the collaborative screwing application presented in Sect. 4.3.1 and Fig. 4.2(a), we have partly explored distance-force coupling. However, in that work, distance was implicitly derived via vision and the environment model. Future research would consist in using explicit distance measurements in similar applications.

As outlined above, another important observation drawn from exchanges, including those at the workshop, concerns the gap in terms of used methodologies between the various “sensor communities”. Although lidar-based mapping for instance is now a consolidated technology, it still relies mainly on point representation (extended, for low spatial resolution, to occupancy grids) and scan matching. Shape recognition/tracking tools from image processing are generally ignored, despite recent advances in terms of both robustness and speed. A similar risk could be encountered in the more recent tactile technology, where taxels (primitive point elements) are for the moment the only universally used model. As the resolution of tactile sensors increases, the use of more sophisticated *geometric primitives* is advisable. Similarly, *active exploration*, which is a solved problem using lidar and vision has been investigated little, when it comes to tact. This is the topic of the thesis of **Zineb Abderrahmane**, that I supervise with André Crosnier and Ganesh Gowrishankar. Lastly, the lidar-based teach-and-repeat scheme presented in Sect. 3.3.3 is a first step towards applying the “*perception to action*” control paradigm, to a sensor other than vision. In other words, we advocate the extension of visual servoing techniques to the other senses, for instance, by treating tactile skin data, as an image.

On similar control aspects, much progress has been done within the stack-of-tasks framework of Nicolas Mansard et al., starting from the seminal work [114]. This framework proved effective for controlling highly redundant robots, while accounting for dynamics and unilateral constraints [154]. Although control with heterogeneous sensors was experimentally validated, theoretical studies on the *robustness* with respect to inaccurate estimation of the sensor Jacobians (the “interaction matrix” in visual servoing) were not explicitly addressed in the stack-of-tasks. Another fundamental open problem that I plan to address, and that has been partially studied in [154–156], concerns the stability of the *transitions* between different sensor tasks.

The latter issue is crucial, in all applications where contact is generated during robot motion. Passing from free to constrained space, the control signals must remain smooth while relying on different senses (vision or distance prior to, and force posterior to, contact). An appropriate tool for guaranteeing such continuity at the sensor task switches is *constrained optimization*, where suitable boundary conditions can be set, for this purpose. This idea, which I plan to explore in the near future, is in line with the importance that optimization has recently gained, in robotics research, including within IDH. Apart from quadratic optimization applied for whole-body control in Sect. 5.3.3, linear optimization has been studied by my colleagues Philippe Fraise and André Crosnier to design parsimonious controllers [116].

All these elements shape an ideal context for applying optimization to multimodal sensor-based control on platforms such as the one that will be presented in the next section.

7.2 The mobile manipulator platform BAZAR

The second research axis that I plan to investigate consists in concentrating the experiences gained in controlling the various platforms cited above on a unique system. This motivation emerged within the **CoBot@LR** project which was financed by the Languedoc-Roussillon region in the context of a young researchers grant that I obtained in 2013. The goal of CoBot@LR is to design *mobile manipulation* solutions for collaborative industrial robotics, i.e., enhancing the mobility of systems such as those proposed in Chapter 4. This objective has led to the design and development, in collaboration with other LIRMM colleagues (in particular, **Robin Passama** and **Benjamin Navarro**) of our Bimanual Agile Zany Anthropomorphic Robot, **BAZAR** (sketched on the right of Fig. 7.1).

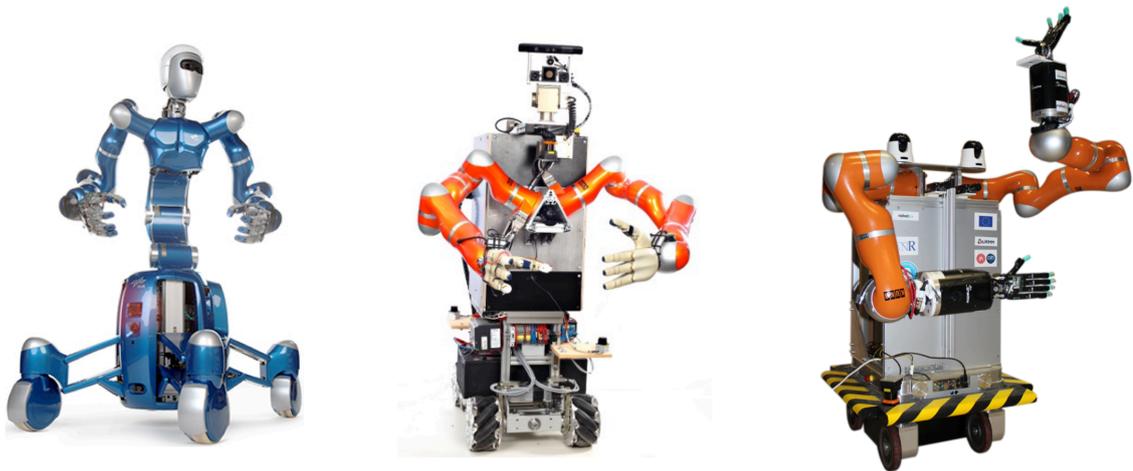


Figure 7.1: Mobile manipulators. Left to right: Justin, TUM-Rosie and BAZAR.

The irony of the name, Persian for “marketplace”, lies in its meaning in the French urban dictionary, a “complicated grouping of different elements”. Indeed, in contrast with most existing mobile manipulators, which are completely designed by a single manufacturer, BAZAR combines the last generation sensors and actuators, from various brands within a unique platform. These are: two laser scanners (Hokuyo UTM-30LX) and four cameras (two D-Link DCS-5222L and two AVT GT1920C) on its base (Neobotix MPO700), a force/torque sensor (ATI Mini45), hand (Shadow Hand with BioTac) and an in-palm 3D range sensor (Leap Motion camera) on each arm (Kuka LWR). Although its design resembles that of Justin and TUM-Rosie (also shown in Fig. 7.1), the base of BAZAR can carry higher payloads thanks to its omnidirectional, steerable wheels.

However, a fundamental issue of *steerable wheels* with regards to purely omnidirectional (e.g., swedish) ones is their control when the desired center of instantaneous rotation moves unpredictably. This systematically occurs in sensor-based control, as the robot behavior is not preplanned and will be driven by unexpected events (e.g., obstacles, or interacting humans). Steerable wheel control for BAZAR is currently being studied by **Mohamed Sorour** in the context of his PhD thesis, that I am co-supervising with Philippe Fraisee and PSA. The preliminary results are very promising [157], and we are confident that this fundamental research problem will be soon solved by Mohamed. With reactive velocity control in the operational space (i.e., about any instantaneous center of rotation), it will then be possible to *extend our results on visual navigation and obstacle avoidance* (Chapter 3) to all steerable systems.

Concerning obstacle avoidance, the tentacle approach presented in Sect. 3.3.2, must be adapted from non-holonomic to steerable omnidirectional wheeled robot kinematics. Besides its theoretical contribution, such generalization would provide a powerful solution for obstacle avoidance with any wheeled robot. This research is being carried out along with PhD student **Abdellah Khelloufi**, in the context of a collaboration with the CDTA Center in Algeria.

In general, BAZAR will be an ideal platform for merging and extending the research results presented throughout this manuscript. Firstly, it can *link all the senses* (force, vision, and distance) outlined in Fig. 1.1 in the context of HRI. Second, it can be used for *whole-body control*, just as the humanoids of Chapter 5, to simultaneously realize multiple tasks, including manipulation, and self collision avoidance, or in *teleoperation setups* to remotely strengthen a human operator as in the works presented in Chapter 6.

Last but not least, we plan to use BAZAR to explore the concurrent use of vision and force for *dual manipulation of flexible objects*. Automatic manipulation of soft materials is a fundamental - yet open - research problem in robotics, despite its numerous applications (e.g., in the food industry). For this, we will profit from the previous IDH works on dual arm control [158], and from our research on merging vision and force for manipulation (Chapter 4). We also envision collaborating with David Navarro from the University of Hong Kong, who has recently applied visual servoing to soft object manipulation [159]. This, and other envisioned collaborations, are summarized in the following Section.

7.3 Envisioned collaborations

To achieve such challenging research objectives alone is clearly utopian and presumptuous. Through my work, I have established a small network which I plan to extend in the coming years to involve important robotics actors worldwide.

In a European context, the recently accepted EU H2020 Project VERSATILE (2017-2020), for which I am Principal Investigator at LIRMM, will provide an ideal

industry-oriented benchmark for BAZAR and all related research. My other main contacts are obviously in Italy and France. Following my PhD, I have maintained a strong link with the Robotics Group of “*La Sapienza*” in Rome through the projects **CNRS PICS VINCI** (2013-2016) and **PHC Galileo SenbHibot** (2013). In Rome, I am also collaborating [160] with the *Fondazione Santa Lucia*, a Center of Excellence in human motion understanding and rehabilitation. In the near future, I plan to work with two other fundamental actors of the Italian Robotics scene: the *IIT* (Italian Institute of Technology, in Genoa), and the *PRISMA Lab* (Università Federico II, in Naples). Their Internationally recognized skills (typically, in dynamic control and deformable object manipulation) are complementary to those of IDH. In France, the participation to three **ANR** Projects (**CityVIP**, **ICARO**, and **SIS-Cob**) has made me encounter the main robotics research actors, both in the academy, and industry (*PSA*, now partner in **CoBot@LR**, and *Airbus*). In the academy, along with the worldwide leading group in visual servoing, *Lagadic* (Rennes) where I spent my postdoc, I foresee synergies with my colleagues at *SIGMA* (Clermont Ferrand), Youcef Mezouar and Juan Corrales-Ramon, with whom I co-organized the IROS 2015 Workshop. Furthermore, I am eager to intensify the collaboration with *LAAS* in Toulouse, with whom we share common interests in pHRI and humanoid robotics (as well as the same Administrative Region, since 2016). In Germany, the collaboration originated with the *Universities of Freiburg* and *Bonn* in the context of the PHC Procope ViNavHuBot project should naturally lead, on a short term, to EU Project proposals. In Spain, I plan to intensify our collaboration with the *Tecnalia Foundation*, and with humanoid manufacturer, *PAL Robotics*.

On a wider scale, my current contacts are mainly in Asia, especially in Japan. Along with the *CNRS-AIST JRL* in Tsukuba, which is an important partner of IDH, these contacts include Prof. Ozawa at the *Ritsumeikan University* for vision-based grasping, and Prof. Ramirez Alpizar at the *University of Osaka*, for deformable object manipulation. As mentioned above, on this theme, I am also seeking a framework for collaborating with David Navarro from the *Chinese University of Hong Kong*.

Appendices

Appendix A

Activity report

A.1 Supervision

Throughout my career, I have supervised the research of eight PhD students (five ongoing), three postdocs and nine 2nd master students. These supervisions are presented below.

A.1.1 PhD students

2011-2015	Antonio Paolillo , supervised with A. De Luca (30%) and M. Vendittelli (40%).
	<i>Vision-based control of humanoid robots interacting with the real world.</i>
	Defended on March 16, 2015, at Università di Roma “La Sapienza”.
	Thesis Jury Maria Domenica Di Benedetto, Reviewer, Giovanni Ulivi, Reviewer, Costanzo Manes, Reviewer, Alessandro De Luca, Principal Supervisor, Marilena Vendittelli, Co-Supervisor.
	Funding : Italian Government Scholarship.
	Joint work presented in Sect. 5.3.1, published in [33].
	Current situation: postdoc at LIRMM (Supervisor: A. Kheddar).

Antonio’s thesis exploits information from an onboard camera to improve the autonomous behavior of humanoid robots. In particular, he focused on 3 challenging

topics: (i) *localization*, (ii) *navigation* and (iii) *interaction* with the environment.

The proposed *localization* method has the prediction-correction structure of an Extended Kalman Filter. Kinematic computations give a prediction of the robot pose, which is then corrected by measurements of the robot torso orientation and camera pose. The *navigation* algorithm uses two visual features extracted from the guidelines at the intersections of walls and floor, to make the robot walk at the center of corridors. The same vision-based navigation strategy was used for *interacting* with the environment (i.e., driving a simulated car with the humanoid HRP-4 [33]). This last part of the thesis (presented in Sect. 5.3.1) was developed during Antonio’s **six month stay at LIRMM**, in the context of our collaboration (**CNRS PICS ViNCI** project).

2012-2015	Don Joven Agravante , supervised with A. Kheddar (30%).
	<i>Human-humanoid collaborative object transportation.</i>
	Defended on December 16, 2015, at Université de Montpellier.
	Thesis Jury Gabriel Abba, Reviewer, François Chaumette, Examiner, Florent Lamiraux, Reviewer, Pierre-Brice Wieber, Examiner, Abderrahmane Kheddar, Principal Supervisor, Andrea Cherubini, Co-Supervisor.
	Funding : EU FP7 RoboHow.
	Joint work presented in Sections 5.3.2 and 5.3.3, and published in [30], [31], [32], [34] and submitted in [35].
	Current situation: postdoc at INRIA Rennes (Supervisor: F. Chaumette).

Joven’s thesis focused on enabling collaborative humanoids. Humanoid robots provide many advantages when working together with humans to perform various tasks. This is because humans have an extensive experience in physically collaborating with each other. In particular, Joven focused on the example of collaboratively carrying and transporting objects together.

He first focused on utilizing *vision and force* for enabling better collaboration (work presented in Sect. 5.3.2 and published in [30]- [32]). He then designed *walking pattern generators* that take into account physical collaboration. Finally, he broke down the task of collaboratively carrying an object together with a human and implemented it within a *whole-body* control optimization framework. The design of the walking pattern generators and of the whole-body collaborative carrying framework have been presented in Sect. 5.3.3, published in [34] and were submitted in [35]).

2012-2015	Damien Petit , supervised with A. Kheddar (30%) and L. Aymerich-Franch (30%).
	<i>Analysis of sensory requirement and control framework for whole body embodiment of a humanoid robot for interaction with the environment and self.</i>
	Defended on December 14, 2015, at Université de Montpellier.
	Thesis Jury Gérard Bailly, Reviewer, Laurence Devillers, Reviewer, Peter F. Dominey, Examiner, Abderrahmane Kheddar, Principal Supervisor, Laura Aymerich-Franch, Co-Supervisor, Andrea Cherubini, Co-Supervisor.
	Funding : EU FP7 VERE.
	Joint work presented in Sect. 6.3.2 and published in [36]- [38].
	Current situation: postdoc at Osaka University.

Damien’s thesis targeted the development of a system, allowing a human participant to *embody* a robotic entity. The feeling of embodiment goes beyond teleoperation, since the user feels as if s/he is the surrogate. The thesis studied this feeling in various scenarios, reproducing assistive domestic situations. Particular interest was taken in controlling the robot via a brain computer interface (BCI), while remaining still at a remote location. The following challenges were addressed:

1. *Surrogate aspect.* Androids replicate a person aspect but lack the ability to move. In contrast, since humanoids have this ability, embodiment is assessed in experiments where they mirror the user movements.
2. *Embodiment through walking.* In telerobotics, there is an inherent latency between the operator command and the robot task realization. Damien designed a high-level navigation framework to study how latency affected the embodiment of the subject while s/he drives the robot.
3. *Interaction with oneself.* Most embodiment studies use a fake limb or mannequin as surrogate. What would be the feeling experienced if the surrogate could actually interact with the user? To answer this question, Damien extended his framework to make the robotic surrogate hand objects to the user and manipulate his/her own arm.

The development of the navigation/interaction framework (points 2 and 3) has been presented in Sect. 6.3.2, and published in [36]- [38].

2014-2017	Benjamin Navarro , supervised with G. Poisson (30%), P. Fraisse (20%), and A. Fonte (20%).
	<i>Modeling and control of a compliant robot, designed for physical human-robot interaction.</i>
	Expected defense in December, 2017, at the Université d'Orléans.
	Funding : ANR SISCob.
	Joint work presented in Sections 4.3.3 and 6.3.1, published in [29], and submitted in [39].

Benjamin's thesis focuses on developing control schemes for safe physical human-robot interaction. These should target both classic and intrinsically safe manipulators, embedding the *Safe Intelligent Sensor* designed within the ANR SISCOB. In his first PhD year, Benjamin has developed an original admittance controller that maintains the exchanged force, velocity and power below the bounds imposed by the ISO218 safety standard (see Sect. 4.3.3 and [29]). This system was deployed on the robot-arm system present at LIRMM. Benjamin extended the robot hand controller to EMG-based operation (work presented in Sect. 6.3.1, and submitted in [39]). In the rest of his thesis, Benjamin will generalize his controllers to SIS-equipped robots.

2014-2017	Mohamed Sorour , supervised with P. Fraisse (30%).
	<i>Control of a dual arm mobile manipulator.</i>
	Expected defense: December, 2017, Université de Montpellier.
	Funding : Languedoc-Roussillon Regional Project CoBot@LR (50%) and PSA Peugeot Citroën (50%).
	Joint work published in [157].

Mohamed's thesis is motivated by an industrial specification of our partner PSA. The goal is to design a control framework, enabling a dual arm mobile manipulator, to operate autonomously in a factory setting. The robot should displace car parts from/to various areas of the factory. To this end, Mohamed will use the BAZAR robot, presented in Sect. 7.2 and composed of Kuka LWR arms and Shadow Hands mounted on a Neobotix MPO700 platform. This platform presents numerous kinematic singularities which Mohamed successfully tackled in [157]. Another drawback with respect to standard omnidirectional platforms is that it can turn around an arbitrary center of rotation only after reconfiguring its wheels. This problem is being currently studied by Mohamed, who will then address the problem of upper body (i.e., dual arm) control.

2015-2018	Zineb Abderrahmane , supervised with A. Crosnier (30%) and G. Gowrishankar (30%).
	<i>Visio-haptic exploration for modeling and classifying unknown objects.</i>
	Expected defense: December, 2018, Université de Montpellier.
	Funding : Algerian Government Averroès Scholarship.

Humans have the amazing ability to recognize hand held objects haptically. In her Phd, Zineb aims at providing similar capabilities to robots. She works with a robotic hand-arm system. The joint readings from the robot hand and the multimodal sensors mounted on the five fingers provide information on the grasped object as well as on physical properties like compliance, surface texture and thermal conductivity. Currently, Zineb is exploring different machine learning algorithms to extract representative object features from the multimodal data and build a descriptor. She should then develop an active exploration procedure that improves the grasps, so as to provide more informative features.

Two other recently starting thesis are:

2016-2019	Sonny Tarbouriech , supervised with P. Fraisse (50%).
	<i>Dual arm kinematic control.</i>
	Funding : French Government CIFRE (with Tecnalía Foundation).
2016-2019	Osama Mazhar , with A. Crosnier (30%) and S. Ramdani (30%).
	<i>Human intention recognition for robot control.</i>
	Funding : University of Montpellier.

A.1.2 Postdocs

In the context of the various projects in which I was involved, at Université de Montpellier, I have supervised the work of three postdocs:

- **Arnaud Méline**, “Development of vision-based control algorithms for a robotic manipulator”, 01/02/2013-30/11/2013, funding : ANR ICARO, supervisors : A. Cherubini 40%, A. Crosnier 30%, P. Fraisse 30%. Arnaud developed image processing algorithms for detecting screws in a flank, using a moving camera.

The position of the screw in the image is exploited to drive the multimodal controller, presented in Sect. 4.3.1 and in [28].

- **José Romero Velazquez**, “Design of a whole-body humanoid state observer to detect contacts”, 01/11/2014-31/5/2015, funding : PSPC ROMEO 2, supervisors : A. Cherubini 40%, A. Crosnier 30%, A. Kheddar 30%.
- **Antonio Paolillo**, “Humanoid robot interaction with sophisticated machinery”, 01/07/2015 - 31/3/2016, funding : PSPC ROMEO 2, supervisors : A. Cherubini 50%, A. Kheddar 50%. During this postdoc, Antonio continued working on humanoid car driving. In particular, he extended his thesis work, by accounting for shared control with the human user, and by making the humanoid actuate the car gas pedal, using feedback from optical flow. He also validated his approach in real outdoor experiments. His work has been presented in Sect. 6.3.3, and submitted in [40].

A.1.3 Master students

Throughout my career, I have supervised the internships of nine 2nd year Master students that are all listed below. The supervisions of Bachelor, and 1st year Master student internships on the other hand, are presented with my teaching activities, in Sect. A.3.1.

During my PhD, at the **University of Rome “La Sapienza”**, I have supervised the internships of four students:

- Sandro Bruscinò (2005): “ASPICE: an assistive robotic system”, supervised with G. Oriolo. This work has contributed to the publication of [161].
- Theodora Capeqi (2006): “Vision for robot navigation”, supervised with L. Iocchi and D. Nardi.
- Marcello Lombardo (2008): “Humanoid robot walk control”, supervised with G. Oriolo. This work has contributed to the publication of [162].
- Manuel Colafrancesco (2008): “Visual navigation”, supervised with G. Oriolo. This work, presented in Sect. 3.3.1, has been published in [18].

As Associate Professor, at the **Université de Montpellier**, I have also supervised the internships of five students:

- Fares Mentseur (2012): “Kinematic modeling and control of a 7 degrees of freedom robot manipulator”.
- Boris Gretchanichenko (2013): “Dynaic identification of the manipulator arm Kuka LWR”, supervised with A. Crosnier.

- Camilo Mousselli (2014): “Modeling and control of a robotic hand”, supervised with A. Crosnier.
- Xavier Gorrion (2015): “EMG control of a robotic hand”, supervised with C. Azevedo. This work, presented in Sect. 6.3.1, has contributed to [39].
- Florian Raffalli (2015): “3D reconstruction from a pair of stereo cameras”, supervised with R. Passama.

It is worth mentioning that, in the Italian Academic system, the final Master internship is much longer than in the French one. An Italian internship generally lasts nine to ten months, compared to the four to five months demanded in France.

A.2 Scientific activities

A.2.1 Project Management

At LIRMM, I am *Principal Investigator* for the following research projects:

- 2013: **PHC Galileo SenbHibot** (3000 €). This project targeted sensor-based human-robot interaction in collaboration with the University of Rome “La Sapienza”. It covered travel expenses for seminars and visits.
- 2013-2016: **CNRS PICS ViNCI** (12000 €) This project addresses vision-based humanoid navigation and interaction, in collaboration with University of Rome “La Sapienza”. The project funded the visit of **Antonio Paolillo** at LIRMM (2014) and contributed to his research results (these have been presented in Sections 5.3.1 and 6.3.3).
- 2014-2017: **French ANR SISCob** (110000 €). This project addresses the design of a safe intelligent sensor to be embedded on robots to make them intrinsically compliant. Along with LIRMM (responsible for the development of safe control laws for pHRI), the project involves PPRIME (Coordinator), PRISME, XLIM et SENSIX. The project has contributed to fund the PhD of **Benjamin Navarro** (co-supervised with PRISME), whose research results have been presented in Sections 4.3.3 and 6.3.1).
- 2014-2017: **Languedoc-Roussillon Project CoBot@LR** (70000 €). This project aims at designing a framework for controlling a dual arm mobile manipulator. The target application proposed by our partner PSA Peugeot Citroën is industrial kitting. The project has contributed to fund the PhD of **Mohamed Sorour** whose research has been published in [157]. An expected result will be the design and control of the BAZAR platform, presented in Sect. 7.2.

- 2015-2016: **PHC Procope ViNavHuBot** (5000 €). This project, inspired by the preliminary results presented in Sect. 3.3.3, addresses visual navigation with mobile and humanoid robots. The partners are the German Universities of Bonn and Freiburg.
- 2017-2020: **EU H2020 VERSATILE** (300000 €). This project, coordinated by the TECNALIA Foundation (Spain) aims at developing flexible yet robust robot solutions for the “factory of the future”.

A.2.2 Involvement in other projects

During my career, I have also been involved in the following projects:

- 2005-2007: **Italian Telethon ASPICE**: development of an assistive robotic system, including BCI control. Partnership between the University of Rome “La Sapienza” and the Fondazione Santa Lucia, Rome (coordinator).
- 2008-2011: **French ANR CityVIP**: autonomous driving of public transportation vehicles in urban environments. Coordinator: Université Blaise Pascal, Clermont Ferrand. This project has funded my **postdoc at INRIA Rennes** and enabled the results presented in Sections 3.3.1 and 3.3.2.
- 2010-2015: **EU FP7 VERE**: human-humanoid embodiment via BCI. Coordinator: Universitat de Barcelona, Spain. Part of the results (work of **Damien Petit**) has been presented in Sect. 6.3.2.
- 2011-2014: **French ANR ICARO**: development of a collaborative industrial robot. Coordinator: AIRBUS. Part of the results has been presented in Sections 4.3.1 and 4.3.2.
- 2012-2016: **EU FP7 RoboHow**: use of web data for robot daily tasks execution. Coordinator : Universität Bremen, Germany. Part of the results (work of **Don Joven Agravante**) has been presented in Sections 5.3.2 and 5.3.3.
- 2013-2016 : **French PSPC ROMEO 2**: development of a humanoid robot for daily life assistance. Coordinator: Aldebaran.

A.2.3 Editorial activities

I have reviewed over 70 articles, submitted to the International Conferences:

- IEEE Int. Conf. on Robotics and Automation, ICRA (2007 - 2017),
- IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS (2006 - 2016),
- IEEE-RAS Int. Conf. on Humanoid Robots, ICHR (2013-2016),
- Robotics: Science and Systems, RSS (2014),

- American Control Conference, ACC (2014),
- Int. Conf. on Advanced Robotics, ICAR (2009, 2015),

and to the Academic Journals:

- IEEE Trans. Robotics (2009 - 2015),
- Int. Journal of Robotics Research (2011),
- Robotics and Autonomous Systems (2006 - 2013),
- Journal of Field Robotics (2014),
- Autonomous Robots (2013),
- IEEE Trans. on Industrial Informatics (2016),
- IEEE Trans. on Systems, Man, and Cybernetics, Part B (2011),
- IEEE Robotics and Automation Magazine (2013),
- IEEE Trans. on Intelligent Transportation Systems (2013),
- IEEE Trans. on Control Systems Technology (2007),
- Journal of Intelligent and Robotic Systems (2010),
- Robotica (2011),
- Advanced Robotics (2015).

A.3 Teaching activities

A.3.1 Teaching

Starting from my Ph.D. studies, I have performed various teaching activities at **Università di Roma “La Sapienza”**, and at the **Institut National des Sciences Appliquées, INSA Rennes** (during my postdoc). In the rest of this section, acronyms CM, TD, and TP respectively refer to: lectures (“Cours Magistraux”), exercise sessions (“Travaux Dirigés”), and practical work (“Travaux Pratiques”).

Here is a summary of my teaching activity prior to recruitment at Université de Montpellier, as Associate Professor:

- 2006-2008: lectures (20 h CM) in *Adaptive control and Machine learning* for 2nd year Master students, at University of Rome “La Sapienza”.
- 2006-2008: supervision of the *Projects* of seven 3rd year Bachelor students, at University of Rome “La Sapienza”.
- 2009-2010: practicals in *Linear Control Systems* (48 h TP) for 3rd year Bachelor students, at INSA, Rennes.
- 2010: practicals in *Robotics* (36 h TP), for 1st year Master students at INSA, Rennes.

Since my arrival in the Department EEA¹ at the Faculté des Sciences, of the **Université de Montpellier**, in September 2011, I have taught the following:

- 2011 - 2014: practicals (81 h TP) and exercises (26 h TD) in *Linear Control Systems*, for 3rd year Bachelor students.
- 2011 - currently: lectures (29 h CM) and practicals (96 h TP) in *Multivariable Control Systems*, for 1st year Master students.
- 2011 - currently: lectures (43 h CM) and practicals (147 h TP) in *Vision and Robot Perception*, for 2nd year Master students.
- 2011 - currently: supervision of the *Projects* and *Internships* of an annual average of eight (1st or 2nd year) Master students.
- 2012 - 2015: lectures (45 h CM) in *Basic Information and Communications Technology (ICT) Skills*, for 2nd year Bachelor students.
- 2013 - currently: lectures (37 h CM) and practicals (68 h TP) in *Foundations of Robotics and Image Processing*, for 1st year Master students.
- 2015 - currently: lectures (12 h CM) in *Optimisation*, for 2nd year Master students.

Apart from the lectures in *Basic ICT Skills* (which were followed by all the specializations at Université de Montpellier), my teaching activities targeted students specializing in Robotics (at Master level) or in EEA (at Bachelor level).

A.3.2 Responsibilities

- 2012-2015: I have been **Teaching responsible and Manager** of the course on *Basic ICT Skills*, that is followed by all 2nd year Bachelor students of the Faculté des Sciences, Université de Montpellier (i.e., approximately 1000 students per year). My work consisted in: hiring the supervisors (approximately 30 per year) for the practicals, preparing the course timetable, booking the computer classrooms, organizing the final multiple choice test, and grading all students.
- 2015-currently: I am **Teaching responsible and Manager** of the *Robotics Specialization (1st and 2nd Master year)*, at the Université de Montpellier (approximately 50 students per year). For both 1st and 2nd year Master students, my tasks include: selecting the student applications, hiring the supervisors for the practicals, and preparing the financial demands to the Department. For the 2nd year Master students, I must also prepare the timetable, and organize the jury and project/internships defenses.

¹Electric, Electronic, and Automation Engineering.

A.4 Dissemination of knowledge

A.4.1 List of publications

All publications are available online at: www.lirmm.fr/~cherubini

Academic Journals

- [J12] A. Paolillo, P. Gergondet, A. Cherubini, M. Vendittelli, A. Kheddar, **Autonomous car driving by a humanoid robot**, *Journal of Field Robotics* (under revision).
- [J11] W. Tigra, B. Navarro, A. Cherubini, X. Gorron, A. Gelis, C. Fattal, D. Guiraud, C. Azevedo Coste, **A novel EMG interface for individuals with quadriplegia to pilot robot hand grasping**, *IEEE Transactions on Neural Systems and Rehabilitation* (to appear).
- [J10] A. Cherubini, R. Passama, A. Crosnier, A. Lasnier, P. Fraisse, **Collaborative manufacturing with physical human-robot interaction**, *Robotics and Computer Integrated Manufacturing*, vol. 40, August 2016, Pages 1–13.
- [J9] M. Iosa, G. Morone, A. Cherubini, S. Paolucci, **The three laws of Neurorobotics: a review on what neurorehabilitation robots should do for patients and clinicians**, *Journal of Medical and Biological Engineering*, First Online, February 2016.
- [J8] A. Cherubini, R. Passama, P. Fraisse, A. Crosnier, **A unified multimodal control framework for human-robot interaction**, *Robotics and Autonomous Systems*, 70, pp. 106-115, 2015.
- [J7] A. Cherubini, F. Spindler, F. Chaumette, **Autonomous Visual navigation and Laser-based moving obstacle avoidance**, *IEEE Trans. on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2101 - 2110, 2014.
- [J6] A. Cherubini, F. Chaumette, **Visual Navigation of a Mobile Robot with Laser-based Collision Avoidance**, *International Journal of Robotics Research*, vol. 32, no. 2, pp. 189-209, 2013.
- [J5] A. Cherubini, F. Chaumette, G. Oriolo, **Visual servoing for path reaching with nonholonomic robots**, *Robotica*, vol. 29, no. 7, pp. 1037-1048, 2011.
- [J4] A. Cherubini, F. Giannone, L. Iocchi, D. Nardi, P. F. Palamara, **Policy gradient learning for quadruped soccer robots**, *Robotics and Autonomous Systems, Special Issue on Advances in Autonomous Robots for Service and Entertainment*, vol. 58, no. 7, pp. 872-878, 2010.
- [J3] A. Cherubini, F. Giannone, L. Iocchi, M. Lombardo, G. Oriolo, **Policy gradient learning for a humanoid soccer robot**, *Robotics and Autonomous Systems*, vol. 57, no. 8, pp. 808-818, 2009.
- [J2] A. Cherubini, G. Oriolo, F. Macri, F. Aloise, F. Cincotti, D. Mattia, **A multimode navigation system for an assistive robotics project**, *Autonomous Robots*, vol. 25, no. 4, pp. 383-404, 2008.

[J1] F. Cincotti, D. Mattia, F. Aloise, S. Bufalari, G. Schalk, G. Oriolo, A. Cherubini, M. G. Marciari, F. Babiloni, **Non-invasive brain-computer interface system: towards its application as assistive technology**, *Brain Research Bulletin*, vol. 75, no. 6, pp. 796-803, 2008.

International conferences

- [C34] M. Ferro, A. Paolillo, A. Cherubini, M. Vendittelli, **Omnidirectional humanoid navigation in cluttered environments based on optical flow information**, in *16th IEEE-RAS Int. Conf. on Humanoid Robots, Humanoids*, 2016.
- [C33] M. Sorour, A. Cherubini, R. Passama, P. Fraise, **Kinematic modeling and singularity treatment of steerable wheeled mobile robots with joint acceleration limits**, *IEEE Int. Conf. on Robotics and Automation, ICRA* 2016.
- [C32] B. Navarro, A. Cherubini, A. Fonte, R. Passama, G. Poisson, P. Fraise, **An ISO10218-compliant adaptive damping controller for safe Physical HRI**, *IEEE Int. Conf. on Robotics and Automation, ICRA* 2016.
- [C31] D. J. Agravante, A. Sherikov, P-B. Wieber, A. Cherubini, A. Kheddar, **Walking pattern generators designed for physical collaboration**, *IEEE Int. Conf. on Robotics and Automation, ICRA* 2016.
- [C30] D. Petit, P. Gergondet, A. Cherubini, A. Kheddar, **An integrated framework for humanoid embodiment with a BCI**, in *IEEE Int. Conf. on Robotics and Automation, ICRA* 2015.
- [C29] D. Petit, P. Gergondet, A. Cherubini, M. Meilland, A. I. Comport, A. Kheddar, **Navigation assistance for a BCI-controlled humanoid robot**, in *IEEE 4th Annual Int. Conf. on Cyber Technology in Automation, Control, and Intelligent Systems, CYBER*, 2014.
- [C28] A. Paolillo, A. Cherubini, F. Keith, A. Kheddar, M. Vendittelli, **Toward Autonomous Car Driving by a Humanoid Robot: A Sensor-Based Framework**, in *14th IEEE-RAS Int. Conf. on Humanoid Robots, Humanoids*, 2014.
- [C27] P. Gergondet, D. Petit, M. Meilland, A. Kheddar, A. I. Comport, A. Cherubini, **Combining 3D SLAM and Visual Tracking to Reach and Retrieve Objects in Daily-Life Indoor Environments**, in *11th Int. Conf. on Ubiquitous Robots and Ambient Intelligence, URAI*, 2014.
- [C26] V. Bonnet, N. Sylla, A. Cherubini, A. Gonzales, C. Azevedo Coste, P. Fraise, G. Venture, **Toward an affordable and user-friendly visual motion capture system**, in *36th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society, EMBC*, 2014.
- [C25] D. J. Agravante, A. Cherubini, A. Bussy, P. Gergondet, A. Kheddar, **Collaborative Human-Humanoid Carrying Using Vision and Haptic Sensing**, *IEEE Int. Conf. on Robotics and Automation, ICRA*, 2014.
- [C24] D. J. Agravante, A. Cherubini, A. Kheddar, **Using vision and haptic sensing for human-humanoid haptic joint actions**, in *6th IEEE International Conference on Robotics, Automation and Mechatronics, RAM*, 2013.

- [C23] C. Sprunk, G. D. Tipaldi, A. Cherubini, W. Burgard, **Lidar-Based Teach-and-Repeat of Mobile Robot Trajectories**, in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS*, 2013.
- [C22] D. J. Agravante, A. Cherubini, A. Bussy, A. Kheddar, **Human-Humanoid Joint Haptic Table Carrying Task with Height Stabilization Using Vision**, in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS*, 2013.
- [C21] A. Cherubini, R. Passama, A. Meline, A. Crosnier, P. Fraisse, **Multimodal Control for Human-Robot Cooperation**, in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS*, JTCF Novel Technology Paper Award for Amusement culture finalist, 2013.
- [C20] A. Cherubini, B. Grechanichenko, F. Spindler, F. Chaumette, **Avoiding Moving Obstacles during Visual Navigation**, in *IEEE Int. Conf. on Robotics and Automation, ICRA 2013*, Pages 3054-3059, Karlsruhe, Germany, May 2013.
- [C19] A. Cherubini, F. Spindler, F. Chaumette, **A New Tentacles-based Technique for Avoiding Obstacles during Visual Navigation**, *IEEE Int. Conf. on Robotics and Automation ICRA*, 2012.
- [C18] A. Cherubini, F. Chaumette, **Visual navigation with obstacle avoidance**, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS*, 2011.
- [C17] A. Cherubini, F. Spindler, F. Chaumette, **A redundancy-based approach for visual navigation with collision avoidance**, *IEEE Symposium on Computational Intelligence in Vehicles and Transportation Systems CIVTS*, 2011.
- [C16] A. Cherubini, F. Chaumette, **A redundancy-based approach for obstacle avoidance in mobile robot navigation**, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS*, 2010.
- [C15] A. Cherubini, F. Chaumette, **Visual navigation with a time-independent varying reference**, *IEEE/RSJ IROS*, 2009.
- [C14] R. Tatsambon Fomena, H. U. Yoon, A. Cherubini, F. Chaumette, S. Hutchinson, **Coarsely calibrated visual servoing of a mobile robot using a catadioptric vision system**, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS*, 2009.
- [C13] A. Cherubini, M. Colafrancesco, G. Oriolo, L. Freda, F. Chaumette, **Comparing appearance-based controllers for nonholonomic navigation from a visual memory**, *ICRA Workshop on safe navigation in open and dynamic environments: application to autonomous vehicles*, 2009.
- [C12] A. Cherubini, M. Leonetti, L. Marchetti, A. De Luca, L. Iocchi, D. Nardi, G. Oriolo, M. Vendittelli, **S.P.Q.R. Team Description paper 2008**, *RoboCup Proceedings, Standard Platform League - NAO Division*, 2008.
- [C11] A. Cherubini, F. Chaumette, G. Oriolo, **An image-based visual servoing scheme for following paths with nonholonomic mobile robots**, *IEEE Int. Conf. on Control, Automation, Robotics and Vision ICARCV*, 2008.
- [C10] A. Cherubini, F. Chaumette, G. Oriolo, **A position-based visual servoing scheme for following paths with nonholonomic mobile robots**, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS*, 2008.

- [C9] A. Cherubini, F. Giannone, L. Iocchi, P.F. Palamara, **An extended policy gradient algorithm for robot task learning**, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS*, 2007.
- [C8] F. Cincotti, F. Aloise, S. Bufalari, G. Schalk, G. Oriolo, A. Cherubini, F. Davide, F. Babiloni, M. G. Marciani, D. Mattia, **Non invasive Brain-Computer Interface system to operate assistive devices**, *29th IEEE Int. Conf. of the Engineering in Medicine and Biology Society*, 2007.
- [C7] A. Cherubini, F. Giannone, L. Iocchi, **Layered learning for a soccer legged robot helped with a 3D simulator**, *11th Int. Robocup Symposium*, 2007.
- [C6] A. Cherubini, G. Oriolo, F. Macri, F. Aloise, F. Cincotti, D. Mattia, **Development of a multimode navigation system for an assistive robotics project**, *IEEE Int. Conf. on Robotics and Automation ICRA*, 2007.
- [C5] A. Cherubini, G. Oriolo, F. Macri, F. Aloise, F. Cincotti, D. Mattia, **A vision-based path planner/follower for an assistive robotics project**, *1st Int. Workshop on Robot Vision, VISAPP*, 2007.
- [C4] F. Aloise, F. Cincotti, F. Babiloni, M. G. Marciani, D. Morelli, S. Paolucci, G. Oriolo, A. Cherubini, F. Sciarra, F. Mangiola, A. Melpignano, F. Davide, D. Mattia, **ASPICE: an interface system for independent life**, *Int. Conf. on Smart Homes and Health Telematica*, 2006.
- [C3] F. Aloise, F. Cincotti, F. Babiloni, M. G. Marciani, D. Morelli, S. Paolucci, G. Oriolo, A. Cherubini, F. Sciarra, F. Mangiola, A. Melpignano, F. Davide, D. Mattia, **The ASPICE Project inclusive design for the motor disabled**, *Int. Working Conference on Advanced Visual Interfaces*, 2006.
- [C2] F. Cincotti, F. Aloise, F. Babiloni, M. G. Marciani, D. Morelli, S. Paolucci, G. Oriolo, A. Cherubini, S. Brusino, F. Sciarra, F. Mangiola, A. Melpignano, F. Davide, D. Mattia, **Brain-Operated Assistive Devices: the ASPICE Project**, *IEEE/RAS-EMBS Int. Conf. on Biomedical Robotics and Biomechatronics*, 2006.
- [C1] L. Iocchi, D. Nardi, A. Cherubini, L. Marchetti, V.A. Ziparo, **S.P.Q.R. + Sicilia Team Description paper**, *RoboCup Proceedings, 4-Legged League*, 2005.

Other publications

- [M3] B. Navarro, P. Kumar, A. Fonte, P. Fraisse, G. Poisson, A. Cherubini, **Active calibration of tactile sensors mounted on a robotic hand**, in *IROS Workshop on Multimodal sensor-based robot control for HRI and soft manipulation*, 2015.
- [M2] A. Cherubini, R. Passama, A. Meline, A. Crosnier, P. Fraisse, **Sensor-based control of a collaborative robot**, in *6th International Workshop on Human-Friendly Robotics, HFR*, 2013.
- [M1] D. J. Agravante, A. Cherubini, A. Kheddar, **Visio-haptic control for Human-Humanoid Cooperative Carrying Tasks**, in *6th International Workshop on Human-Friendly Robotics, HFR* 2013.
- [T1] A. Cherubini, **Vision-based techniques for following paths with mobile robots**, D.I.S., University of Rome “La Sapienza”, April 2008.

A.4.2 Invited seminars

During my career, I have been invited to present my work in the following occasions:

- 2007 : “*The ASPICE project*”, Automation Institute(IAT), Universität Bremen, Germany.
- 2007 : “*Multimodal navigation in ASPICE*”, Royal Institute of Technology, Stockholm, Sweden.
- 2009 : “*Visual navigation with obstacle avoidance*”, Université de Technologie de Compiègne, France.
- 2010 : “*Outdoor visual navigation with obstacle avoidance*”, VisuLab - Università di Parma, Italy.
- 2012 : “*Outdoor Visual Navigation*”, Albert-Ludwigs-Universität Freiburg, Germany.
- 2012 : “*Vision-based control of mobile robots*”, CNRS-AIST Joint Robotics Laboratory, Tsukuba, Japan.
- 2012 : *At the crossroads of neurorehabilitation and robotics*, Fondazione Santa Lucia, Rome, Italy.
- 2013 : “*An overview of our research on Human-inspired Cybernetic Systems*”, MPI Tübingen, Germany.
- 2014 : “*See and touch: Visio-haptic control for safe human-robot interaction*”, ETH Zurich, Switzerland.
- 2014 : “*Visio-haptic control for safe human-robot interaction*”, IEEE ICRA Workshop “Affective Haptics for Human-Robot Interaction”, Hong-Kong, China.
- 2015 : “*Visio-haptic control for safe physical interaction with humanoids*”, plenary at the French Humanoid Research Conference, JNRH, Nantes, France.
- 2016 : “*Sensor-based control of humanoids*”, University of Bonn, Germany.
- 2016 : “*Visio-haptic robot control*”, Scuola Superiore Sant’Anna, Pisa, Italy.
- 2016 : “*Visio-haptic control for physical human-robot interaction - recent works of the LIRMM IDH Group*”, German-French Conference on Humanoid and Legged Robots, HLR 2016, Toulouse, France.

A.4.3 Invitations to PhD defenses

I have been invited as *Reviewer* to the following PhD defenses:

- **Deon George Sabatta**

ETH Zurich, Switzerland, December 5, 2014.

Algorithms for vision-based path following along previously taught paths

Jury: **R. Siegwart** (Principal Supervisor), L. Van Gool (Co-Supervisor), A. Cherubini and C. Pradalier (Reviewers).

Deon extended two established path following controllers to teach-and-repeat visual navigation. The first one relies on a scaled distance to path, derived

from the visual memory. For the second, a novel image-based cost function is proposed, to apply model predictive control to visual navigation.

- **Daniel Maier**

Albert-Ludwigs-Universitaet Freiburg, Germany, April 10, 2015.

Camera-based humanoid robot navigation

Jury: **M. Bennewitz** (Principal Supervisor), W. Burgard (Examiner), B. Nebel (Examiner), A. Cherubini (Reviewer).

This thesis focused on cameras as primary sensors for humanoid robot deployment. Daniel addressed the following problems: self-calibration of the robot kinematic model, safe navigation based on self-supervised image classification, visual localization and mapping, and manipulation by object pose tracking.

- **Alessandro Di Fava**

Scuola Superiore Sant’Anna, Pisa, Italy, November 2, 2016.

Multisensory Advanced Robot Control: from perceptual sensors to human demonstrations.

Jury: **E. Ruffaldi** (Principal Supervisor), C. A. Avizzano (Examiner), A. Kheddar and A. Cherubini (Reviewer).

This thesis investigates technological and algorithmic solutions to enable development of what the recent Strategic Research Agenda calls Better Action and Awareness. The main goal is to improve the robot autonomy, by understanding and interpreting the environment surrounding the robot.

As *Co-Supervisor*, I have taken part to the PhD defenses of **Damien Petit** on December, 14, 2015 and of **Don Joven Agravante** on December, 16, 2015. The details of these defenses have been given in Sect. A.1.1.

A.4.4 Participation in Program Committees

- I have been invited to serve as **Associated Editor** at the IEEE/RSJ IROS Conference, in 2014, 2015 and 2016. The mission consists in several tasks. First, each paper must be assigned to 2-3 reviewers. The reviews must then be summarized in a report, giving a final decision (either acceptance or refusal). I have been assigned 6 papers in 2014 (on Planning and Optimization), and respectively 11 and 8 in 2015 and 2016 (on Perception and Localization).
- I have been the **founder and main organizer** of the Workshops on *Multi-modal sensor-based robot control for Human-Robot Interaction and Soft Manipulation* at IROS in 2015 and 2016. The work consisted in several tasks. First, I found four colleagues to help in the organization. Second, we submitted a proposal to the IROS Workshop Committee, including the website that I designed and managed. We then contacted six experts for invited talks, and managed the reviewing process of the submitted papers. Finally, we organized

the workshop on site. The success of the 2015 Edition (over 60 participants), has led to acceptance of our proposals for 2016 Edition and to a related *Robotics and Autonomous Systems* journal *Special Issue* for which I will be **Editor in Chief**.

- I have been among the six **organizers** of the Workshop *Tactile sensing for manipulation: new progress and challenges* at the IEEE-RAS Int. Conf. on Humanoid Robots, Humanoids 2016. The work consisted in submitting a proposal to the ICHR Workshop Committee, in contacting eleven experts for invited talks, and in reviewing the submitted papers.

A.4.5 National and International Expertise

- 2014 : expert for the **European Program CHIST ERA**. The work consisted in reviewing a 60 page project on Robotics.
- 2014 : expert for the **Italian National Young Researchers Program SIR** (Theme: Robotics). The work consisted in reviewing first 6 short pre-proposals (5 pages each), and then 2 preselected proposals (15 pages each).
- 2014: invited to the **Selection Committee for an Associate Professor position at University of Cergy Pontoise, France** (profile: Robotics and Cognitive Science). The work consisted in preselecting the best among 17 applications, attending the oral presentations of the 5 preselected candidates and selecting one in agreement with the other Committee members.
- 2015: expert for the **French National ANR ACHN Program** for hosting high level foreign researchers. The work consisted in reviewing a 25 page project on Image Processing.
- 2016: expert for the **Italian National Research Program PRIN** (Theme: Robotics). The work consisted in reviewing three projects of 30 pages each.

A.4.6 Miscellaneous

Awards

In 2013 I was awarded, with 18 other young researchers, the prestigious prize “*Chercheurs d’Avenir*” from the Region Languedoc-Roussillon. This award funded my project CoBot@LR (70000 €, see Sect. A.2.1).

Media dissemination

Reports about my work in ASPICE have appeared on the Italian public TV *RAI* and on Italian newspaper “*Il sole 24 ore*”. In 2007 and 2010, I was interviewed by the French magazine “*Sciences Ouest*”. In 2013, I was interviewed during the TV

show “*Campus LR*”, and for the magazine “*La Gazette de Montpellier*”. Work on humanoid robot driving has been reported by Spanish newspaper “*El país*”.

Conference chair

I have been invited to chair the session “*Non-holonomic motion planning*” at IROS 2008 and the session “*Visual servoing*” at ICRA 2013.

Appendix B

Résumé en français

B.1 Mémoire

B.1.1 Activités de recherche (Chapitres 1-6)

Après la soutenance de ma thèse (2008), j'ai poursuivi mon activité de recherche dans le domaine de la robotique. Ce manuscrit résume les travaux effectués durant mon postdoctorat à l'INRIA Rennes (2008-2011), et ensuite, en tant que Maître de Conférence au LIRMM/Université de Montpellier (2011-présent).

Ma recherche, au cours de ces huit années, a visé à développer des lois de commande référencées capteurs, ciblant l'interaction physique entre hommes et robots. Nous avons conçu et développé des nouvelles méthodes pour exploiter les informations provenant de divers capteurs (de vision, force, et distance) pour que le robot puisse collaborer de manière efficace et sûre avec l'opérateur humain.

Puisque le robot et l'homme risquent, à l'avenir, d'évoluer dans les mêmes lieux, le premier doit pouvoir identifier le second et déterminer sa position, pour effectuer des mouvements d'évitement ou d'interaction. D'autres verrous importants concernent la sécurité de l'homme, qui doit rester prioritaire, et la simplicité des interfaces de commande. Il est en effet fondamental, tout utilisateur, même novice, puisse contrôler de manière intuitive le robot.

En vue de ces trois verrous, il est clair que le système de perception de robot compagnon, est essentiel pour traiter des interactions réciproques. Parmi les diverses modalités sensorielles, nous avons montré l'importance fondamentale de vision, force et distance.

Les travaux présentés dans le manuscrit sont regroupés suivant quatre axes principaux (un par chapitre):

- **Navigation visuelle autonome, en présence d'obstacles.** Ce travail a consisté à développer un système de navigation, permettant au robot de rejouer de manière autonome un parcours appris, en utilisant uniquement des amers visuels naturels. Une première contribution a consisté à étudier plusieurs modèles de l'interaction robot-camera, afin de choisir les informations visuelles les

plus adéquates à la commande. Dans notre système de navigation, le chemin est représenté de manière topologique, par une série d'images clé à atteindre. Les transitions d'une image clé à la suivante étant initialement assez brusques, nous avons proposé l'introduction d'une référence variable dans la loi de commande, qui a amélioré les performances du système. Par la suite, nous avons attaqué le problème de l'évitement d'obstacles, pendant la navigation, en utilisant, en plus de la vision, les données provenant d'un capteur de distance (lidar). L'intégration de vision et distance s'est faite de manière compacte, dans un contrôleur *hybride* unique, permettant d'éviter aussi les obstacles mobiles.

- **Commande multimodale pour la collaboration avec des robots manipulateurs industriels.** Une forte demande de technologies permettant l'interaction physique entre homme et robot, vient du monde de l'industrie. Aujourd'hui, dans les usines de production, les postes d'assemblage sont soit manuels, soit automatiques, et des barrières de protection séparent systématiquement les robots des opérateurs humains. Cependant, l'évolution récente des technologies et des normes laisse envisager un nouveau scénario, où homme et robot pourront travailler côte à côte. Dans cette optique, les humains réaliseront les opérations complexes, pendant que les robots effectueront les activités facilement automatisables, peu ergonomiques ou dangereuses, le tout dans une zone de travail partagée. Nous avons exploité la complémentarité de vision et force, dans trois applications industrielles. Le premier travail a permis de généraliser les résultats obtenus pour la navigation, en concevant un contrôleur hybride, permettant de gérer plusieurs capteurs pour l'interaction. Dans le deuxième travail, nous avons conçu une cellule manufacturière, où le robot alterne phases actives et phases passives, pendant une opération d'assemblage collaboratif. Enfin, nous avons conçu une loi de commande en admittance sûre par rapport aux contraintes de la norme ISO10218.
- **Robots humanoïdes pour la collaboration et la manipulation d'outils complexes.** Les mêmes approches peuvent être utilisées pour contrôler des robot humanoïdes. Il s'agit là de plateformes qui, de part leur ressemblance à l'homme, peuvent l'aider dans des tâches collaboratives, tel le transport, et dans la manipulation de machines ou outils conçus pour l'usage humain, mais inaccessibles (par exemple, suite à une catastrophe naturelle). La perception multimodale, permet en effet de lever certaines indéterminations de ces tâches. Typiquement, nous avons montré la complémentarité de la vision, par rapport à l'utilisation de la force seule, lors d'une tâche de transport collaboratif de charges lourdes. Ceci a donné suite à un deuxième travail, où le générateur de marche du robot humanoïde a été adapté, pour prendre en compte les forces échangées avec l'homme pendant le transport. Le troisième travail a concerné la conception d'un schéma de commande permettant à un robot humanoïde de conduire une voiture, en exploitant données visuelles et haptiques.

- **Retour capteurs dans des schémas de téléopération robotique.** La dernière application ciblée a été l'utilisation des robots pour étendre les capacités sensorimotrices d'humains (par exemple, handicapés). A ce fin, différents signaux physiologiques peuvent être utilisés, tels l'électromyographie, ou encore l'électroencéphalographie. Nous avons utilisé ces deux types de signaux, pour permettre à un individu d'incarner respectivement une main robotique, et un robot humanoïde. Une troisième cas d'étude a été la conception d'une architecture de commande pour un robot conducteur, permettant différents niveaux d'autonomie. Dans tous ces travaux, l'importance de la perception du robot est primaire, car elle permet de fermer des boucles de commande à haute fréquence, qui ne seraient pas contrôlables directement par l'opérateur humain.

Tous ces travaux ont inspiré mon projet de recherche, qui est résumé ci-dessous.

B.1.2 Projet de recherche (Chapitre 7)

Dans les années à venir, j'envisage de continuer à développer des approches de perception hybride, agissant directement au niveau de la commande et non pas, comme dans la fusion traditionnelle, au niveau de l'estimation d'état. Je poursuivrai principalement deux axes de recherche: d'un côté sur les méthodologies le plus appropriés pour la commande basée capteurs, de l'autre en développant des nouvelles approches, sur un système complet, le manipulateur mobile BAZAR présent au LIRMM.

Concernant les méthodes, plusieurs développements sont à envisager dans le cadre de ma recherche. D'abord, suite à mon implication personnelle dans la communauté IEEE RAS (de part l'organisation de workshops sur la commande multimodale, à IROS 2015 et 2016), je souhaite étendre mon domaine d'étude à d'autres modes sens, tel l'ouïe. Un deuxième axe concerne l'intégration de distance et force, notamment dans les transitions entre espace libre et contraint (contact). Toujours dans le cadre des méthodes, nous avons perçu un manque d'échange entre les communautés travaillant sur les différents capteurs. Il serait intéressant de combler ce manque, par exemple, en appliquant les méthodes de traitement d'images aux données tactiles des peaux artificielles. Un dernier aspect concerne l'utilisation des techniques d'optimisation, qui sont de plus en plus populaires dans la communauté, y compris dans notre équipe, et dont pourrait profiter aussi la commande multimodale.

Le deuxième volet de mon projet concerne le développement de schémas de commande multimodale pour l'interaction physique, sur le robot manipulateur mobile BAZAR du LIRMM. Ce robot comprendra capteurs et actionneurs très avancés, tous embarqués sur une base mobile pseudo-omnidirectionnelle. De part ses caractéristiques, BAZAR permettra de réunir les différents résultats de nos travaux sur une seule plateforme. Un des verrous principaux concerne la commande de ce système. S'agissant d'une base pseudo omnidirectionnelle, les méthodes de navigation visuelle

avec évitement d’obstacles nécessiteront d’une adaptation. En effet, les roues orientables de ce robot ne permettent pas les changements instantanés du centre de rotation, ce qui rendra nécessaire la conception de loi de commandes innovantes. Un dernier aspect concernera la commande duale pour la manipulation d’objets flexibles, possible grâce aux deux bras manipulateurs embarqués sur BAZAR. Il s’agit là d’un sujet très innovateur, mais pour lequel, de part l’expertise présente au sein de l’équipe, nous sommes optimistes.

B.2 Rapport d’activités

B.2.1 Encadrement

Thèses

- **Antonio Paolillo**, “Vision-based control of humanoid robots interacting with the real world”, Università di Roma “La Sapienza”, 2011-2015, financement : Gouvernement Italien, encadrants : A. De Luca 30%, M. Vendittelli 40%, A. Cherubini 30%. Son travail est présenté en Sect. 5.3.1.
- **Don Joven Agravante**, “Transport collaboratif homme-humanoïde”, Université de Montpellier, 2012-2015, financement : EU FP7 RoboHow, encadrants : A. Kheddar 30%, A. Cherubini 70%. Son travail est présenté en Sections 5.3.2 et 5.3.3.
- **Damien Petit**, “Commande de robot humanoïde avec interface neuronale”, Université de Montpellier, 2012-2015, financement : EU FP7 VERE, encadrants : A. Kheddar 30%, A. Cherubini 40%, L. Aymerich-Franch 30%. Son travail est présenté en Sect. 6.3.2.
- **Benjamin Navarro**, “Modélisation et commande d’un robot compliant. Application à l’interaction humain-robot”, Université d’Orléans, 2014-2017, financement : ANR SISCob, encadrants : G. Poisson 30% A. Cherubini 30%, P. Fraise 20%, A. Fonte 20%. Son travail est présenté en Sections 4.3.3 et 6.3.1.
- **Mohammed Sorour**, “Contribution à la commande d’un manipulateur mobile à 2 bras”, Université de Montpellier, 2014-2017, financement : Bourse Région LR 50%, PSA 50%, encadrants : P. Fraise 30%, A. Cherubini 70%.
- **Zineb Abderrahmane**, “Exploration visio-haptique pour la modélisation et la classification d’objets de formes inconnues”, Université de Montpellier, 2015-2018, financement : Bourse Averroès (Gouvernement Algérien), encadrants : A. Crosnier 30%, A. Cherubini 40%, G. Gowrishankar 30%.
- **Sonny Tarbouriech**, “Manipulation robotique à deux bras”, Université de Montpellier, 2016-2019, financement : CIFRE (Gouvernement Français), encadrants : P. Fraise 50%, A. Cherubini 50%.
- **Osama Mazhar**, “Reconnaissance des intentions humaines pour la commande robotique”, Université de Montpellier, 2016-2019, financement : Université de

Montpellier, encadrants : A. Crosnier 30%, A. Cherubini 40%, S. Ramdani 30%.

Post-doctorants

- Arnaud Méline, “Développement d’algorithmes de commande visuelle pour un robot manipulateur”, 01/02/2013-30/11/2013, financement : ANR ICARO, encadrants : A. Cherubini 40%, A. Crosnier 30%, P. Fraisse 30%. Travail présenté en Sect. 4.3.1.
- José Romero Velazquez, “Design of a whole-body humanoid state observer to detect contacts”, 01/11/2014-31/5/2015, financement : PSPC ROMEO 2, encadrants : A. Cherubini 40%, A. Crosnier 30%, A. Kheddar 30%.
- Antonio Paolillo, “Humanoid robot interaction with sophisticated machinery”, 01/07/2015 - 31/3/2016, financement : PSPC ROMEO 2, encadrants : A. Cherubini 50%, A. Kheddar 50%. Travail présenté en Sect. 6.3.3.

Masters

À l’Université de Rome “La Sapienza”:

- Sandro Brusco (2005) : “Un système robotisé pour l’assistance”,
- Theodora Capeqi (2006) : “Vision pour la navigation robotique”,
- Marcello Lombardo (2008) : “Contrôle d’un robot bipède”,
- Manuel Colafrancesco (2008) : “Navigation visuelle robotique”.

À l’Université de Montpellier:

- Fares Mentseur (2012) : “Modélisation cinématique d’un bras manipulateur”,
- Boris Gretchanichenko (2013) : “Identification dynamique du bras Kuka LWR”,
- Camilo Mousselli (2014) : “Modélisation et commande d’une main robotique”,
- Xavier Gorron (2015) : “Commande par EMG d’une main robotique”,
- Florian Raffalli (2015) : “Reconstruction 3D à partir de caméras stéréo”.

Devenir des doctorants

- Antonio Paolillo (thèse mars 2015) : Post-doctorant au LIRMM, Montpellier (Superviseur: A. Kheddar).
- Don Joven Agravante (thèse décembre 2015) : Post-doctorant à l’INRIA Rennes (Superviseur: F. Chaumette).
- Damien Petit (thèse décembre 2015) : Post-doctorant à l’Université d’Osaka.

B.2.2 Responsabilités scientifiques

Responsabilité de projets

- 2013 : PHC Galileo SenbHibot. Responsable scientifique pour le LIRMM. Ce projet concernait l'interaction homme-robot, en collaboration avec l'Université "La Sapienza" de Rome.
- 2013-2016 : ViNCI (Programme CNRS PICS). Responsable scientifique pour le LIRMM. Ce projet concerne l'interaction visio-haptique avec un robot humanoïde, en collaboration avec l'Université "La Sapienza" de Rome. Une partie des résultats a été présentée en Sections 5.3.1 et 6.3.3.
- 2014-2017 : ANR SISCob (Programme "Sciences et technologies de production, l'usine numérique"). Responsable scientifique pour le LIRMM (workpackage: loi de commandes sûres pour l'interaction physique homme-robot, dont les résultats présentés en Sect.4.3.3). Ce projet concerne la conception d'un capteur intelligent pour la robotique, et comprends 4 autres partenaires: PPRIME (Coordinateur), PRISME, XLIM et SENSIX.
- 2014-2017 : CoBot@LR (Programme "Chercheurs d'Avenir", financé par la Région Languedoc-Roussillon). Responsable scientifique pour le LIRMM (commande visio-haptique d'un manipulateur mobile à deux bras). Ce projet concerne la robotique collaborative en milieu industriel, en partenariat avec PSA.
- 2015-2016 : PHC Procope ViNavHuBot. Responsable scientifique pour le LIRMM. Ce projet, inspiré par une première collaboration, qui a abouti aux travaux présentés en Sect. 3.3.3, concerne la navigation visuelle de robots mobiles (humanoïdes ou sur roues). Partenaires : Universités de Bonn, et de Freiburg, Allemagne.
- 2017-2020 : VERSATILE (Programme Européen H2020). Responsable scientifique pour le LIRMM. Ce projet concerne le développement de stratégies de perception et de commande nécessaires au robot collaborateur de l'usine du futur. Coordinateur : Fondation Tecnalia (Espagne).

Participation à projets

- 2005-2007 : ASPICE (Telethon Italie) : commande robotique par interface neuronale directe. Partenariat entre l'Université "La Sapienza" et la Fondazione Santa Lucia de Rome (Coordinateur).
- 2008-2011 : ANR CityVIP (Programme TSFA) : conduite automatique de véhicules de transport publiques, en milieu urbain. Coordinateur: Université Blaise Pascal, Clermont Ferrand. Une partie des résultats a été présentée en Sections 3.3.1 et 3.3.2.
- 2010-2015 : VERE (Programme Européen FP7) concernant l'incorporation dans un robot humanoïde d'un opérateur humain (par exemple, handicapé) à l'aide d'une interface cerveau-ordinateur (BCI). Coordinateur : Universitat de Barcelona, Espagne. Une partie des résultats a été présentée en Sect. 6.3.2.

- 2011-2014 : ANR ICARO (Programme CONTINT) : développement d'un robot collaboratif. Coordinateur : AIRBUS. Une partie des résultats a été présentée en Sections 4.3.1 et 4.3.2.
- 2012-2016 : RoboHow (Programme Européen FP7) concernant l'utilisation de base de données pour permettre aux robots d'effectuer des tâches au quotidien. Coordinateur : Universität Bremen, Allemagne. Une partie des résultats a été présentée en Sections 5.3.2 et 5.3.3.
- 2013-2016 : ROMEO 2 (Programme PSPC) poursuivant le développement d'un robot humanoïde, assistant et compagnon pour la vie quotidienne. Coordinateur: Aldebaran.

Révisions d'articles

Réviser, de plus de 70 articles, pour les Conférences Internationales avec Comité de lecture IEEE ICRA, IEEE/RSJ IROS, IEEE ICHR, ACC, RSS, ICAR, ainsi que pour les Revues Internationales IEEE Trans. Robotics, Int. Journal of Robotics Research, Robotics and Autonomous Systems, Autonomous Robots, IEEE Trans. on Industrial Informatics, IEEE Trans. on Intelligent Transportation Systems, IEEE Robotics and Automation Magazine, Journal of Field Robotics, Journal of Intelligent and Robotic Systems, IEEE Trans. on Systems, Man, and Cybernetics Part B, Robotica, Advanced Robotics, et IEEE Trans. Control Systems Technology.

B.2.3 Activités d'enseignement

Enseignement

Avant d'arriver à Montpellier:

- 2006-2008 : cours de *Commande adaptative et Apprentissage automatique* (20 h CM) à l'Université de Rome "La Sapienza", niveau M2.
- 2006-2008 : encadrement, à l'Université de Rome "La Sapienza", des *Projets* de 7 étudiants, niveau L3.
- 2009-2010 : travaux pratiques en *Automatique Linéaire Mono-variable* (48 h TP) à l'INSA, Rennes, niveau L3.
- 2010 : travaux pratiques en *Robotique* (36 h TP), INSA, Rennes, niveau M1.

À la Faculté des Sciences, de l'Université de Montpellier:

- 2011 - 2014: travaux pratiques (81 h TP) et dirigés (26 h TD) en *Automatique des Systèmes Linéaire Mono-variable*, niveau L3.
- 2011 - présent : cours magistraux (29 h CM) et travaux pratiques (96 h TP) en *Automatique des Systèmes Multi-variables*, niveau M1.
- 2011 - présent : cours magistraux (43 h CM) et travaux pratiques (147 h TP) en *Vision et Capteurs pour la robotique*, niveau M2.

- 2011 - présent : encadrement des *projets et stages* d'environ 8 étudiants par an, niveau Master (première et deuxième année).
- 2012 - 2015 : cours magistraux (45 h CM) d'*Outils Informatiques*, niveau L2.
- 2013 - présent : cours magistraux (37 h CM) et travaux pratiques (68 h TP) en *Bases de la Robotique et du Traitement d'images*, niveau M1.
- 2015 - présent : cours magistraux (12 h CM) en *Optimisation*, niveau M2.

Responsabilités pédagogiques

- 2012-2015 : Responsable pédagogique et administratif de l'Unité d'Enseignement *Outils Informatiques*, suivie par tous les étudiants inscrits en 2ème année de Licence à l'UFR Faculté des Sciences de l'Université de Montpellier (environ 1000 étudiants par an). Ce travail a consisté à : recruter moniteurs et vacataires (environ 30 par an), préparer les emplois de temps, réserver les salles d'informatique, organiser le QCM C2I, gérer les équivalences, centraliser les notes.
- 2015-présent : Responsable pédagogique et administratif de la *Spécialité Robotique première et deuxième année du Master EEA*, à l'Université de Montpellier (environ 50 étudiants). Pour la première et deuxième année, il s'agit de: recruter les étudiants, recruter moniteurs et vacataires, et préparer les projets pédagogiques (demandes de moyens). Pour le Master 2ème année, il s'agit de : préparer l'emploi de temps, organiser les jury et les soutenances de projet/stage, faire un suivi des stages en entreprise, organiser les conférences métier.

B.2.4 Diffusion de travaux (rayonnement, vulgarisation)

Publications

Au cours de ma carrière, j'ai publié 12 articles sur revues Internationales, et 34 articles dans des Conférences Internationales avec Comité de lecture (voir Sect. A.4.1).

Séminaires invités

- 2007 : "*The ASPICE project*", Universität Bremen, Allemagne.
- 2007 : "*Multimodal navigation in ASPICE*", KTH, Stockholm, Suède.
- 2009 : "*Visual navigation with obstacle avoidance*", Université de Technologie de Compiègne, France.
- 2010 : "*Outdoor visual navigation with obstacle avoidance*", VisuLab - Università di Parma, Italie.
- 2012 : "*Outdoor Visual Navigation*", Albert-Ludwigs-Universitaet Freiburg, Allemagne.
- 2012 : "*Vision-based control of mobile robots*", CNRS-AIST Joint Robotics Laboratory, Tsukuba, Japon.

- 2012 : *At the crossroads of neurorehabilitation and robotics*, Fondazione Santa Lucia, Rome, Italie.
- 2013 : *“An overview of our research on Human-inspired Cybernetic Systems”*, MPI Tübingen, Allemagne.
- 2014 : *“See and touch: Visio-haptic control for safe human-robot interaction”*, ETH Zurich, Suisse.
- 2014 : *“Visio-haptic control for safe human-robot interaction”*, IEEE ICRA Workshop “Affective Haptics for Human-Robot Interaction”, Hong-Kong, Chine.
- 2015 : *“Visio-haptic control for safe physical interaction with humanoids”*, Journées Nationales de la Robotique Humanoïde, Nantes, France.
- 2016 : *Sensor-based control of humanoids*, Université de Bonn, Allemagne.
- 2016 : *Visio-haptic robot control*, Scuola Superiore Sant’Anna, Pisa, Italie.
- 2016 : *Visio-haptic control for physical human-robot interaction - recent works of the LIRMM IDH Group*, German-French Conference on Humanoid and Legged Robots, HLR 2016, Toulouse, France.

Participation à des jurys de thèse

Au titre de rapporteur:

- ETH Zurich, Suisse : Deon George Sabatta, “Algorithms for vision-based path following along previously taught paths” (2014).
- Albert-Ludwigs-Universitaet Freiburg, Allemagne : Daniel Maier, “Camera-based humanoid robot navigation” (2015).
- Scuola Superiore Sant’Anna, Pisa, Italie : Alessandro Di Fava, “Multisensory Advanced Robot Control: from perceptual sensors to human demonstrations” (2016).

Au titre d’encadrant:

- Université de Montpellier : Don Joven Agravante, “Transport collaboratif homme-humanoïde” (2015).
- Université de Montpellier : Damien Petit, “Commande de robot humanoïde avec interface cerveau-ordinateur” (2015).

Comités de programme

- Éditeur Associé de la conférence IEEE/RSJ IROS en 2014, 2015 et 2016. Le travail consiste d’abord à trouver 2-3 réviseurs pour chaque article assigné par l’Éditeur, ensuite à faire une synthèse des révisions (sous forme de rapport), en donnant un avis (favorable ou défavorable) sur l’acceptation du papier. J’ai été responsable de: 6 articles en 2014 (thèmes: planification et optimisation), et de 11 articles en 2015 (thèmes: perception et localisation).

- Organisateur et initiateur des Workshops “Multimodal sensor-based robot control for Human-Robot Interaction and Soft Manipulation” aux conférences IROS 2015 et 2016. Le travail consisté à: trouver les co-organisateurs (4 collègues à l’international), soumettre une proposition au comité de programme des Workshops IROS, contacter des experts (six) pour les présentations invitées, gérer les révisions des résumés soumis (10 soumissions, avec 2 révision chacune), et organiser sur place la journée. Le premier Workshop (2015) a eu un grand succès (avec environ 60 participants) et sera re-proposé à IROS 2016. Une Special Issue (dont je serai Éditeur Principal) sur les mêmes thématiques, est en cours de préparation avec les Éditeurs de la revue “Robotics and Autonomous Systems”.
- Organisateur, avec 5 collègues, du Workshop “Tactile sensing for manipulation: new progress and challenges” à la conférence IEEE-RAS Int. Conf. on Humanoid Robots, Humanoids 2016. Le travail a consisté à: soumettre une proposition au comité de programme des Workshops ICHR, contacter des experts (onze) pour les présentations invitées, et gérer les révisions des résumés soumis.

Expertises Nationales/Internationales

- 2014, expert pour le Programme Européen CHIST ERA. Le travail a consisté à expertiser un projet de Robotique de 60 pages.
- 2014, expert pour le Programme Jeunes Chercheurs (SIR) du Ministère de l’Instruction, de l’Université et de la Recherche Italien (thème: Robotique). Le travail a consisté, dans un premier temps, à expertiser 6 pré-propositions courtes de projets de robotique (de 5 pages chacune), ensuite à expertiser deux propositions pré-sélectionnées (de 15 pages chacune).
- 2014, membre du comité de sélection du recrutement du poste MCF section 61, Université de Cergy Pontoise (2014), profil : Robotique et Sciences cognitives.
- 2015, expert pour le Programme ANR ACHN (Accueil de Chercheurs de Haut Niveau). Le travail a consisté à expertiser un projet de Traitement de signal de 25 pages.
- 2016, expert pour le Programme PRIN (Projets de Recherche d’Intérêt National) du Ministère de l’Instruction, de l’Université et de la Recherche Italien (thème: Robotique). Le travail a consisté à expertiser trois propositions de projets de robotique, de 30 pages chacune.

Autres

- Lauréat, avec 18 autres jeunes chercheurs (toutes disciplines confondues), du prix régional Chercheurs d’Avenir 2013 (Languedoc- Roussillon). Ce prix a permis de financer le projet CoBot@LR à hauteur de 70000 euros.
- Co-président de la session “Planification non-holonome” à la conférence IROS

2008, et de la session “Asservissement visuel” à ICRA 2013.

- Vulgarisation par le biais de journaux grand public (dont “El País” en Espagne, en 2014) et de la télévision publique italienne RAI (dans le cadre du projet ASPICE). Interviews pour les magazines françaises “Sciences Ouest” (2007 et 2010), et “La Gazette de Montpellier” (2015).

Appendix C

Selected journal articles

I have hereby attached the original printouts of articles:

A. Cherubini, F. Chaumette, **Visual Navigation of a Mobile Robot with Laser-based Collision Avoidance**, *International Journal of Robotics Research*, vol. 32, no. 2, pp. 189-209, 2013.

A. Cherubini, F. Spindler, F. Chaumette, **Autonomous Visual navigation and Laser-based moving obstacle avoidance**, *IEEE Trans. on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2101 - 2110, 2014.

A. Cherubini, R. Passama, P. Fraisse, A. Crosnier, **A unified multimodal control framework for human-robot interaction**, *Robotics and Autonomous Systems*, 70, pp. 106-115, 2015.

A. Cherubini, R. Passama, A. Crosnier, A. Lasnier, P. Fraisse, **Collaborative manufacturing with physical human-robot interaction**, *Robotics and Computer Integrated Manufacturing*, vol. 40, August 2016, Pages 1–13.

Visual navigation of a mobile robot with laser-based collision avoidance

Andrea Cherubini^{1,2} and François Chaumette¹

Abstract

In this paper, we propose and validate a framework for visual navigation with collision avoidance for a wheeled mobile robot. Visual navigation consists of following a path, represented as an ordered set of key images, which have been acquired by an on-board camera in a teaching phase. While following such a path, the robot is able to avoid obstacles which were not present during teaching, and which are sensed by an on-board range scanner. Our control scheme guarantees that obstacle avoidance and navigation are achieved simultaneously. In fact, in the presence of obstacles, the camera pan angle is actuated to maintain scene visibility while the robot circumnavigates the obstacle. The risk of collision and the eventual avoiding behaviour are determined using a tentacle-based approach. The framework can also deal with unavoidable obstacles, which make the robot decelerate and eventually stop. Simulated and real experiments show that with our method, the vehicle can navigate along a visual path while avoiding collisions.

Keywords

vision-based navigation, visual servoing, collision avoidance, integration of vision with other sensors

1. Introduction

A great amount of robotics research focuses on vehicle guidance, with the ultimate goal of automatically reproducing the tasks usually performed by human drivers (Buehler et al., 2008; Zhang et al., 2008; Nunes et al., 2009; Broggi et al., 2010). In many recent works, information from visual sensors is used for localization Guerrero et al. (2008); Scaramuzza and Siegwart (2008) or for navigation (Bonin-Font et al., 2008; López-Nicolás et al., 2010). In the case of autonomous navigation, an important task is obstacle avoidance, which consists of either generating a collision-free trajectory to the goal (Minguez et al., 2008), or decelerating to prevent collision when bypassing is impossible (Wada et al., 2009). Most obstacle avoidance techniques, particularly those that use motion planning (Latombe, 1991), rely on knowledge of a global and accurate map of the environment and obstacles.

Instead of utilizing such a global model of the environment, which would infringe the perception to action paradigm (Sciavicco and Siciliano, 2000), we propose a framework for obstacle avoidance with simultaneous execution of a visual servoing task (Chaumette and Hutchinson, 2006). Visual servoing is a well-known method that uses vision data directly in the control loop, and that has

been applied on mobile robots in many works Mariottini et al. (2007); Allibert et al. (2008); Becerra et al. (2011); López-Nicolás and Sagüés (2011). For example, Mariottini et al. (2007) exploited the epipolar geometry to drive a non-holonomic robot to a desired configuration. A similar approach is presented by Becerra et al. (2011), where the singularities are dealt with more efficiently. The same authors achieved vision-based pose stabilization using a state observer in López-Nicolás and Sagüés (2011). Trajectory tracking is tackled by Allibert et al. (2008) by integrating differential flatness and predictive control.

The visual task that we focus on is appearance-based navigation, which has been the target of our research of Šegvić et al. (2008), Cherubini et al. (2009) and Diosi et al. (2011). In the framework that we have developed in the past,¹ the path is a topological graph, represented by a database of ordered key images. In contrast with other similar approaches, such as that of Booij et al. (2007), our graph

¹INRIA Rennes - Bretagne Atlantique, IRISA, Rennes, France

²LIRMM - Université de Montpellier 2 CNRS, Montpellier, France

Corresponding author:

Andrea Cherubini, LIRMM - Université de Montpellier 2 CNRS, 161 Rue Ada, 34392 Montpellier, France.

Email: andrea.cherubini@lirmm.fr

does not contain forks. Furthermore, as opposed to Royer et al. (2007), Goedemé et al. (2007), Zhang and Kleeman (2009), Fontanelli et al. (2009) and Courbon et al. (2009), we do not use the robot pose for navigating along the path. Instead, our task is purely image-based (as in the approach of Becerra et al. (2010)), and it is divided into a series of subtasks, each consisting of driving the robot towards the next key image in the database. More importantly, to the best of the authors' knowledge, appearance-based navigation frameworks have never been extended to take into account obstacles.

Obstacle avoidance has been integrated into many model-based navigation schemes. Yan et al. (2003) used a range finder and monocular vision to enable navigation in an office environment. The desired trajectory is deformed to avoid sensed obstacles in the work of Lamiraux et al. (2004). Ohya et al. (2008) use a model-based vision system with retroactive position correction. Simultaneous obstacle avoidance and path following are presented by Lapierre et al. (2007), where the geometry of the path (a curve on the ground) is perfectly known. In the approach of Lee et al. (2010), obstacles are circumnavigated while following a path; the radius of the obstacles (assumed cylindrical) is known a priori. In practice, all of these methods are based on the environment 3D model, including, for example, walls and doors, or on the path geometry. In contrast, we propose a navigation scheme which does not require an environment or obstacle model.

One of the most common techniques for model-free obstacle avoidance is the potential field method, originally introduced by Khatib (1985). The gap between global path planning and real-time sensor-based control has been closed with an elastic band (Quinlan and Khatib, 1993), a deformable collision-free path, whose initial shape is generated by a planner, and then deformed in real time, according to the sensed data. Similarly, in the work of Bonnafous et al. (2001) and Von Hundelshausen et al. (2008), a set of trajectories (arcs of circles or 'tentacles') is evaluated for navigating. However, in the work of Bonnafous et al. (2001), a sophisticated probabilistic elevation map is used, and the selection of the optimal tentacle is based on its risk and interest, which both require accurate pose estimation. Similarly, in Von Hundelshausen et al. (2008), the trajectory computation relies on GPS way points, hence once more on the robot pose.

Here, we focus on this problem: a wheeled vehicle, equipped with an actuated pinhole camera and with a forward-looking range scanner, must follow a visual path represented by key images, without colliding with the ground obstacles. The camera detects the features required for navigating, while the scanner senses the obstacles (in contrast with other works, such as that of Kato et al. (2002), only one sensor is used to detect the obstacles). In this sense, our work is similar to that of Folio and Cadenat (2006), where redundancy enables reactive obstacle avoidance, without requiring any 3D model. A robot is redundant

when it has more degrees of freedom (DOFs) than those required for the primary task; then, a secondary task can also be executed. In the work of Folio and Cadenat (2006), the two tasks are respectively visual servoing and obstacle avoidance. However, there are various differences with that work. First, we show that the redundancy approach is not necessary, since we design the two tasks so that they are independent. Second, we can guarantee asymptotic stability of the visual task at all times, in the presence of non-occluding obstacles. Moreover, our controller is compact, and the transitions between safe and unsafe contexts is operated only for obstacle avoidance, while in the work of Folio and Cadenat (2006), three controllers are needed, and the transitions are more complex. This compactness leads to smoothness of the robot behaviour. Finally, Folio and Cadenat (2006) considered a positioning task in indoor environments, whereas we aim at continuous navigation on long outdoor paths.

Let us summarize the other major contributions of our work. An important contribution is that our approach is merely appearance-based, hence simple and flexible: the only information required is the database of key images, and no model of the environment or obstacles is necessary. Hence, there is no need for sensor data fusion nor planning, which can be computationally costly, and requires precise calibration of the camera/scanner pair. We guarantee that the robot will never collide in the case of static, detectable obstacles (in the worse cases, it will simply stop). We also prove that our control law is always well-defined, and that it does not present any local minima. To the best of the authors' knowledge, this is the first time that obstacle avoidance and visual navigation merged directly at the control level (without the need for sophisticated planning) are validated in real outdoor urban experiments.

The framework that we present here is inspired from that designed and validated in our previous work (Cherubini and Chaumette, 2011). However, many modifications have been applied, in order to adapt that controller to the real world. First, for obstacle avoidance, we have replaced classical potential fields with a new tentacle-based technique inspired by Bonnafous et al. (2001) and Von Hundelshausen et al. (2008), which is perfectly suitable for appearance-based tasks, such as visual navigation. In contrast with those works, our approach does not require the robot pose, and exploits the robot geometric and kinematic characteristics (this aspect will be detailed later in the paper). A detailed comparison between the potential field and the tentacle techniques is given in Cherubini et al. (2012). In that work, we showed that with tentacles, smoother control inputs are generated, higher velocities can be applied, and only dangerous obstacles are taken into account. In summary, the new approach is more robust and efficient than its predecessor. A second modification with respect to Cherubini and Chaumette (2011) concerns the design of the translational velocity, which has been changed to improve visual tracking and avoid undesired deceleration in the presence of

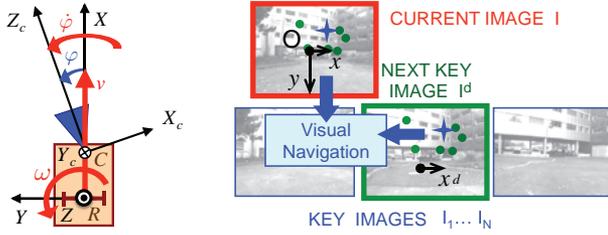


Fig. 1. General definitions. Left: top view of the robot (rectangle), equipped with an actuated camera (triangle); the robot and camera frame (respectively, \mathcal{F}_R and \mathcal{F}_C) are shown. Right: database of the key images, with current and next key images emphasized; the image frame \mathcal{F}_I is also shown, as well as the visual features (circles) and their centroid (cross).

non-dangerous obstacles. Another important contribution of the present work is that, in contrast with the tentacle-based approaches designed in Von Hundelshausen et al. (2008) and Bonnafous et al. (2001), our method does not require the robot pose. Finally, the present article reports experiments, which, for the first time in the field of visual navigation with obstacle avoidance, have been carried out in real-life, unpredictable urban environments.

The article is organized as follows. In Section 2, the characteristics of our problem (visual path following with simultaneous obstacle avoidance) are presented. The control law is presented in full detail in Section 3, and a short discussion is carried out in Section 4. Simulated and real experimental results are presented respectively in Sections 5 and 6, and summarized in the conclusion.

2. Problem definition

2.1. General definitions

The reader is referred to Figure 1. We define the robot frame $\mathcal{F}_R(R, X, Y)$ (R is the robot centre of rotation), image frame $\mathcal{F}_I(O, x, y)$ (O is the image centre) and camera frame $\mathcal{F}_C(C, X_c, Y_c, Z_c)$ (C is the optical centre). The robot control inputs are

$$\mathbf{u} = (v, \omega, \dot{\varphi}).$$

These are, respectively, the translational and angular velocities of the vehicle, and the camera pan angular velocity. We use the normalized perspective camera model:

$$x = \frac{X_c}{Z_c}, \quad y = \frac{Y_c}{Z_c}.$$

We assume that the camera pan angle is bounded: $|\varphi| \leq \frac{\pi}{2}$, and that C belongs to the camera pan rotation axis, and to the robot sagittal plane (i.e. the plane orthogonal to the ground through X). We also assume that the path can be followed with continuous $v(t) > 0$. This ensures safety, since only obstacles in front of the robot can be detected by our range scanner.

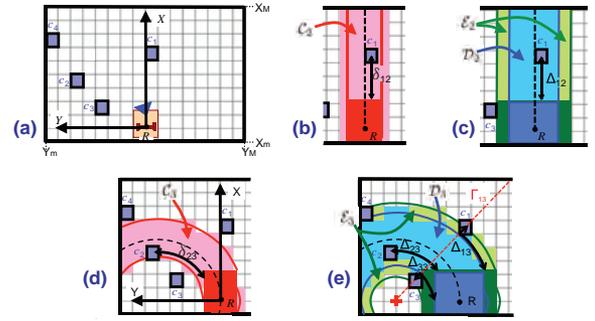


Fig. 2. Obstacle models with four occupied cells c_1, \dots, c_4 . (a) Occupancy grid, straight (b, c) and sharpest counterclockwise (d, e) tentacles (dashed). When a total of three tentacles are used, the straight and sharpest counterclockwise are characterized, respectively, by index $j = 2$ and $j = 3$. For these two tentacles, we have drawn: classification areas (collision C_j , dangerous central D_j , dangerous external E_j), corresponding boxes and delimiting arcs of circle, and cell risk and collision distances (Δ_{ij}, δ_{ij}). For tentacle $j = 3$ in the bottom right, we have also drawn the tentacle centre (cross) and the ray of cell c_1 , denoted by Γ_{13} .

2.2. Visual path following

The path that the robot must follow is represented as a database of ordered key images, such that successive pairs contain some common static visual features (points). First, the vehicle is manually driven along a taught path, with the camera pointing forward ($\varphi = 0$), and all of the images are saved. Afterwards, a subset (database) of N key images I_1, \dots, I_N representing the path (Figure 1, right) is selected. Then, during autonomous navigation, the current image, denoted I , is compared with the next key image in the database, $I^d \in \{I_1, \dots, I_N\}$, and a relative pose estimation between I and I^d is used to check when the robot passes the pose where I^d was acquired.

For key image selection, as well as visual point detection and tracking, we use the algorithm presented in Royer et al. (2007). The output of this algorithm, which is used by our controller, is the set of points visible both in I and I^d . Then, navigation consists of driving the robot forward, while I is driven towards I^d . We maximize similarity between I and I^d using only the abscissa x of the centroid of the points matched on I and I^d . When I^d has been passed, the next image in the set becomes the desired one, and so on, until I_N is reached.

2.3. Obstacle representation

Along with the visual path following problem, we consider obstacles which are on the path, but not in the database, and sensed by the range scanner in a plane parallel to the ground. We use the occupancy grid in Figure 2(a): it is linked to \mathcal{F}_R , with cell sides parallel to X and Y . Its longitudinal and lateral extensions are limited ($X_m \leq X \leq X_M$ and $Y_m \leq Y \leq Y_M$), to ignore obstacles that are too far to

jeopardize the robot. The size of the grid should increase with the robot velocity, to guarantee the sufficient time for obstacle avoidance. An appropriate choice for $|X_m|$ is the length of the robot, since obstacles behind cannot hit it as it advances. In this work, we use $X_M = Y_M = 10$ m, $X_m = -2$ m, $Y_m = -10$ m. Any grid cell c centred at (X, Y) is considered occupied if an obstacle has been sensed in c . The cells have size 0.2×0.2 m. For the cells entirely lying in the scanner area, only the current scanner reading is considered. For all other cells in the grid, we use past readings, which are progressively displaced using odometry.

We use, along with the set of all occupied grid cells

$$\mathcal{O} = \{c_1, \dots, c_n\},$$

a set of drivable paths (tentacles). Each tentacle j is a semi-circle that starts in R , is tangent to X , and is characterized by its curvature (i.e. inverse radius) κ_j , which belongs to \mathcal{K} , a uniformly sampled set:

$$\kappa_j \in \mathcal{K} = \{-\kappa_M, \dots, 0, \dots, \kappa_M\}.$$

The maximum desired curvature $\kappa_M > 0$, must be feasible considering the robot kinematics. Since, as we will show, our tentacles are used both for perception and motion execution, a compromise between computational cost and control accuracy must be reached to tune the size of \mathcal{K} , i.e. its sampling interval. Indeed, a large set is costly since, as we show later, various collision variables must be calculated on each tentacle. On the other hand, extending the set enhances motion precision, since more alternative tentacles can be selected for navigation. In the simulations and experiments, we used 21 tentacles. In Figure 2(b)–(e), the straight and the sharpest counterclockwise ($\kappa = \kappa_M$) tentacles are dashed. When a total of three tentacles are used, these correspond respectively to $j = 2$ and $j = 3$.

Each tentacle j is characterized by three classification areas (collision, dangerous central and dangerous external), which are obtained by rigidly displacing, along the tentacle, three rectangular boxes, with increasing size. The boxes are all overestimated with respect to the real robot dimensions. For each tentacle j , the sets of cells belonging to the three classification areas (shown in Figure 2) are denoted \mathcal{C}_j , \mathcal{D}_j and \mathcal{E}_j . Cells belonging to the dangerous central set, are not considered in the dangerous external set as well, so that $\mathcal{D}_j \cap \mathcal{E}_j = \emptyset$. The sets \mathcal{O} , \mathcal{C} , \mathcal{D} and \mathcal{E} are used to calculate the variables required in the control law defined in Section 3.1: in particular, the largest classification areas \mathcal{D} and \mathcal{E} are used to select the safest tentacle and its risk, while the thinnest one \mathcal{C} determines the eventual necessary deceleration.

In summary, as we mentioned in Section 1, our tentacles exploit the robot geometric and kinematic characteristics. Specifically, the robot geometry (i.e. the vehicle encumbrance) defines the three classification areas \mathcal{C} , \mathcal{D} and \mathcal{E} , hence the cell potential danger, while the robot kinematics (i.e. the maximum desired curvature, κ_M) define the bounds

on the set of tentacles \mathcal{K} . Both aspects give useful information on possible collisions with obstacles ahead of the robot, which will be exploited, as we will show in Section 3, to choose the best tentacle and to eventually slow down or stop the robot.

2.4. Task specifications

Let us recall the Jacobian paradigm which relates a robot kinematic control inputs with the desired task. We name $\mathbf{s} \in \mathbb{R}^m$ the task vector, and $\mathbf{u} \in \mathbb{R}^m$ the control inputs. The task dynamics are related to the control inputs by

$$\dot{\mathbf{s}} = \mathbf{J}\mathbf{u}, \quad (1)$$

where \mathbf{J} is the task Jacobian of size $m \times m$. In this work, $m = 3$, and the desired specifications are

1. orienting the camera in order to drive the abscissa of the feature centroid x to its value at the next key image in the database x^d ;
2. making the vehicle progress forward along the path (except if obstacles are unavoidable);
3. avoiding collision with the obstacles, while remaining near the 3D taught path.

The required task evolution can be written

$$\dot{\mathbf{s}}^* = \dot{\mathbf{s}}^d - \Lambda (\mathbf{s} - \mathbf{s}^d), \quad (2)$$

with \mathbf{s}^d and $\dot{\mathbf{s}}^d$ indicating the desired values of the task, and of its first derivative, and $\Lambda = \text{diag}(\lambda_1 \dots \lambda_m)$ a positive-definite diagonal gain matrix.

Since we assume that the visual features are static, the first specification on camera orientation can be expressed by

$$\dot{x}^* = -\lambda_x (x - x^d), \quad (3)$$

with λ_x a positive scalar gain. This guarantees that the abscissa of the centroid of the points converges exponentially to its value at the next key image x^d , with null velocity there ($\dot{x}^d = 0$). The dynamics of this task can be related to the robot control inputs by

$$\dot{x} = \mathbf{J}_x \mathbf{u} = \begin{bmatrix} j_v & j_\omega & j_{\dot{\varphi}} \end{bmatrix} \mathbf{u}, \quad (4)$$

where j_v , j_ω and $j_{\dot{\varphi}}$ are the components of the centroid abscissa Jacobian \mathbf{J}_x related to each of the three robot control inputs. Their form will be determined in Section 3.2.

The two other specifications (vehicle progression with collision avoidance) are related to the danger represented by the obstacles present in the environment. If it is possible, the obstacles should be circumnavigated. Otherwise, the vehicle should stop to avoid collision. To determine the best behaviour, we assess the danger at time t with a situation risk function $H : \mathbb{R}^{*+} \mapsto [0, 1]$, that will be fully defined in Section 3.3.

- In the safe context ($H = 0$), no dangerous obstacles are detected on the robot path. In this case, it is desirable that the robot acts as in the teaching phase, i.e. following the taught path with the camera looking forward. If the current pan angle φ is non-null, which is typically the case when the robot has just avoided an obstacle, an exponential decrease of φ is specified. Moreover, the translational velocity v must be reduced in the presence of sharp turns, to ease the visual tracking of quickly moving features in the image; we specify this using a function v_s that will be detailed in Section 3.4. In summary, the specifications in the safe context are

$$\begin{cases} \dot{x} = -\lambda_x (x - x^d) \\ v = v_s \\ \dot{\varphi} = -\lambda_\varphi \varphi \end{cases}, \quad (5)$$

with λ_φ a positive scalar gain. The corresponding current and desired task dynamics are

$$\dot{\mathbf{s}}_s = \begin{bmatrix} \dot{x} \\ v \\ \dot{\varphi} \end{bmatrix}, \quad \dot{\mathbf{s}}_s^* = \begin{bmatrix} -\lambda_x (x - x^d) \\ v_s \\ -\lambda_\varphi \varphi \end{bmatrix}. \quad (6)$$

Using (4) we can derive the Jacobian relating $\dot{\mathbf{s}}_s$ and \mathbf{u} :

$$\dot{\mathbf{s}}_s = \mathbf{J}_s \mathbf{u} = \begin{bmatrix} j_v & j_\omega & j_\varphi \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{u}. \quad (7)$$

Note that matrix \mathbf{J}_s is invertible if $j_\omega \neq 0$, and we show in Section 3.2 that this condition is indeed ensured.

- In the unsafe context ($H = 1$), dangerous obstacles are detected. The robot should circumnavigate them by following the best tentacle (selected by considering both the visual and avoidance tasks as we show in Section 3.3). This heading variation drives the robot away from the 3D taught path. Correspondingly, the camera pan angle must be actuated to maintain visibility of the database features, i.e. to guarantee (3). The translational velocity must be reduced for safety reasons (i.e. to avoid collisions); we specify this using a function v_u , that is defined in Section 3.5. In summary, the specifications in the unsafe context are

$$\begin{cases} \dot{x} = -\lambda_x (x - x^d) \\ v = v_u \\ \omega = \kappa_b v_u \end{cases}, \quad (8)$$

where κ_b is the best tentacle curvature, so that the translational and angular velocities guarantee that the robot precisely follows it, since $\omega/v = \kappa_b$. The current and desired task dynamics corresponding to (8) are

$$\dot{\mathbf{s}}_u = \begin{bmatrix} \dot{x} \\ v \\ \omega \end{bmatrix}, \quad \dot{\mathbf{s}}_u^* = \begin{bmatrix} -\lambda_x (x - x^d) \\ v_u \\ \kappa_b v_u \end{bmatrix}. \quad (9)$$

Using (4) we can derive the Jacobian relating $\dot{\mathbf{s}}_u$ and \mathbf{u} :

$$\dot{\mathbf{s}}_u = \mathbf{J}_u \mathbf{u} = \begin{bmatrix} j_v & j_\omega & j_\varphi \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \mathbf{u}. \quad (10)$$

Matrix \mathbf{J}_u is invertible if $j_\varphi \neq 0$, and we show in Section 3.2 that this condition is also ensured.

- In intermediate contexts ($0 < H < 1$), the robot should navigate between the taught path and the best tentacle. The transition between these two extremes will be driven by situation risk function H .

3. Control scheme

3.1. General scheme

An intuitive choice for controlling (1) in order to fulfill the desired tasks \mathbf{s}_u and \mathbf{s}_s would be

$$\mathbf{u} = \mathbf{J}^{-1} \dot{\mathbf{s}}^* \quad (11)$$

with

$$\mathbf{s} = H \mathbf{s}_u + (1 - H) \mathbf{s}_s$$

and therefore (considering $\dot{H} = 0$):

$$\mathbf{J} = H \mathbf{J}_u + (1 - H) \mathbf{J}_s = \begin{bmatrix} j_v & j_\omega & j_\varphi \\ 1 & 0 & 0 \\ 0 & H & 1 - H \end{bmatrix}.$$

In fact, away from singularities of \mathbf{J} , controller (11) leads to the linear system:

$$\dot{\mathbf{s}} - \dot{\mathbf{s}}^d = -\Lambda (\mathbf{s} - \mathbf{s}^d)$$

for which, as desired, $(\mathbf{s}^d, \dot{\mathbf{s}}^d)$ are exponentially stable equilibria, for any value of $H \in [0, 1]$ (since Λ is a positive-definite diagonal matrix). Note that replacing (2) in (11), leads to the well-known controller for following trajectory $s^d = s^d(t)$, given by (Chaumette and Hutchinson, 2007)

$$\mathbf{u} = -\Lambda \mathbf{J}^{-1} (\mathbf{s} - \mathbf{s}^d) + \mathbf{J}^{-1} \dot{\mathbf{s}}^d.$$

However, the choice of controller (11) is not appropriate for our application, since \mathbf{J} is singular whenever

$$H j_\varphi + (H - 1) j_\omega = 0. \quad (12)$$

This condition, which depends on visual variables (j_φ and j_ω) as well as on an obstacle variable (H), can occur in practice.

Instead, we propose the following control law to guarantee a smooth transition between the inputs:

$$\mathbf{u} = H \mathbf{J}_u^{-1} \dot{\mathbf{s}}_u^* + (1 - H) \mathbf{J}_s^{-1} \dot{\mathbf{s}}_s^*. \quad (13)$$

Replacing this equation in (7) and (10), guarantees that controller (13) leads to convergence to the desired tasks (5) and (8):

$$\begin{aligned} \dot{\mathbf{s}}_s &= \dot{\mathbf{s}}_s^* & \text{if } H = 0 \\ \dot{\mathbf{s}}_u &= \dot{\mathbf{s}}_u^* & \text{if } H = 1 \end{aligned}$$

and that, in these cases, the desired states are globally asymptotically stable for the closed loop system. In Section 4, we show that global asymptotic stability of the visual task is also guaranteed in the intermediate cases ($0 < H < 1$).

In the following, we define the variables introduced in Section 2.4. We show how to derive the centroid abscissa Jacobian \mathbf{J}_x , the situation risk function H , the best tentacle along with its curvature κ_b , and the translational velocities in the safe and unsafe context (v_s and v_u , respectively).

3.2. Jacobian of the centroid abscissa

We will hereby derive the components of \mathbf{J}_x introduced in (4). Let us define $\mathbf{v} = (v_c, \omega_c)$ the camera velocity, expressed in \mathcal{F}_C . Since we have assumed that the features are static, the dynamics of x can be related to \mathbf{v} by

$$\dot{x} = \mathbf{L}_x \mathbf{v}$$

where \mathbf{L}_x is the interaction matrix of x (Chaumette and Hutchinson, 2006). In the case of a point of depth Z_c , it is given by (Chaumette and Hutchinson, 2006)

$$\mathbf{L}_x = \begin{bmatrix} -\frac{1}{Z_c} & 0 & \frac{x}{Z_c} & xy & -1 - x^2 & y \end{bmatrix}. \quad (14)$$

In theory, since we consider the centroid and not a physical point, we should not use (14) for the interaction matrix, but the exact and more complex form given in Tahri and Chaumette (2005). However, using (14) provides a sufficiently accurate approximation (Cherubini et al., 2009). It also has the strong advantage that it is not necessary to estimate the depth of all points, using techniques such as those described by Davison et al. (2007), De Luca et al. (2008) and Durand et al. (2010). Only an approximation of Z_c , i.e. one scalar, is sufficient. In practice, we set a constant fixed value. This strategy has proved successful for visual navigation in Cherubini et al. (2009).

For the robot model that we are considering, the camera velocity \mathbf{v} can be expressed as a function of \mathbf{u} by using the geometric model:

$$\mathbf{v} = {}^C \mathbf{T}_R \mathbf{u}$$

with:

$${}^C \mathbf{T}_R = \begin{bmatrix} \sin \varphi & -X^C \cos \varphi & 0 \\ 0 & 0 & 0 \\ \cos \varphi & X^C \sin \varphi & 0 \\ 0 & 0 & 0 \\ 0 & -1 & -1 \\ 0 & 0 & 0 \end{bmatrix}.$$

In this matrix, X^C is the abscissa of the optical centre C in the robot frame \mathcal{F}_R . This parameter is specific of the robot platform. Since C belongs to the robot sagittal plane, and since the robot is constrained on the ground plane, this is the only coordinate of C in \mathcal{F}_R required for visual servoing.

Then, multiplying \mathbf{L}_x by ${}^C \mathbf{T}_R$, we obtain the components of \mathbf{J}_x :

$$\begin{aligned} j_v &= \frac{-\sin \varphi + x \cos \varphi}{Z_c} \\ j_\omega &= \frac{X^C (\cos \varphi + x \sin \varphi)}{Z_c} + 1 + x^2 \\ j_\varphi &= 1 + x^2. \end{aligned} \quad (15)$$

From (15) it is clear that $j_\varphi \geq 1 \forall x \in \mathbb{R}$; hence, \mathbf{J}_x is never singular (see (10)). Furthermore, it is possible to ensure that $j_\omega \neq 0$, so that \mathbf{J}_s is also invertible (see (7)). In fact, in (15) we can guarantee that $j_\omega \neq 0$, by setting $Z_c > \frac{X^C}{2}$ in the \mathbf{J}_x components. Indeed, condition $j_\omega \neq 0$ is equivalent to

$$\frac{X^C (\cos \varphi + x \sin \varphi)}{Z_c} + 1 + x^2 \neq 0. \quad (16)$$

Since $|\varphi| \leq \frac{\pi}{2}$: $\cos \varphi + x \sin \varphi \geq -x$, $\forall x \in \mathbb{R}$. Hence, a sufficient condition for (16) is

$$x^2 - \frac{X^C}{Z_c} x + 1 > 0,$$

which occurs $\forall x \in \mathbb{R}$ when $X^C/Z_c < 2$. In practice, this condition can be guaranteed, since X^C is an invariant characteristic of the robot platform, and Z_c is a tunable control parameter, which can be set to a value greater than $X^C/2$. In addition, the value of X^C on most robots platforms is usually smaller than 1 m, which is much less than the scene depth in outdoor environments. In Cherubini et al. (2009), we have shown that overestimating Z_c does not jeopardize navigation.

On the other hand, we can infer from (15) that the singularity of controller (11), expressed by (12) can occur frequently. For example, whenever Z_c is large, yielding $j_\varphi \approx j_\omega$, and concurrently $H \approx 0.5$, \mathbf{J} becomes singular. This confirms the great interest in choosing control scheme (13), which is always well defined if $Z_c > \frac{X^C}{2}$.

3.3. Situation risk function and best tentacle

To derive the situation risk function H used in (13), we first calculate a candidate risk function $H_j \in [0, 1]$ for each tentacle, as will be explained below. Each H_j is derived from the risk distance of all occupied cells in the dangerous areas.

This distance is denoted $\Delta_{ij} \geq 0$ for each $c_i \in \mathcal{O} \cap (\mathcal{D}_j \cup \mathcal{E}_j)$. For occupied cells in the central set \mathcal{D}_j , Δ_{ij} is the distance that the middle boundary box would cover along tentacle j before touching the cell centre. For occupied cells in the external set, only a subset $\tilde{\mathcal{E}}_j$ is taken into account: $\tilde{\mathcal{E}}_j \subseteq \mathcal{O} \cap \mathcal{E}_j$. This subset contains only cells which reduce the clearance in the tentacle normal direction. For each external occupied cell, we denote Γ_{ij} the ray starting at the tentacle centre and passing through c_i . Cell c_i is added to $\tilde{\mathcal{E}}_j$ if and only if, in $\mathcal{D}_j \cup \mathcal{E}_j$, there is at least an occupied cell crossed by Γ_{ij} on the other side of the tentacle. In the example of Figure 2(e), $\mathcal{O} \cap \mathcal{E}_3 = \{c_1, c_3, c_4\}$, whereas $\tilde{\mathcal{E}}_3 = \{c_1, c_3\}$. Cell c_4 is not considered dangerous, since it is external, and does not have a counterpart on the other

side of the tentacle. Then, for cells in $\bar{\mathcal{E}}_j$, Δ_{ij} is the sum of two terms: the distance from the centre of c_i to its normal projection on the perimeter of the dangerous central area, and the distance that the middle boundary box would cover along tentacle j before reaching the normal projection. The derivation of Δ_{ij} is illustrated, in Figure 2, for four occupied cells. Note that for a given cell, Δ_{ij} may have different values (or even be undefined) according to the tentacle that is considered.

When all risk distances on tentacle j are calculated, we compute Δ_j as their minimum:

$$\Delta_j = \inf_{c_i \in (\mathcal{O} \cap \mathcal{D}_j) \cup \bar{\mathcal{E}}_j} \Delta_{ij}.$$

If $(\mathcal{O} \cap \mathcal{D}_j) \cup \bar{\mathcal{E}}_j \equiv \emptyset$, $\Delta_j = \infty$. In the example of Figure 2, $\Delta_2 = \Delta_{12}$ and $\Delta_3 = \Delta_{33}$. Obviously, overestimating the bounding box sizes leads to more conservative Δ_j .

We then use Δ_j and two hand-tuned thresholds Δ_d and Δ_s ($0 < \Delta_d < \Delta_s$) to design the tentacle risk function:

$$H_j = \begin{cases} 0 & \text{if } \Delta_j \geq \Delta_s \\ \frac{1}{2} \left[1 + \tanh \left(\frac{1}{\Delta_j - \Delta_d} + \frac{1}{\Delta_j - \Delta_s} \right) \right] & \text{if } \Delta_d < \Delta_j < \Delta_s \\ 1 & \text{if } \Delta_j \leq \Delta_d. \end{cases} \quad (17)$$

Note that H_j smoothly varies from 0, when the dangerous cells associated with tentacle j (if any) are far, to 1, when they are near. If $H_j = 0$, the tentacle is tagged as clear. In practice, threshold Δ_s must be set to the risk distance for which the context ceases to be safe (H becomes greater than 0), so that the robot starts to leave the taught (occupied) path. On the other hand, Δ_d must be tuned as the risk distance for which the context becomes unsafe ($H = 1$), so that the robot follows the best tentacle to circumnavigate the obstacle. In our work, we used the values $\Delta_s = 6$ m and $\Delta_d = 4.5$ m. The risk function H_j corresponding to these values is plotted in Figure 3.

The H_j of all tentacles are then compared, in order to determine H in (13). Initially, we calculate the path curvature $\kappa = \omega/v \in \mathbb{R}$ that the robot would follow if there were no obstacles. Replacing $H = 0$ in (13), it is

$$\kappa = [\lambda_x (x^d - x) - j_v v_s + \lambda_\varphi j_\varphi \varphi] / j_\omega v_s,$$

which is always well defined, since $j_\omega \neq 0$ and we have set $v_s > 0$. We obviously constrain κ to the interval of feasible curvatures $[-\kappa_M, \kappa_M]$. Then, we derive the two neighbours of κ among all of the existing tentacle curvatures:

$$\kappa_n, \kappa_m \in \mathcal{K} \quad \text{such that } \kappa \in [\kappa_n, \kappa_m].$$

Let κ_n be the nearest, i.e. the curvature of the tentacle that best approximates the safe path.² We denote it as the visual task tentacle. The situation risk function H_v of that tentacle is then obtained by linear interpolation of its neighbours:

$$H_v = \frac{(H_m - H_n) \kappa + H_n \kappa_m - H_m \kappa_n}{\kappa_m - \kappa_n}. \quad (18)$$

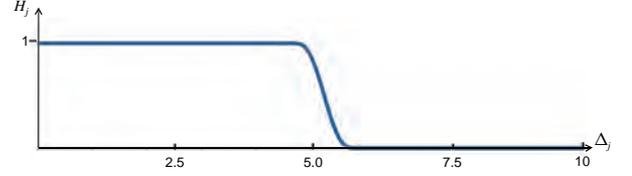


Fig. 3. Risk H_j as a function of the tentacle risk distance Δ_j (m) when $\Delta_s = 6$ m and $\Delta_d = 4.5$ m.

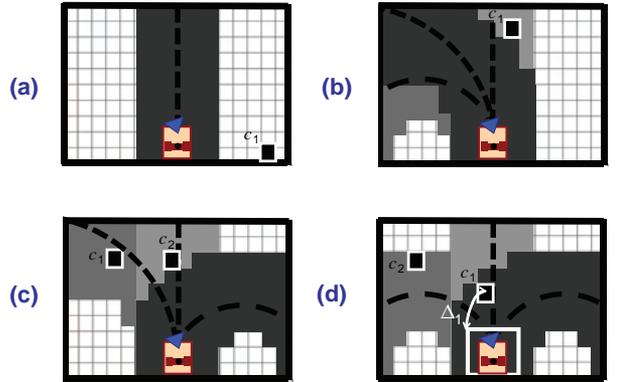


Fig. 4. Strategy for selecting the best tentacle among 5 in four different scenarios. The cells associated to the visual task tentacle, to the previous best tentacle, and to the best tentacle are shown in increasingly dark grey; the corresponding tentacles are dashed, and the occupied cells c_1 and c_2 are shown in black. (a) Since it is clear, the visual task tentacle with curvature κ_n is selected: $\kappa_b = \kappa_n$. (b) The clear tentacle with curvature in $[\kappa_n, \kappa_{pb}]$ nearest to κ_n is chosen. (c) Since all tentacles with curvature in $[\kappa_n, \kappa_{pb}]$ are occupied, the clear one nearest to the visual task tentacle is chosen. (d) Since all tentacles are occupied, we select the one with smallest H_j , hence, largest risk distance Δ_j (here, Δ_1) given the middle boundary box.

In practice, H_v measures the risk on the visual path, by considering only obstacles on the visual task tentacle and on its neighbour tentacle. In particular, for the context to be safe (i.e. in order to follow the taught path and realize the desired safe task in (6)), it is sufficient that the neighbour tentacles are clear ($H_n = H_m = 0$). In this way, obstacles on the sides do not deviate the robot away from the taught path.

Let us now detail our strategy for determining the best tentacle curvature κ_b for navigation. This strategy is illustrated by the four examples in Figure 4, where 5 tentacles are used. In the figure, the dangerous cells (i.e. for each tentacle j , the cells in $\mathcal{D}_j \cup \mathcal{E}_j$) associated with the visual task tentacle and to the best tentacle are respectively shown in light and dark grey. The occupied cells are shown in black. The best tentacle is derived from the tentacle risk functions defined just above. If $H_v = 0$ (as in Figure 4(a)), the visual task tentacle can be followed: we set $\kappa_b = \kappa_n$, and we apply (13) with $H = 0$. Instead, if $H_v \neq 0$, we seek a clear tentacle ($H_j = 0$). First, to avoid abrupt control changes, we only search among the tentacles between the

visual task one and the best one at the previous iteration,³ denoted by κ_{pb} , and mid-grey in the figure. If many clear ones are present, the nearest to the visual task tentacle is chosen, as in Figure 4(b). If none of the tentacles with curvature in $[\kappa_n, \kappa_{pb}]$ is clear, we search among the others. Again, the best tentacle will be the clear one that is closest to κ_n and, in case of ambiguity, the one closest to κ_{nm} . If a clear tentacle has been found (as in Figure 4(c)), we select it and set $H = 0$. Instead, if no tentacle in \mathcal{K} is clear, the one with minimum H_j calculated using (17) is chosen, and H is set equal to that H_j . In the example of Figure 4(d), tentacle 1 is chosen and we set $H = H_1$, since $\Delta_1 = \sup\{\Delta_1, \dots, \Delta_5\}$, hence $H_1 = \inf\{H_1, \dots, H_5\}$. Eventual ambiguities are again solved first with the distance from κ_n , then from κ_{nm} .

3.4. Translational velocity in the safe context

We define here the translational velocity in the safe context v_s . When the feature motion in the image is fast, the visual tracker is less effective, and the translational velocity should be reduced. This is typically the case at sharp robot turns, and when the camera pan angle φ is strong (since the robot is far from the taught 3D path). Hence, we define v_s as

$$v_s(\omega, \varphi) = v_m + \frac{v_M - v_m}{4} \sigma \quad (19)$$

with function σ defined as

$$\sigma : \mathbb{R} \times \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \rightarrow [0, 4]$$

$$(\omega, \varphi) \mapsto [1 + \tanh(\pi - k_\omega |\omega|)] [1 + \tanh(\pi - k_\varphi |\varphi|)].$$

Function (19) has an upper bound $v_M > 0$ (for $\varphi = \omega = 0$), and smoothly decreases to the lower bound $v_m > 0$, as either $|\varphi|$ or $|\omega|$ grow. Both v_M and v_m are hand-tuned variables, and the decreasing trend is determined by empirically tuned positive parameters k_ω and k_φ . This definition of v_s yields better results, both in terms of performances and smoothness than that in Cherubini and Chaumette (2011), which was only characterized by the image x variation. On the left of Figure 5, we have plotted v_s for $v_m = 0.4 \text{ m s}^{-1}$, $v_M = 1 \text{ m s}^{-1}$, $k_\omega = 13$ and $k_\varphi = 3$.

3.5. Translational velocity in the unsafe context

The unsafe translational velocity v_u must adapt to the potential danger; it is derived from the obstacles on the best tentacle, defined in Section 3.3. In fact, v_u is derived from the collision distance δ_b , which is a conservative approximation of the maximum distance that the robot can travel along the best tentacle without colliding. Since the thinner box contains the robot, if R follows the best tentacle, collisions can only occur in occupied cells in \mathcal{C}_b . In fact, the collision with cell c_i will occur at the distance, denoted by $\delta_{ib} \geq 0$, that the thinner box would cover along the best tentacle, before touching the centre of c_i . The derivation of δ_{ib} is illustrated in Figure 2 for four occupied cells.

Then, we define δ_b as the minimum among the collision distances of all occupied cells in \mathcal{C}_b :

$$\delta_b = \inf_{c_i \in \mathcal{O} \cap \mathcal{C}_b} \delta_{ib}.$$

If all cells in \mathcal{C}_b are free, $\delta_b = \infty$. In the example of Figure 2, assuming the best tentacle is the straight one ($b = 2$), $\delta_b = \delta_{12}$. Again, oversizing the box leads to more conservative δ_b .

The translational velocity must be designed accordingly. Let δ_d and δ_s be two hand-tuned thresholds such that $0 < \delta_d < \delta_s$. If the probable collision is far enough ($\delta_b \geq \delta_s$), the translational velocity can be maintained at the safe value defined in (19). Instead, if the dangerous occupied cell is near ($\delta_b \leq \delta_d$), the robot should stop. To comply with the boundary conditions $v_u(\delta_d) = 0$ and $v_u(\delta_s) = v_s$, in the intermediate situations we apply a constant deceleration:

$$a = v_s^2 / 2(\delta_d - \delta_s) < 0.$$

Since the distance required for braking at velocity v_u (δ_b) is

$$\delta_b - \delta_d = -v_u^2 / 2a,$$

the general expression of the unsafe translational velocity becomes

$$v_u(\delta_b) = \begin{cases} v_s & \text{if } \delta_b \geq \delta_s \\ v_s \sqrt{\delta_b - \delta_d / \delta_s - \delta_d} & \text{if } \delta_d < \delta_b < \delta_s \\ 0 & \text{if } \delta_b \leq \delta_d, \end{cases} \quad (20)$$

in order to decelerate as the collision distance δ_b decreases. In practice, threshold δ_d will be chosen as the distance to collision at which the robot should stop. Instead, threshold δ_s must be defined according to the maximum applicable deceleration (denoted $a_M < 0$), in order to brake before reaching distance δ_d , even when the safe velocity v_s is at its maximum v_M :

$$\delta_s > \delta_d - \frac{2a_M}{v_M^2}.$$

In our work, we used $\delta_d = 2.7 \text{ m}$ and $\delta_s = 5 \text{ m}$, as shown in the right-hand side of Figure 5, where we have plotted v_u in function of δ_b for three values of v_s : 0.4, 0.7 and 1 m s^{-1} .

4. Discussion

In this section, we instantiate and comment on our control scheme for visual navigation with obstacle avoidance. Using all the variables defined above, we can explicitly write our controller (13) for visual navigation with obstacle avoidance:

$$\begin{cases} v = (1 - H) v_s + H v_u \\ \omega = (1 - H) \frac{\lambda_x (x^d - x) - j_v v_s + \lambda_\varphi j_\varphi \varphi}{j_\omega} + H \kappa_b v_u \\ \dot{\varphi} = H \frac{\lambda_x (x^d - x) - (j_v + j_\omega \kappa_b) v_u}{j_\varphi} - (1 - H) \lambda_\varphi \varphi. \end{cases} \quad (21)$$

This control law has the following interesting properties.

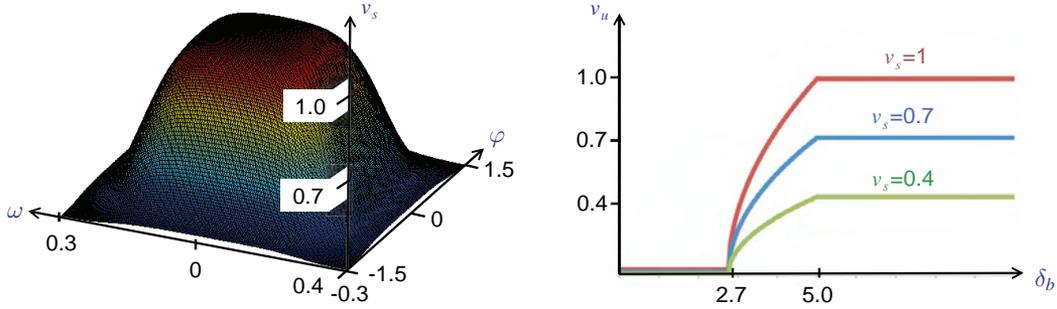


Fig. 5. Left: safe translational velocity v_s (m s⁻¹) as a function of ω (rad s⁻¹) and φ (rad). Right: unsafe translational velocity v_u (m s⁻¹) as a function of δ_b (m) for three different values of v_s .

1. In the safe context ($H = 0$), Equation (21) becomes

$$\begin{cases} v = v_s \\ \omega = \frac{\lambda_x(x^d - x) - j_v v_s + \lambda_\varphi j_\varphi \dot{\varphi}}{j_\omega} \\ \dot{\varphi} = -\lambda_\varphi \varphi. \end{cases} \quad (22)$$

In Section 3.1, we proved that this controller guarantees global asymptotic stability of the safe task \dot{s}_s^* . As in Cherubini et al. (2009) and Diosi et al. (2011), where obstacles were not considered, the image error is controlled only by ω , which also compensates the centroid displacements due to v and to $\dot{\varphi}$ through the image Jacobian components (15), to fulfil the visual task (3). The two remaining specifications in (5), instead, are achieved by inputs v and $\dot{\varphi}$: the translational velocity is regulated to improve tracking according to (19), while the camera is driven forward, to $\varphi = 0$. Note that, to obtain $H = 0$ with the tentacle approach, it is sufficient that the neighbour tentacles are clear ($H_n = H_m = 0$), whereas in the potential field approach used by Cherubini and Chaumette (2011), even a single occupied cell would generate $H > 0$. Thus, one advantage of the new approach is that only obstacles on the visual path are taken into account.

2. In the unsafe context ($H = 1$), Equation (21) becomes

$$\begin{cases} v = v_u \\ \omega = \kappa_b v_u \\ \dot{\varphi} = \frac{\lambda_x(x^d - x) - (j_v + j_\omega \kappa_b) v_u}{j_\varphi}. \end{cases} \quad (23)$$

In Section 3.1, we proved that this controller guarantees global asymptotic stability of the unsafe task \dot{s}_u^* . In this case, the visual task (3) is executed by $\dot{\varphi}$, while the two other specifications are ensured by the two other DOFs: the translational velocity is reduced (and even zeroed to $v = v_u = 0$ for very near obstacles such that $\delta_b \leq \delta_d$), while the angular velocity makes the robot follow the best tentacle ($\omega/v = \kappa_b$). Note that, since no 3D positioning sensor (e.g. GPS) is used, closing the loop on the best tentacle is not possible; however, even if the robot slips (e.g. due to a flat tyre), at the following iterations tentacles with stronger curvature will be

- selected to drive it towards the desired path, and so on. Finally, the camera velocity $\dot{\varphi}$ in (23) compensates the robot rotation, to keep the features in view.
3. In intermediate situations ($0 < H < 1$), the robot navigates between the taught path, and the best path considering obstacles. The situation risk function H representing the danger on the taught path, drives the transition, but not the speed. In fact, note that, for all $H \in [0, 1]$, when $\delta_b \geq \delta_s$: $v = v_s$. Hence, a high velocity can be applied if the path is clear up to δ_s (e.g. when navigating behind another vehicle).
4. Control law (21) guarantees that obstacle avoidance has no effect on the visual task, which can be achieved for any $H \in [0, 1]$. Note that plugging the expressions of v , ω and $\dot{\varphi}$ from (21) into the visual task equation:

$$\dot{x} = j_v v + j_\omega \omega + j_\varphi \dot{\varphi} \quad (24)$$

- yields (3). Therefore, the desired state x^d is globally asymptotically stable for the closed-loop system, $\forall H \in [0, 1]$. This is true even in the special case where $v = 0$. In fact, the robot stops and v becomes null, if and only if $H = 1$ and $v_u = 0$, implying that $\omega = 0$ and $\dot{\varphi} = \frac{\lambda_x(x^d - x)}{j_\varphi}$, which allows realization of the visual task. In summary, from a theoretical control viewpoint (i.e. without considering image processing nor field of view or joint limits constraints), this proves that if at least one point in I^d is visible, the visual task of driving the centroid abscissa to x^d will be achieved, even in the presence of unavoidable obstacles. This strategy is very useful for recovery: since the camera stays focused on the visual features, as soon as the path returns free, the robot can follow it again.
5. Controller (21) does not present local minima, i.e. non-desired state configurations for which \mathbf{u} is null. In fact, when $H < 1$, $\mathbf{u} = 0$ requires both v_u and v_s to be null, but this is impossible since from (19), $v_s > v_m > 0$. Instead, when $H = 1$, it is clear from (23) that for \mathbf{u} to be null it is sufficient that $x^d - x = 0$ and $v_u = 0$. This corresponds to null desired dynamics: $\dot{s}_u^* = 0$ (see (9)). This task is satisfied, since plugging $\mathbf{u} = 0$ into (10), yields precisely $\dot{s}_u = 0 = \dot{s}_u^*$.

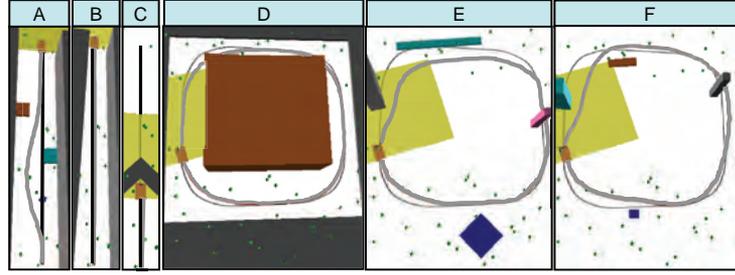


Fig. 6. Six obstacle scenarios for replaying two taught paths (black) with the robot (rectangle): a straight segment (scenarios A to C), and a closed-loop followed in the clockwise sense (D to F). Visual features are represented by the spheres, the occupancy grid by the rectangular area, and the replayed paths are drawn in grey.

6. If we tune the depth Z_c to infinity in (15), $j_v = 0$, and $j_\varphi = j_\omega = 1 + x^2$. Thus, control law (21) becomes

$$\begin{cases} v = (1 - H)v_s + Hv_u \\ \omega = (1 - H)\lambda_x \frac{x^d - x}{j_\omega} + (1 - H)\lambda_\varphi \varphi + H\kappa_b v_u \\ \dot{\varphi} = H\lambda_x \frac{x^d - x}{j_\omega} - (1 - H)\lambda_\varphi \varphi + H\kappa_b v_u. \end{cases}$$

Note that, for small image error ($x \approx x^d$), $\dot{\varphi} \approx -\omega$. In practice, the robot rotation is compensated for by the camera pan rotation, which is an expected behaviour.

5. Simulations

In this section and in the following, we provide details of the simulated and real experiments that were used to validate our approach. Simulations are in the video shown in Extension 1.

For simulations, we made use of Webots,⁴ where we designed a car-like robot equipped with an actuated 320×240 pixels 70° field of view camera, and with a 110° scanner of range 15 m. Both sensors operate at 30 Hz. The visual features, represented by spheres, are distributed randomly in the environment, with depths with respect to the robot varying from 0.1 to 100 m. The offset between R and C is $X^C = 0.7$ m, and we set $Z_c = 15$ m that meets the condition $Z_c > X^C/2$. We use 21 tentacles, with $\kappa_M = 0.35$ m^{-1} (the robot maximum applicable curvature). For the situation risk function, we use $\Delta_s = 6$ m and $\Delta_d = 4.5$ m. These parameters correspond to the design of H shown in Figure 3. The safe translational velocity is designed with $v_m = 0.4$ m s^{-1} , $v_M = 1$ m s^{-1} , $\kappa_\omega = 13$ and $\kappa_\varphi = 3$, as in the graph on the left-hand side of Figure 5. For the unsafe translational velocity, we use $\delta_s = 5$ m and $\delta_d = 2.7$ m as on the right-hand side of Figure 5 (top curve). The simulations were helpful for tuning the control gains, in all experiments, to $\lambda_x = 1$ and $\lambda_\varphi = 0.5$.

At first, no obstacle is present in the environment, and the robot is driven along a taught path. Then, up to five obstacles are located, near and on the taught path, and the robot must replay the visual path, while avoiding them. In addition, the obstacles may partially occlude the features.

Although the sensors are noise-free, and feature matching is ideal, these simulations allow validation of controller (21).

By displacing the obstacles, we have designed the six scenarios shown in Figure 6. For scenarios A, B and C, the robot has been previously driven along a 30 m straight path, and $N = 8$ key images have been acquired, whereas in scenarios D, E and F, the taught path is a closed loop of length 75 m and $N = 20$ key images, which is followed in the clockwise sense. In all scenarios, the robot is able to navigate without colliding, and this is done with the same parameters. The metrics used to assess the experiments are the image error with respect to the visual database $x - x^d$ (in pixels), averaged over the whole experiment and denoted \bar{e} , and the distance, at the end of the experiment, from the final 3D key pose (ϵ , in centimetres). The first metric \bar{e} is useful to assess the controller accuracy in realizing the visual path following task. The latter metric is less relevant, since task (3) is defined in the image space, and not in the pose space.

In all six scenarios, path following has been achieved, and in some cases, navigation was completed using only three image points. Obviously, this is possible in simulations, since feature detection is ideal: in the real case, which includes noise, three points may be insufficient. Some portions of the replayed paths, corresponding to the obstacle locations, are far from the taught ones. However, these deviations would have been indispensable to avoid collisions, even with a pose-based control scheme. Let us detail the robot behaviour in the six scenarios:

- Scenario A: two walls, which were not present during teaching, are parallel to the path, and three boxes are placed in between. The first box is detected, and overtaken on the left. Then, the vehicle passes between the second box and the left wall, and then overtakes the third box on the right. Finally, the robot converges back to the path, and completes it. Although the walls occlude features on the sides, the experiment is successful, with $\bar{e} = 5$, and $\epsilon = 23$.
- Scenario B: it is similar to scenario A, except that there are no boxes, and that the left wall is shifted towards the right one, making the passage narrower towards the end. This makes the robot deviate in order to pass in the centre of the passage. We obtain $\bar{e} = 6$, $\epsilon = 18$.

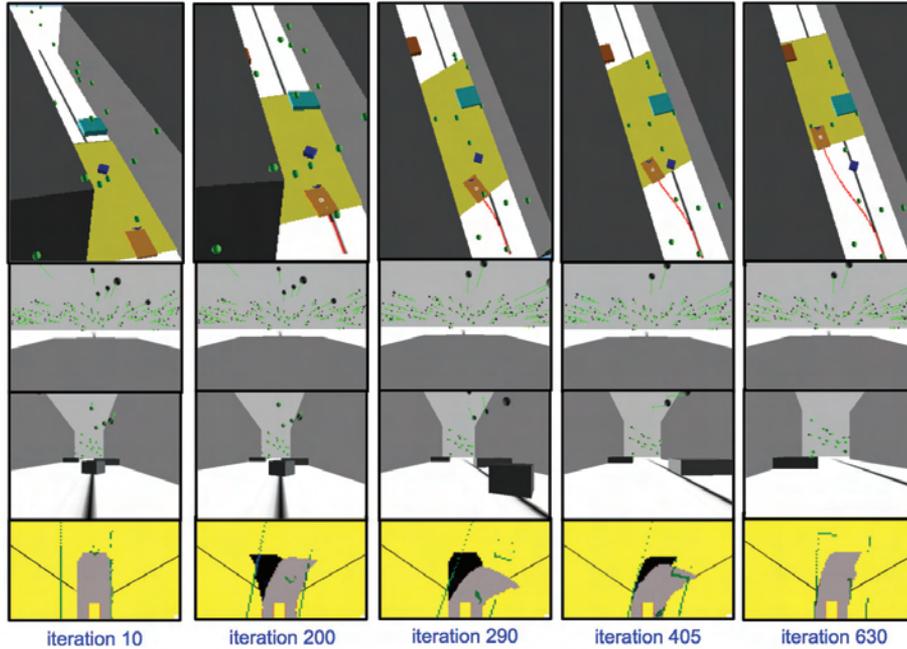


Fig. 7. Scenario A. For each of the five relevant iterations we show (top to bottom) the robot overtaking the first obstacle, the next key image, the current image and the occupancy grid.

- Scenario C: this scenario is designed to test the controller in the presence of unavoidable obstacles. Two walls forming a corner, are located on the path. This soon makes all tentacles unsafe: $\Delta_j \leq \Delta_d \forall j$, yielding $H = 1$. In addition, as the robot approaches the wall, the collision distance on the best tentacle δ_b decreases, and eventually becomes smaller than δ_d , to make $v_u = 0$ and stop the robot (see (23)). Although the path is not completed (making metric ϵ irrelevant), the collision is avoided, and $\bar{e} = 4$ pixels. As proved in Section 4, convergence of the visual task ($x = x^d$) is achieved, in spite of $\mathbf{u} = 0$. In particular, here, the centroid abscissa on the third key image in the database is reached.
- Scenario D: high walls are present on both sides of the path; this leads to important occlusions (less than 50% of the database features are visible), and to a consequent drift from the taught path. Nevertheless, the final key image is reached, without collisions, and with $\bar{e} = 34$ and $\epsilon = 142$. Although this metric is higher than in the previous scenarios (since the path is longer and there are numerous occlusions), it is still reasonably low.
- Scenario E: two obstacles are located on the path, and two others are near the path. The first obstacle is overtaken on the left, before avoiding the second one, also on the left. Then, the robot converges to the path and avoids the third obstacle on the right, before reaching the final key image. We obtain $\bar{e} = 33$ and $\epsilon = 74$. The experiment shows one of the advantages of our tentacle-based approach: lateral data in the grid is ignored (considering the fourth obstacle, would have made the robot curve away from the path).
- Scenario F: here, the controller is assessed in a situation where classical obstacle avoidance strategies (e.g. potential fields) often fail because of local minima. In fact, when detected, the first obstacle is centred on the X axis and orthogonal to it. This may induce an ambiguity, since occupancies are symmetric with respect to X . However, the visual features distribution, and consequent visual task tentacle κ_n drive the robot to the right of the obstacle, which is thus avoided. We have repeated this experiment with 10 randomly generated visual feature distributions, and in all cases the robot avoided the obstacle. The scenario involves four more obstacles, two of which are circumnavigated externally, and two on the inside. Here, $\bar{e} = 29$ and $\epsilon = 75$.

In Figure 7, we show five stages of scenario A. In this figure, as well as later in Figures 12 and 14, the segments linking the current and next key image points are drawn in the current image. In the occupancy grid, the dangerous cell sets associated with the visual task tentacle and to the best tentacle (when different) are respectively shown in grey and black, and two black segments indicate the scanner amplitude. Only cells that can activate H (i.e. cells at distance $\Delta < \Delta_s$) have been drawn. At the beginning of the experiment (iteration 10), the visual features are driving the robot towards the first obstacle. When it is near enough, the obstacle triggers H (iteration 200), forcing the robot away from the path towards the best tentacle, while the camera rotates clockwise to maintain feature visibility (iterations

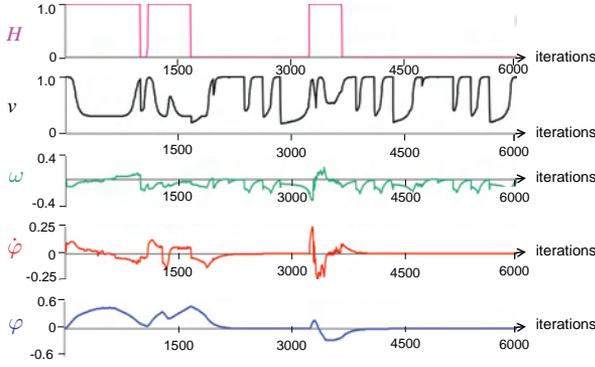


Fig. 8. Evolution of relevant variables in scenario E; top to bottom: H , v (in m s^{-1}), ω (in rad s^{-1}), $\dot{\phi}$ (in rad s^{-1}) and ϕ (in rad).

200, 290 and 405). Finally (iteration 630), the controller drives the robot back to the path, and the camera to the forward-looking direction.

Further details can be obtained by studying some relevant variables. We focus on scenario E, for which we have plotted in Figure 8 the values of H , v , ω , $\dot{\phi}$ and ϕ during navigation. The curve of H shows when the obstacles, respectively the first (at iterations 0–1,000), second (1,100–1,700) and third (3,200–3,700), have intervened in control law (21). As mentioned previously, the fourth obstacle does not trigger H , since it is too far to the side to jeopardize the robot. Let us now discuss the trend of the five curves. Since the beginning, the first obstacle is detected: the tentacle selection induces a negative rotation on the robot (ω curve), a positive one on the camera ($\dot{\phi}$), and a reduction of v . The strategy proves efficient, since the robot overtakes the obstacle. Soon afterwards, the second obstacle triggers H , and provokes a deceleration on v . Concurrently, the camera pan angle ϕ becomes positive to track the visual features which are mostly on the left of the robot (just like the taught path, as shown in Figure 6). When the second obstacle is bypassed, the camera pan is reset to zero. The reduction of v at iteration 2,800 is due only to the sharp turn (i.e. to the reduction of v_s), since the path is safe at this point. Then, the third obstacle triggers H , and is easily circumnavigated. From iteration 3,700 onwards, the situation risk function is cancelled. Correspondingly, the variables are driven by (22). Note also that the camera angle ϕ is reset to 0 in less than 200 iterations, and remains null until the end of the experiment. The small spikes in the angular velocity ω , which appear throughout the experiment, correspond to the changes of database key images (except when they are provoked by the obstacles, as discussed above).

The six simulations have been repeated by setting the feature depth Z_c to infinity. For all six scenarios, the image accuracy, assessed with \bar{e} , is very near to that obtained when $Z_c = 15$ m. On the other hand, the pose accuracy, assessed with ϵ , is lower when $Z_c = \infty$, as shown in Table 1. The difference is relevant on long paths (scenarios D, E

Table 1. Final 3D error ϵ (in centimetres) when $Z_c = 15$ m and $Z_c = \infty$ (for scenario C, since the path is not completed, ϵ is irrelevant).

Scenario	A	B	D	E	F
$Z_c = 15$ m	23	18	142	74	75
$Z_c = \infty$	26	19	151	80	82

and F). Although the navigation task is defined in the image space, these experiments show that tuning Z_c , even coarsely, according to the environment, can contribute to the controller performance in the 3D space. This aspect had already emerged in part in our previous work (Cherubini et al., 2009).

6. Real experiments

After the simulations, the framework has been ported on our CyCab vehicle, set in car-like mode (i.e. using the front wheels for steering), for real outdoor experimental validation. The robot is equipped with a coarsely calibrated 320×240 pixels 70° field of view, B&W Marlin (F-131B) camera mounted on a TRACLabs Biclops Pan/Tilt head (the tilt angle is null, to keep the optical axis parallel to the ground), and with a two-layer, 110° scanning angle, laser SICK LD-MRS. A dedicated encoder on the TRACLabs head precisely measures the pan angle ϕ required in our control law (see (21)). The grid is built by projecting the laser readings from the two layers on the ground. Exactly the same configuration (i.e. the same parameters, gains and grid size) tuned in Webots is used on the real robot. The centroid depth value that we used in simulations ($Z_c = 15$ m) proved effective in all real experiments as well, although the scenarios were very variegated. This confirms, as shown in Cherubini et al. (2009), that a very coarse approximation of the scene depth is sufficient to effectively tune Z_c . The velocity ($v_M = 1$, as in Webots) has been reduced due to the image processing rate (10 Hz), to limit the motion of features between successive images; the maximum speed attainable by the CyCab is 1.3 m s^{-1} anyway. Since camera (10 Hz) and laser (12.5 Hz) processing are not synchronized, they are implemented on two different threads, and the control input \mathbf{u} derived from control law (21) is sent to the robot as soon as the visual information is available (10 Hz).

It is noteworthy to point out that the number of tentacles that must be processed, and correspondingly the computational cost of the laser processing thread, increase with the context danger. For clarity, let us discuss two extreme cases: a safe and an occupied contexts. To verify that a context is safe (i.e. that $H_v = 0$ in (17)), all of the cells in the dangerous areas $\mathcal{D} \cup \mathcal{E}$ of only the two neighbour tentacles must be explored. Instead, in a scenario where the grid is very occupied, all of the tentacles in \mathcal{K} may need to be explored. In general, this second case will be more costly

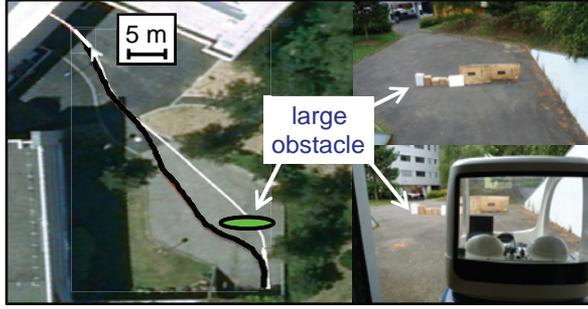


Fig. 9. Scenario A (a long obstacle is avoided): taught (white) and replayed (black) paths.

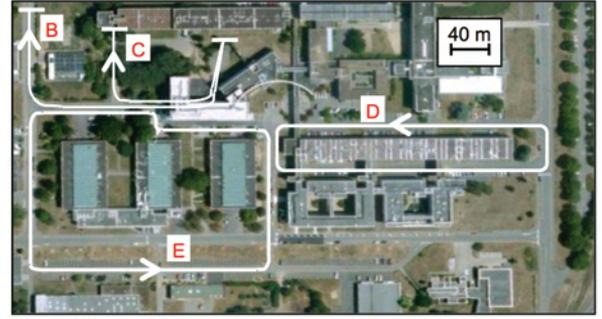


Fig. 11. Map of the four navigation paths B, C, D, E.

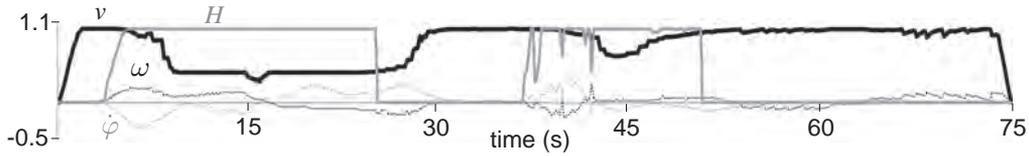


Fig. 10. Control inputs in scenario A: v (solid black, in m s^{-1}), ω (dotted black, rad s^{-1}), $\dot{\phi}$ (dotted grey, rad s^{-1}) and H (solid grey).

than the first. However, in practice, since only the minimum risk and collision distances (Δ_j and δ_b) are required by our controller, exploration of a tentacle stops as soon as the nearest cell is found occupied, so that the tentacles are rarely entirely explored. The experiments showed that the computational cost of laser processing, using the chosen number of tentacles (i.e. 21, as mentioned in Section 2.3), was never a critical issue with respect to that of image processing.

First, we have assessed the performance of our control scheme when a very long obstacle is positioned perpendicularly on the taught path (denoted as path A, and shown in Figure 9). In Figure 10, we have plotted the control inputs \mathbf{u} , and the situation risk function H . The smooth trend of \mathbf{u} at the beginning and end of the experiments is due to the acceleration saturation carried out at the CyCab low-level control. The obstacle is overtaken on the left, while the camera rotates right to maintain scene visibility (dotted black and dotted grey curves in Figure 10). The robot is able to successfully reach the final key image and complete navigation, although it is driven over 5 m away from the taught 3D path by the obstacle. In practice, soon after the obstacle is detected (i.e. after 5 s), tentacles with first positive (5–16 s), and then negative (16–25 s) curvature are selected. Since these tentacles are clear, v is reduced only for visual tracking, by (19) (solid black curve in Figure 10). This is a major feature of the tentacle method, which considers only the actual collision risk of obstacles for reducing the translational velocity. After 25 s, the environment returns clear ($H = 0$), so the visual tentacle can be followed again, and the robot is driven back to the path. Then (38–52 s) a small bush on the left triggers H and causes a small counterclockwise rotation along with a slight decrease in v . Then the context returns safe, and the visual path can be followed

for the rest of the experiment. The translational velocity averaged over the experiment is 0.79 m s^{-1} , which is more than twice the speed reached in Cherubini and Chaumette (2011).

After these results, we have run a series of experiments, on longer and more crowded paths (denoted B to E in Figure 11) on our campus. All campus experiments here are also visible in the video shown in Extension 2. The Cycab was able to complete all paths (including 650 m long path E), while dealing with various natural and unpredictable obstacles, such as parked and driving cars and pedestrians. The translational velocity averaged over these experiment was 0.85 m s^{-1} .

Again, by assessing the collision risk only along the visual path, non-dangerous obstacles (e.g. walls or cars parked on the sides) are not taken into account. This aspect is clear from Figure 12(left), where a stage of the navigation experiment on path E is illustrated. From top to bottom, we show the next key image in the database I_d , the current image I and three consecutive occupancy grids processed at that point of the experiment. As the snapshots illustrate, the cars parked on the right (which were not present during teaching) do not belong to any of the visual task tentacle classification areas. Hence, they are considered irrelevant, and do not deviate the robot from the path.

Another nice behaviour is shown in Figure 12(centre): if a stationary car is unavoidable, the robot decelerates and stops with (20), but, as soon as the car departs, the robot gradually accelerates (again with (20)), to resume navigation. In fact, as we mentioned in Section 4, when the best tentacle is clear up to distance δ_s , a high velocity can be applied: $v = v_s$, independently of the value of H . In the future, this feature of our approach could even be utilized for vehicle following.

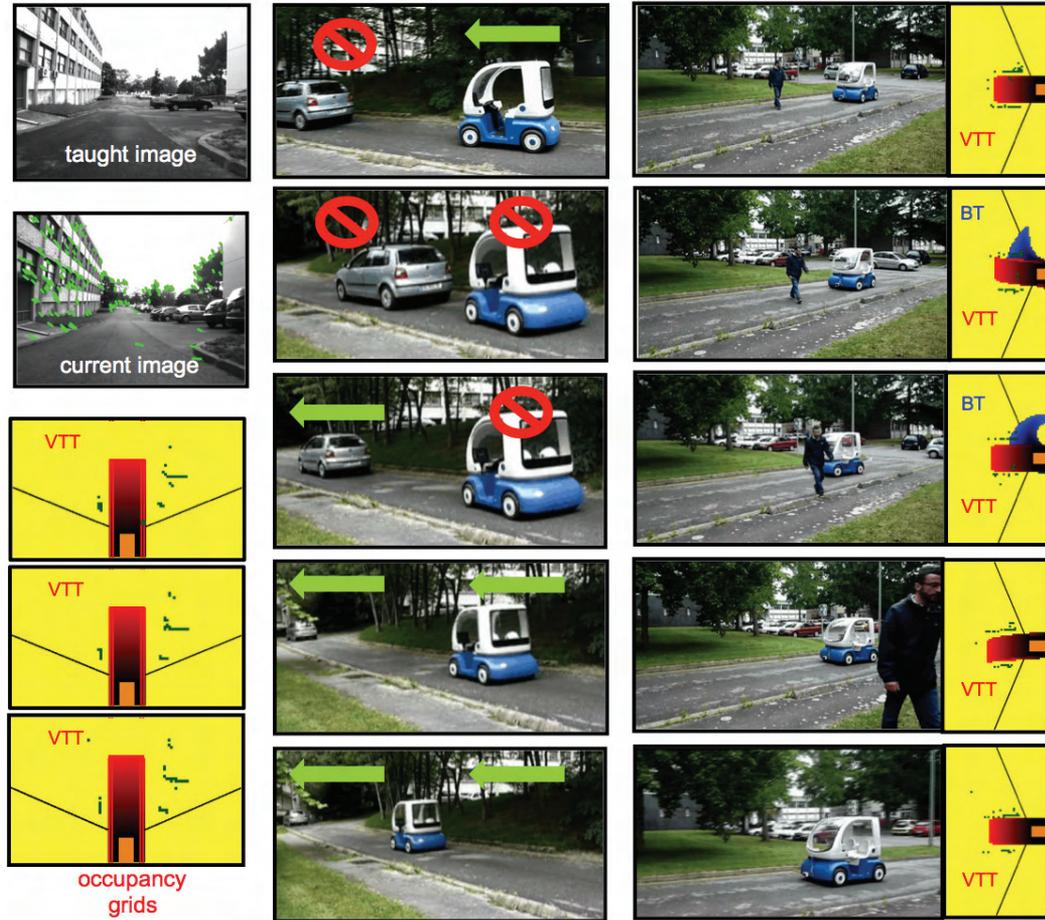


Fig. 12. Validation with: irrelevant obstacles (left), traffic (centre) and a moving pedestrian (right). The visual task tentacle and best tentacle (when different) are respectively indicated as VTT and BT in the occupancy grids.

An experiment with a crossing pedestrian is presented in Figure 12(right). The pedestrian is considered irrelevant, until it enters the visual task tentacle (second image). Then, the clockwise tentacles are selected to pass between the person and the right sidewalk. When the path is clear again, the robot returns to the visual task tentacle, which is first counterclockwise (fourth image) and then straight (fifth image).

In October 2011, as part of the final demonstration of the French ANR project CityVIP, we have validated our framework in a urban context, in the city of Clermont Ferrand. The experiments have taken place in the crowded neighbourhood of the central square Place de Jaude, shown in Figure 13. For four entire days, our Cycab has navigated autonomously, continuously replaying a set of visual paths of up to 700 m each, amidst a variety of unpredictable obstacles, such as cars, pedestrians, bicycles and scooters. In Figure 14, we show some significant snapshots of the experiments that were carried out in Clermont Ferrand. These include photos of the Cycab, as well as images acquired by the on-board camera during autonomous navigation. These experiments are also visible in the video shown in Extension 3.

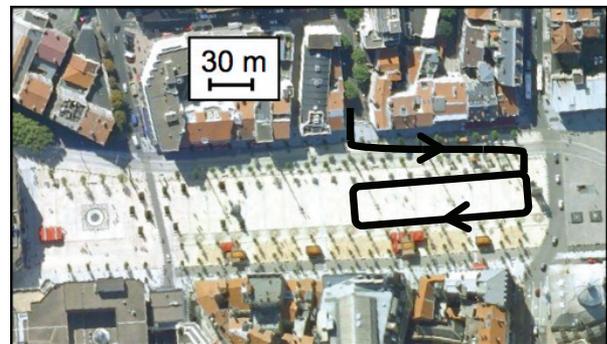


Fig. 13. City centre of Clermont Ferrand, with one of the navigation paths where the urban experiments have been carried out.

In Figure 14(a)–(c), Cycab is moving in a busy street, crowded with pedestrians and vehicles. First, in Figure 14(a), we show a typical behaviour adopted for avoiding a crossing pedestrian: here, Cycab brakes as a lady with black skirt crosses the street. The robot would either stop or circumnavigate the person, and in 4 days no one has

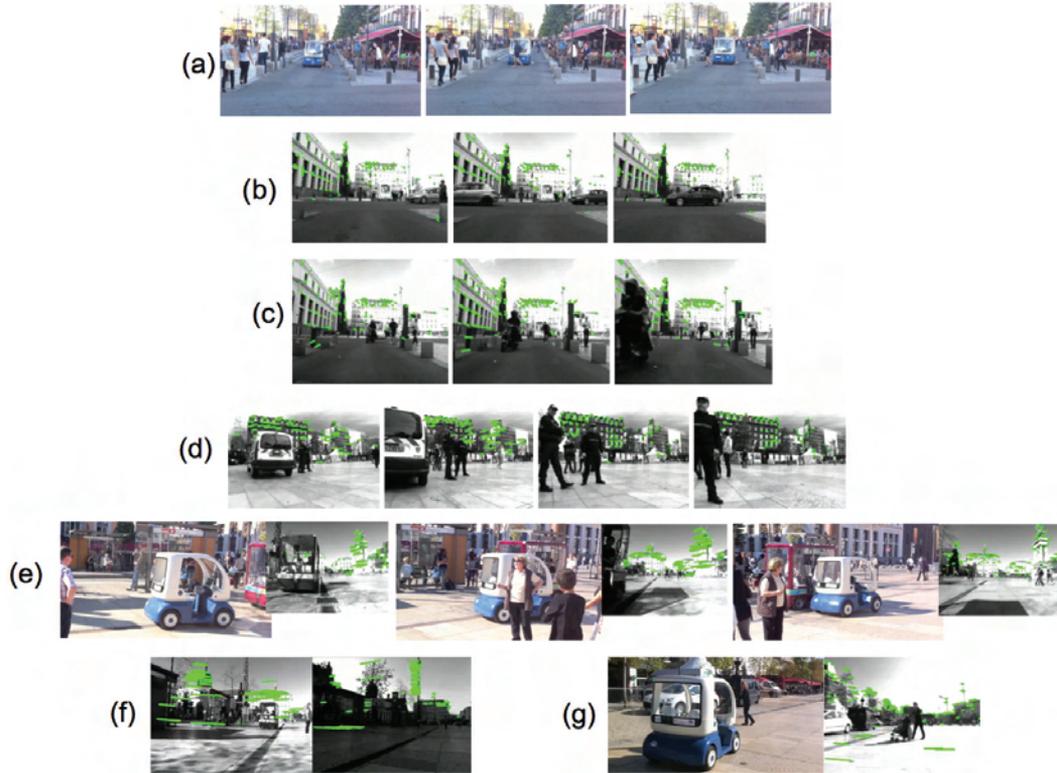


Fig. 14. Snapshots of the urban experiments. (a) Avoiding a crossing pedestrian. (b)–(c) Navigating close to moving cars and to a scooter, respectively. (d)–(e) Avoiding a stationing police patrol and a stationing vehicle, respectively. (f) Navigating with different light conditions, using the same taught database. (g) Avoiding a pedestrian with a baby pushchair.

ever even closely been endangered nor touched by the vehicle. In many experiments, Cycab has navigated among fast moving vehicles (cars in Figure 14(b), and a scooter in Figure 14(c)), and manual security intervention was never necessary. The robot has also successfully avoided many fixed obstacles, including a stationing police patrol (Figure 14(d)) and another electric vehicle (Figure 14(e)). Obviously, when all visual features are occluded by an obstacle or lost, the robot stops.

Moreover, we have thoroughly tested the behaviour of our system with respect to varying light, which is an important aspect in outdoor appearance-based navigation. Varying light has been very common in the extensive Clermont Ferrand experiments, which would last the whole day, from the first light to sunset, both with cloudy and clear sky. In some experiments, we could control the robot in different lighting conditions, using the same taught database. For instance, Figure 14(f) shows two images acquired approximately at the same position at 5 p.m. (top) and 11 a.m. (bottom), while navigating with the same key images. However, in spite of the robustness of the image processing algorithms, which has been proved by Royer et al. (2007), in some cases (e.g. when the camera was overexposed to sunlight), the visual features required for navigation could not be detected. Future work in adapting the camera automatic shutter settings should solve this issue.

In the current version of our framework, moving obstacles are not specifically recognized and modelled. Although, as the experiments show, we are capable of avoiding slowly moving obstacles (e.g. crossing pedestrians or baby pushchairs as in Figure 14(g)), the main objective of our future work will be to directly tackle this problem within the control law, in order to avoid fast obstacles as well. This can be done, for example, by estimating the velocity of the detected objects, and then using it to predict their future position. In our opinion, the main difficulty, in comparison with the case of static obstacles, will concern the accuracy and computation cost of this estimation process.

Overall, Cycab has navigated using an average of approximately 60 visual points on each image, and some paths have even been completed using less than 30 points. Along with all of the cited technical aspects, the experiments highlighted the reactions of non-robotic persons to the use of autonomous ground vehicles in everyday life. Most passersby had not been informed of the experiments, and responded with curiosity, surprise, enthusiasm and, rarely, slight apprehension.

7. Conclusions

A framework with simultaneous obstacle avoidance and outdoor vision-based navigation, without any 3D model

or path planning has been presented. It merges a novel, reactive, tentacle-based technique with visual servoing, to guarantee path following, obstacle bypassing, and collision avoidance by deceleration. Since our method is purely sensor-based and pose-independent, it is perfectly suited for visual navigation. Extensive outdoor experiments, even in urban environments, show that it can be applied in realistic and challenging situations including moving obstacles (e.g. cars and pedestrians). To the best of the authors' knowledge, this is the first time that outdoor visual navigation with obstacle avoidance has been carried out in urban environments at approximately 1 m s^{-1} on over 500 m, using neither GPS nor maps. In the near future, we plan to explicitly take into account the velocity of moving obstacles within our controller, in order to avoid fast obstacles, which are currently hard to deal with. Perspective work also includes automatic prevention of the visual occlusions provoked by the obstacles.

Notes

1. See <http://www.irisa.fr/lagadic/demo/demo-cycab-vis-navigation/vis-navigation>.
2. Without loss of generality, we consider that intervals are defined even when the first endpoint is greater than the second, e.g. $[\kappa_n, \kappa_{nn})$ should be read $(\kappa_{nn}, \kappa_n]$ if $\kappa_n > \kappa_{nn}$.
3. At the first iteration, we set $\kappa_{pb} = \kappa_n$.
4. See <http://www.cyberbotics.com>.

Funding

This work has been supported by ANR (French National Agency) CityVIP project (grant number ANR-07 TSFA-013-01).

References

- Allibert G, Courtial E and Touré Y (2008) Real-time visual predictive controller for image-based trajectory tracking of a mobile robot. In *International Federation of Automatic Control World Congress*, Seoul, Korea.
- Becerra HM, Courbon J, Mezouar Y and Sagüés C (2010) Wheeled mobile robots navigation from a visual memory using wide field of view cameras. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan.
- Becerra HM, López-Nicolás G and Sagüés C (2011) A sliding-mode-control law for mobile robots based on epipolar visual servoing from three views. *IEEE Transactions on Robotics* 27: 175–183.
- Bonin-Font F, Ortiz A and Oliver G (2008) Visual navigation for mobile robots: a survey. *Journal of Intelligent and Robotic Systems* 53: 263–296.
- Bonnafous D, Lacroix S and Siméon T (2001) Motion generation for a rover on rough terrains. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Maui, HI.
- Booi O, Terwijn B, Zivkovic Z and Kröse B (2007) Navigation using an appearance based topological map. In *IEEE International Conference on Robotics and Automation*, Rome, Italy.
- Broggi A, Bombini L, Cattani S, Cerri P and Fedriga RI (2010) Sensing requirements for a 13000 km intercontinental autonomous drive. In *IEEE Intelligent Vehicles Symposium*, San Diego, CA.
- Buehler M, Lagnemma K and Singh S (eds) (2008) Special Issue on the 2007 DARPA Urban Challenge, Parts I–III. *Journal of Field Robotics* 25(8–10): 423–860.
- Chaumette F and Hutchinson S (2006) Visual servo control, Part I: Basic approaches. *IEEE Robotics and Automation Magazine* 13(4): 82–90.
- Chaumette F and Hutchinson S (2007) Visual servo control, Part II: Advanced approaches. *IEEE Robotics and Automation Magazine* 14(1): 109–118.
- Cherubini A and Chaumette F (2011) Visual navigation with obstacle avoidance. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, CA.
- Cherubini A, Colafrancesco M, Oriolo G, Freda L and Chaumette F (2009) Comparing appearance-based controllers for nonholonomic navigation from a visual memory. In *ICRA Workshop on safe navigation in open and dynamic environments*, Kobe, Japan.
- Cherubini A, Spindler F and Chaumette F (2012) A new tentacles-based technique for avoiding obstacles during visual navigation. In *IEEE International Conference on Robotics and Automation*, St. Paul, MN.
- Courbon J, Mezouar Y and Martinet P (2009) Autonomous navigation of vehicles from a visual memory using a generic camera model. *IEEE Transactions on Intelligent Transportation Systems* 10: 392–402.
- Davison AJ, Reid ID, Molton ND and Stasse O (2007) MonoSLAM: real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29: 1052–1067.
- De Luca A and Oriolo G (1994) Local incremental planning for nonholonomic mobile robots. In *IEEE International Conference on Robotics and Automation*, San Diego, CA.
- De Luca A, Oriolo G and Robuffo Giordano P (2008) Feature depth observation for image-based visual servoing: theory and experiments. *The International Journal of Robotics Research* 27: 1093–1116.
- Diosi A, Segvic S, Remazeilles A and Chaumette F (2011) Experimental evaluation of autonomous driving based on visual memory and image based visual servoing. *IEEE Transactions on Intelligent Transportation Systems* 12: 870–883.
- Durand Petiteville A, Courdresses M and Cadenat V (2010) A new predictor/corrector pair to estimate the visual features depth during a vision-based navigation task in an unknown environment. In *International Conference on Informatics in Control, Automation and Robotics*, Rome, Italy.
- Elfes A (1989) Using occupancy grids for mobile robot perception and navigation. *Computer* 22(6): 46–57.
- Folio D and Cadenat V (2006) A redundancy-based scheme to perform safe vision-based tasks amidst obstacles. In *IEEE International Conference on Robotics and Biomimetics*, Kunming, China.
- Fontanelli D, Danesi A, Belo FAW, Salaris P and Bicchi A (2009) Visual servoing in the large. *The International Journal of Robotics Research* 28: 802–814.
- Goedemé T, Nuttin M, Tuytelaars T and Van Gool L (2007) Omnidirectional vision based topological navigation. *International Journal of Computer Vision* 74: 219–236.

- Guerrero JJ, Murillo AC and Sagüés C (2008) Localization and Matching using the planar trifocal tensor with bearing-only data. *IEEE Transactions on Robotics* 24: 494–501.
- Harris C and Stephens M (1988) A combined corner and edge detector. In *4th Alvey Vision Conference*.
- Kato T, Ninomiya Y and Masaki I (2002) An obstacle detection method by fusion of radar and motion stereo. *IEEE Transactions on Intelligent Transportation Systems* 3: 182–188.
- Khatib O (1985) Real-time obstacle avoidance for manipulators and mobile robots. In *IEEE International Conference on Robotics and Automation*, Durham, UK.
- Lamiroux F, Bonnafous D and Lefebvre O (2004) Reactive path deformation for nonholonomic mobile robots. *IEEE Transactions on Robotics* 20: 967–977.
- Lapierre L, Zapata R and Lepinay P (2007) Simultaneous path following and obstacle avoidance control of a unicycle-type robot. In *IEEE International Conference on Robotics and Automation*, Rome, Italy.
- Latombe JC (1991) *Robot Motion Planning*. Dordrecht: Kluwer Academic.
- Lee T-S, Eoh G-H, Kim J and Lee B-H (2010) Mobile robot navigation with reactive free space estimation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan.
- López-Nicolás G, Gans NR, Bhattacharya S, Sagüés C, Guerrero JJ and Hutchinson S (2010) An optimal homography-based control scheme for mobile robots with nonholonomic and field-of-view constraints. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 40: 1115–1127.
- López-Nicolás G and Sagüés C (2011) Vision-based exponential stabilization of mobile robots. *Autonomous Robots* 30: 293–306.
- Lowe DG (2004) Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 60: 91–110.
- Mariottini GL, Oriolo G and Prattichizzo D (2007) Image-based visual servoing for nonholonomic mobile robots using epipolar geometry. *IEEE Transactions on Robotics* 23: 87–100.
- Minguez J, Lamiroux F and Laumond JP (2008) Motion planning and obstacle avoidance. In Siciliano B and Khatib O (eds.), *Springer Handbook of Robotics*. Berlin: Springer, pp. 827–852.
- Nunes U, Laugier C and Trivedi M (2009) Introducing perception, planning, and navigation for intelligent vehicles. *IEEE Transactions on Intelligent Transportation Systems* 10: 375–379.
- Ohya A, Kosaka A and Kak A (1998) Vision-based navigation by a mobile robot with obstacle avoidance using a single-camera vision and ultrasonic sensing. *IEEE Transactions on Robotics and Automation* 14: 969–978.
- Quinlan S and Khatib O (1993) Elastic bands: connecting path planning and control. In *IEEE International Conference on Robotics and Automation*, Atlanta, GA.
- Royer E, Lhuillier M, Dhome M and Lavest J-M (2007) Monocular vision for mobile robot localization and autonomous navigation. *International Journal of Computer Vision* 74: 237–260.
- Scaramuzza D and Siegwart R (2008) Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles. *IEEE Transactions on Robotics* 24: 1015–1026.
- Sciavicco L and Siciliano B (2000) *Modeling and Control of Robot Manipulators*. New York: Springer.
- Šegvić S, Remazeilles A, Diosi A and Chaumette F (2008) A mapping and localization framework for scalable appearance-based navigation. *Computer Vision and Image Understanding* 113: 172–187.
- Shi J and Tomasi C (1994) Good features to track. In *IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, WA.
- Tahri O and Chaumette F (2005) Point-based and region-based image moments for visual servoing of planar objects. *IEEE Transactions on Robotics* 21: 1116–1127.
- Von Hundelshausen F, Himmelsbach M, Hecker F, Mueller A and Wuensche H-J (2008) Driving with tentacles-Integral structures of sensing and motion. *Journal of Field Robotics* 25: 640–673.
- Wada T, Doi S and Hiraoka S (2009) A deceleration control method of automobile for collision avoidance based on driver's perceptual risk. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*.
- Yan Z, Xiaodong X, Xuejun P and Wei W (2003) Mobile robot indoor navigation using laser range finder and monocular vision. In *IEEE International Conference on Intelligent Systems and Signal Processing*, Faro, Portugal.
- Zhang Y, Lin WC and Chin Y-KS (2008) Driving skill characterization: a feasibility study. In *IEEE International Conference on Robotics and Automation*, Pasadena, CA.
- Zhang AM and Kleeman L (2009) Robust appearance based visual route following for navigation in large-scale outdoor environments. *The International Journal of Robotics Research* 28: 331–356.

Appendix: Index to Multimedia Extensions

The multimedia extension page is found at <http://www.ijrr.org>

Table of Multimedia Extensions

Extension	Type	Description
1	Video	Simulations of our navigation scheme in six different scenarios. In this and the following two videos, the segments linking the current and next key image points are drawn in light green in the current image. In the occupancy grid, the dangerous cell sets associated with the visual task tentacle and to the best tentacle (when different) are respectively shown in red and blue, and two black segments indicate the scanner amplitude. Only cells that can activate H (i.e. cells at distance $\Delta < \Delta_s$) have been drawn.
2	Video	Experiments on our campus.
3	Video	Extensive validation in the city centre of Clermont Ferrand.

Autonomous Visual Navigation and Laser-based Moving Obstacle Avoidance

Andrea Cherubini, *Member, IEEE*, Fabien Spindler, and François Chaumette, *Fellow, IEEE*

Abstract—Moving obstacle avoidance is a fundamental requirement for any robot operating in real environments, where pedestrians, bicycles and cars are present. In this paper, we propose and validate a framework for avoiding moving obstacles during visual navigation with a wheeled mobile robot. Visual navigation consists of following a path, represented as an ordered set of key images, which have been acquired by an on-board camera in a teaching phase. While following such path, our robot is able to avoid static as well as moving obstacles, which were not present during teaching, and which are sensed by an on-board lidar. The proposed approach takes explicitly into account obstacle velocities, estimated using an appropriate Kalman-based observer. The velocities are then used to predict the obstacle positions within a tentacle-based approach. Finally, our approach is validated in a series of real outdoor experiments, showing that when the obstacle velocities are considered, the robot behaviour is safer, smoother, and faster than when it is not.

Index Terms—Visual Servoing, Visual Navigation, Collision Avoidance.

I. INTRODUCTION

One of the main objectives of recent robotics research is the development of vehicles capable of autonomously navigating in unknown environments [3], [4], [5]. The success of the DARPA Urban Challenges [6] has heightened expectations that autonomous cars will soon be able to operate in environments of realistic complexity. In this field, information from visual sensors is often used for localization [7], [8] or for navigation purposes [9], [10], [11].

Nevertheless, a critical issue for successful navigation remains motion safety, especially when the robot size and dynamics make it potentially harmful. Hence, an important task is obstacle avoidance, which consists of either generating a collision-free trajectory to the goal, or of decelerating to prevent collision whenever bypassing is impossible [12]. Obstacle avoidance has traditionally been handled by two techniques [13]: the deliberative approach [14], usually consisting of a motion planner, and the reactive approach [15], based on the instantaneous sensed information.

The task that we focus on is outdoor visual navigation: a wheeled vehicle, equipped with an actuated pinhole camera and with a forward-looking lidar, must follow a path represented by key images, without colliding. In the past, obstacle avoidance has been integrated in visual navigation [16], [17]

and path following [18], [19], by using the path geometry or the environment 3D model (including, for example, walls and doors). However, since our task is defined in the image space, we seek a reactive (i.e., merely sensor-based) solution, which does not need a global model of the environment and trajectory. Moreover, the proposed solution copes with moving obstacles, which are common in dynamic, real-world environments.

Reactive strategies include: the vector field histogram [20], obstacle-restriction method [21], and closest gap [22]. In our previous work [2], we presented a novel sensor-based method guaranteeing that obstacle avoidance had no effect on visual navigation. However, in that work, we did not consider moving obstacles. In the literature, researchers have taken into account the obstacle velocities to deal with this issue. We hereby survey the main papers dedicated to moving obstacle avoidance.

The approach presented in [23] is one of the first where static and moving obstacles are avoided, based on their current positions and velocities relative to the robot. The maneuvers are generated by selecting robot velocities outside of the *velocity obstacles*, that would provoke a collision at some future time. Planning in the velocity space makes it possible to consider the robot dynamics. This paradigm has been adapted in [24] to the car-like robot kinematic model, and extended in [25] to take into account unpredictably moving obstacles. This has been done by using reachability sets to find matching constraints in the velocity space, called Velocity Obstacle Sets.

Another pioneer method that has inspired many others is the Dynamic Window [26], that is derived directly from the dynamics of the robot, and is especially designed to deal with constrained velocities and accelerations. The method consists of two steps: first, a valid subset of the control space is generated, and then an optimal solution (driving the robot with maximum obstacle clearance) is sought within it. A generalization of the dynamic map that accounts for moving obstacle velocities and shapes is presented in [27], where a union of polygonal zones corresponding to the non admissible velocities controls the robot, and prevents collisions. In [28], the Dynamic Window has been integrated in a graph search algorithm for path planning, to drive the robot trajectories within a global path. A planning approach is also used in [29], where the likelihood of obstacle positions is input to a Rapidly-exploring Random Tree algorithm.

In [30], motion safety is characterized by three criteria, respectively related to the model of the robotic system, to the model of the environment and to the decision making process. The author proves that motion safety cannot be guaranteed in the presence of moving objects (i.e., the robot may inevitably

A. Cherubini is with the Laboratory for Computer Science, Microelectronics and Robotics LIRMM - Université de Montpellier 2 CNRS, 161 Rue Ada, 34392 Montpellier, France. {Andrea.Cherubini}@lirmm.fr.

F. Spindler and F. Chaumette are with Inria Rennes - Bretagne Atlantique, IRISA. {firstname.lastname}@inria.fr.

This paper has been published in part in [1] and in [2].

Manuscript received XXX; revised XXX.

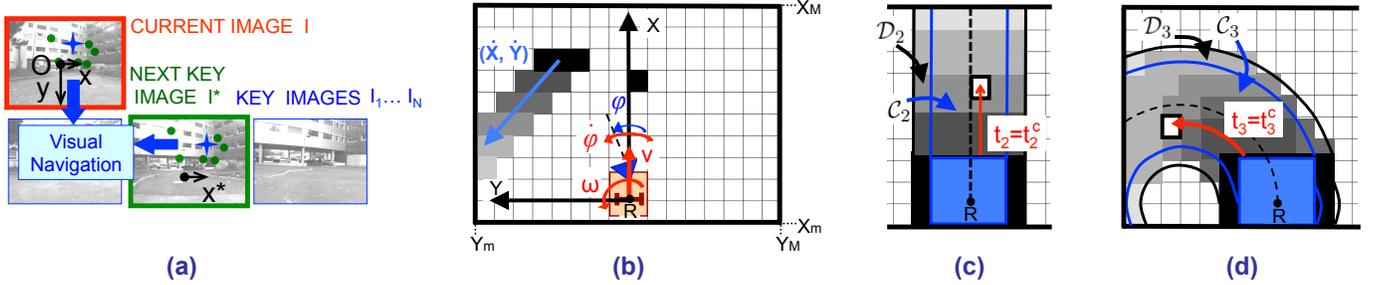


Fig. 1. (a) Current and next key images, and key image database. (b) Top view of the robot, with actuated camera and control inputs v , ω and ϕ . A static (right) and moving (left) object are observed (black); we show the object velocity (\dot{X}, \dot{Y}) , and future occupied cells c_i in grey, increasingly light with t_{i0} . (c, d) Tentacles (dashed black), with classification areas, corresponding boxes and delimiting arcs of circle, and cells $c_i \in \mathcal{D}_j$ displayed in grey, increasingly light with increasing t_{ij} .

collide at some time in the future). More recently [31], the same researchers have defined the Braking Inevitable Collision States (BICS) as states such that, whatever the future braking trajectory, a collision will occur. In that paper, the BICS are used to achieve passive motion safety.

Although all these approaches have proved effective, none of them deals with moving obstacle avoidance during visual navigation. In [2], we presented a framework that guarantees that obstacle avoidance has no effect on the visual task. In the present paper, we further improve that framework, by designing a reactive approach that can deal with moving obstacles as well. Our approach is based on tentacles [32], i.e. candidate trajectories (arcs of circles) that are evaluated during navigation, both for assessing the context, and for designing the task in case of danger. The main contribution of this paper is the improvement of that framework, to take into account the obstacle velocities. First, we have designed a Kalman-based observer for estimating the obstacle velocities. Then, we have adapted the tentacles designed in [2], to effectively take into account these velocities. Finally, we experimentally validate our approach in a series of experiments.

The article is organized as follows. In Section II, all the relevant variables are defined. In Section III and IV, we explain respectively how the obstacle velocities are estimated, and how they are used to predict possible collisions. Then, in Section V, the control law from [2] is recalled, and adapted to deal with moving obstacles. Experimental results are reported in Section VI, and summarized in Section VII.

II. PROBLEM DEFINITION

This section is, in part, taken from [2]. Referring to Fig. 1(a, b), we define the robot frame $\mathcal{F}_{\mathcal{R}}(R, X, Y)$ (R is the robot center of rotation) and image frame $\mathcal{F}_{\mathcal{I}}(O, x, y)$ (O is the image center). The robot control inputs are:

$$\mathbf{u} = [v, \omega, \phi]^T.$$

These are the translational and angular velocities of the vehicle, and the camera pan angular velocity. We use the normalized perspective camera model, and we assume that the sequence of images that defines the path can be tracked with continuous $v(t) > 0$. This ensures safety, since only obstacles in front of the robot can be detected by our lidar.

The path that the robot must follow is represented as a database of ordered key images, such that successive pairs

contain some common static visual features (points). First, the vehicle is manually driven along a *taught* path, with the camera pointing forward ($\phi = 0$), and all the images are saved. Afterwards, a subset (database) of N key images I_1, \dots, I_N representing the path (Fig. 1(a)) is selected. Then, during autonomous navigation, the current image, noted I , is compared with the next key image $I^* \in \{I_1, \dots, I_N\}$, and a relative pose estimation between I and I^* is used to check when the robot passes the pose where I^* was acquired. For key image selection, and visual point detection and tracking, we use the algorithm proposed in [33]. The output of this algorithm, which is used by our controller, is the set of points visible both in I and I^* . Then, navigation consists of driving the robot forward, while I is driven to I^* . We maximize similarity between I and I^* using only the abscissa x of the centroid of points matched on I and I^* to control the robot heading. If no points can be matched between I and I^* , the robot stops. This can typically occur in the presence of an occluding obstacle; however, navigation is resumed as soon as the obstacle moves and the features are again visible. When I^* has been passed, the next image in the set becomes the desired one, and so on, until I_N is reached.

Along with the visual path following problem, we consider obstacles which are on the path, but not in the database, and sensed by the lidar in a plane parallel to the ground. For obstacle modeling, we use the occupancy grid presented in Fig. 1(b): it is linked to $\mathcal{F}_{\mathcal{R}}$, with cell sides parallel to X and Y . Its extension is limited ($X_m \leq X \leq X_M$ and $Y_m \leq Y \leq Y_M$), to ignore obstacles that are too far to jeopardize the robot. An appropriate choice for $|X_m|$ is the length of the robot, since obstacles behind cannot hit it as it advances. Any grid cell $\mathbf{c} = [X, Y]^T$ is considered currently occupied (black in Fig. 1(b)) if an obstacle has been sensed there. For cells lying in the lidar area, only the current lidar reading is considered. For the other cells, we use past readings, displaced with odometry. The set of occupied cells with their estimated velocities, denoted by \mathcal{O} , is used, along with the robot geometric and kinematic characteristics, to derive possible future collisions. This approach is different from the one in [2], where only the *currently* occupied cells in the grid were considered. To estimate the obstacle velocities, and therefore update \mathcal{O} , we have designed an obstacle observer, detailed just below.

III. OBSTACLE OBSERVER

The detection and tracking of obstacles is crucial for collision-free navigation. Of particular interest are potentially dynamic objects (i.e., objects that could move) since their presence and potential change of state will influence the planning of actions and trajectories. Obviously, estimating the velocity of these objects is fundamental.

Compared to areas where known road network information can provide background separation, unknown environments present a more challenging scenario, due to low signal to noise ratio. Recent works [34], [35] have tackled these issues. In [34], classes of interest for autonomous driving (i.e., cars, pedestrians and bicycles) are identified, using shape information and a RANSAC-based edge selection algorithm. The authors of [35] apply a foreground model that incorporates geometric as well as temporal cues; then, moving vehicles are tracked using a particle filter. Both works rely on the Velodyne HDL-64E S2, a laser range finder that provides rich 3D point clouds, to classify moving obstacles. Instead, we target solutions based uniquely on a 2D lidar, and we are not interested in recognizing the object classes.

In practice, we base our work on two assumptions:

- 1) All objects are rigid.
- 2) The instantaneous curvature of the object trajectories (i.e., the ratio between their angular and translational velocities) is small enough to assume that their motion is purely translational over short time intervals. Hence, the translational velocities of all points on an object are identical and equal to that of its centroid.

Both assumptions are plausible for the projection on the ground of walls, most vehicles and even pedestrians.

Under these assumptions, for each object, the state to be estimated will be composed of the coordinates of its centroid in \mathcal{F}_R , and by their derivatives¹:

$$\mathbf{x} = [X, Y, \dot{X}, \dot{Y}]^\top.$$

Using a first-order Markov model (which is plausible for low object accelerations), the state at time t is evolved from the state at $t - \Delta t$ (Δt is the sampling time) according to:

$$\mathbf{x}(t) = \mathbf{F}\mathbf{x}(t - \Delta t) + \mathbf{w}(t), \quad (1)$$

where $\mathbf{w}(t) \sim N(0, \mathbf{Q})$ is assumed to be Gaussian white noise, with covariance \mathbf{Q} and, the state transition model is:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

At time t , an observation $\mathbf{z}(t)$ of the object centroid coordinates is derived from lidar data. It is related to the state by:

$$\mathbf{z}(t) = \mathbf{H}\mathbf{x}(t) + \mathbf{v}(t), \quad (2)$$

¹If assumption 2 is not met, the orientation and angular velocity of the object must be added to the state vector \mathbf{x} , that is to be estimated by our observer.

where $\mathbf{v}(t) \sim N(0, \mathbf{R})$ is assumed to be Gaussian white noise with covariance \mathbf{R} , and the observation model is:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

Let us outline the steps of our recursive algorithm for deriving $\hat{\mathbf{x}}(t)$ (our estimate of $\mathbf{x}(t)$), based on current observations $\mathbf{z}(t)$, and on previous states $\mathbf{x}(t - \Delta t)$.

- 1) At time t , all currently occupied cells in \mathcal{O} are clustered in objects, using a threshold on pairwise cell distance, and the current observation of the centroid coordinates $\mathbf{z}(t)$ is derived for each object.
- 2) All of the object centroids that have been observed at some time in the recent past (we look back in the last $2s$) are displaced by odometry, to derive their coordinates $X(t - \Delta t)$ and $Y(t - \Delta t)$.
- 3) The observed and previous object centroids (outputs of steps 1 and 2) are pairwise matched according to their distance². We then discern between three cases:
 - For matched objects, the previous centroid velocity is obtained by numerical differentiation, and input to a Kalman filter, based on equations (1-2), and on the outputs of step 2, to derive $\hat{\mathbf{x}}(t)$.
 - For unmatched objects currently observed, we set:

$$\hat{\mathbf{x}}(t) = \begin{bmatrix} \mathbf{z}(t) \\ 0 \\ 0 \end{bmatrix}.$$

- For unmatched unobserved objects, the centroid coordinates are memorized (these will be the inputs for step 2).

The output of our algorithm is, at each iteration t , the estimate of the object centroid coordinates and of its velocities:

$$\hat{\mathbf{x}}(t) = [\hat{X}(t), \hat{Y}(t), \hat{\dot{X}}(t), \hat{\dot{Y}}(t)]^\top.$$

Then, each currently occupied cell \mathbf{c}_i is associated to the estimated velocity of the object it belongs to, or to null velocity, if it has not been associated to any object. Set \mathcal{O} is finally formed by all the occupied cell states:

$$\begin{bmatrix} \mathbf{c}_i \\ \dot{\mathbf{c}}_i \end{bmatrix}^\top \in \mathcal{O},$$

that encode the cell current coordinates and velocities in the robot frame.

Let us briefly discuss the particular case of a group of near moving obstacles. If these are close, they are clustered into a single object. If one or more obstacles leave the group, these will be treated as new objects, and their estimated velocity is immediately reset, since there are no matches in the past. The choice of restarting the observer in this case is reasonable, since to leave the group, these obstacles had to substantially vary their velocity.

²High obstacle velocity can hinder this step: if the ratio between the obstacle velocity relative to the vehicle, and the obstacle processing algorithm framerate is strong, the obstacle centroid position in the grid will strongly vary between successive iterations, making matching impossible.

In the next section, we will show how \mathcal{O} is used to predict possible collisions, and accordingly adapt the control strategy. We will assess the danger of each cell by considering the time that the robot will navigate before eventually colliding with it. Without loss of generality, in the next section this time is measured from the current instant t .

IV. OBSTACLE MODELLING

A. Obstacle occupation times

At this stage, the trajectory of each occupied cell in \mathcal{O} can be predicted to evaluate possible collisions with the robot. More concretely, we will just estimate the times at which each cell in the grid will be - eventually - occupied by an obstacle. We assume that velocities of all occupied cells in \mathcal{O} remain constant over time horizon T . This is a plausible assumption, since the estimations of the obstacles positions and velocities are updated at every iteration, by the approach outlined above. Then, for each \mathbf{c}_i that may be occupied by an obstacle within T , we can predict initial

$$t_{i0}(\mathbf{c}_i, \mathcal{O}) \in [0, T]$$

and final

$$t_{if}(\mathbf{c}_i, \mathcal{O}) \in [t_{i0}, T]$$

obstacle occupation times, as a function of the set of occupied cell states \mathcal{O} . For cells occupied by a static object and belonging to \mathcal{O} , we obtain $t_0 = 0$ and $t_f = T$. For cells that will not be occupied within time T , we set $t_0 = t_f = \infty$. Examples of a static (1 cell) and moving (3 cells) object are shown in Fig. 1(b), with future occupied cells \mathbf{c}_i displayed in grey, increasingly light with increasing t_{i0} . Below, we explain how the cell occupation times t_0 and t_f will be used to check collisions with the possible robot trajectories.

B. Tentacles

As in [2], we use a set of drivable paths (tentacles), both for perception and motion execution. Each tentacle j is a semicircle that starts in R , is tangent to X , and is characterized by its curvature (i.e., inverse radius) κ_j , which belongs to \mathcal{K} , a uniformly sampled set:

$$\kappa_j \in \mathcal{K} = \{-\kappa_M, \dots, 0, \dots, \kappa_M\}.$$

The maximum desired curvature $\kappa_M > 0$, must be feasible considering the robot kinematics. Since, as we will show, our tentacles are used both for perception and motion execution, a compromise between computational cost and control accuracy must be reached to tune the size of \mathcal{K} , i.e., its sampling interval. Indeed, a large set is costly since, as we show later, various collision variables must be calculated on each tentacle. On the other hand, extending the set enhances motion precision, since more alternative tentacles can be selected for navigation. In Fig. 1(c, d), the straight and the sharpest counterclockwise ($\kappa = \kappa_M$) tentacle are dashed. When a total of 3 tentacles is used, these correspond respectively to $j = 2$ and $j = 3$. Each tentacle j is characterized by two classification areas (*dangerous* and *collision*), which are obtained by rigidly displacing, along the tentacle, two

rectangular boxes, with decreasing size, both overestimated with respect to the real robot dimensions. For each tentacle j , the sets of cells belonging to the two classification areas (shown in Fig. 1) are noted \mathcal{D}_j and $\mathcal{C}_j \subset \mathcal{D}_j$. As we will show below, the largest classification area \mathcal{D} will be used to select the safest tentacle, while the thinnest one \mathcal{C} determines the eventual necessary deceleration.

In summary, the tentacles exploit the robot geometric and kinematic characteristics. Specifically, the robot geometry (i.e., the vehicle encumbrance) defines the two classification areas \mathcal{C} and \mathcal{D} , hence the cell potential danger, while the robot kinematics (i.e., the maximum desired curvature, κ_M) define the bounds on the set of tentacles \mathcal{K} . Both aspects give useful information on possible collisions with obstacles ahead of the robot, which will be exploited, as we will show in Section V, to choose the best tentacle and to eventually slow down or stop the robot.

C. Robot occupation times

For each dangerous cell in tentacle j , i.e., for each cell $\mathbf{c}_i \in \mathcal{D}_j$, we compute the *robot occupation time* t_{ij} . This is an estimate of the time at which the large box will enter the cell, assuming the robot follows the tentacle at the current velocity. To calculate t_{ij} , we assume that the robot motion is uniform, and displace the box at the current robot linear velocity v , and at angular velocity $\omega_j = \kappa_j v$. We can then calculate robot occupation time t_{ij} :

$$t_{ij}(\mathbf{c}_i, v, \kappa_j) \in \mathbb{R}^+.$$

For instance, if the robot is not moving ($v = 0$), for every tentacle j , the cells on the box will have $t_{ij} = 0$, and all other cells in \mathcal{D}_j will have $t_{ij} = \infty$. Also note that for a given cell, t_i may differ according to the tentacle that is considered. In Fig. 1(c, d), the cells $\mathbf{c}_i \in \mathcal{D}_j$ have been displayed in grey, increasingly light with increasing t_{ij} .

D. Dangerous and collision instants

Once the obstacle and robot occupation times have been calculated for each cell, we can derive the earliest time instant at which a collision between obstacle and robot may occur on each tentacle j . By either checking all cells in \mathcal{D}_j , or focusing just on \mathcal{C}_j , we discern between *dangerous instants* and *collision instants*. These are defined as:

$$t_j = \inf_{\mathbf{c}_i \in \mathcal{D}_j} \{t_{ij} : t_{i0} \leq t_{ij} \leq t_{if}\},$$

and

$$t_j^c = \inf_{\mathbf{c}_i \in \mathcal{C}_j} \{t_{ij} : t_{i0} \leq t_{ij} \leq t_{if}\}.$$

In both cases, we seek the earliest time at which a cell is simultaneously occupied by the obstacle and by the robot box. Assuming constant robot and obstacle velocities, these metrics give an approximation of the time that the robot can travel along the tentacle without colliding. Obviously, overestimating the bounding boxes size leads also to more conservative values of t_j and t_j^c . In the following, we explain how these metrics are used: in particular, with t_j we assess the

danger on each tentacle to decide whether to follow it or not, while t_j^c determines if the robot should decelerate on tentacle j . Computation of t_j and t_j^c is illustrated, for $j = \{2, 3\}$, in the example of Fig. 1.

E. Tentacle risk function

The danger on each tentacle is assessed by *tentacle risk function* H_j . This scalar function is derived from the tentacle dangerous instant, and will be used by the controller as explained in Section V. We use t_j and tuned thresholds $t_d > 0$ and $t_s > t_d$ (d stands for dangerous, and s for safe), to design the tentacle risk function:

$$H_j = \begin{cases} 0 & \text{if } t_j \geq t_s \\ \frac{1}{2} \left[1 + \tanh \left(\frac{1}{t_j - t_d} + \frac{1}{t_j - t_s} \right) \right] & \text{if } t_d < t_j < t_s \\ 1 & \text{if } t_j \leq t_d. \end{cases}$$

Note that H_j smoothly varies from 0, when possible collisions are in the far future, to 1, when they are forthcoming. If $H_j = 0$, the tentacle is tagged as *clear*. All the H_j are compared (with a strategy explained below), to determine H in (3) and select the *best tentacle* for navigation.

V. CONTROL SCHEME

In our control scheme, the desired behaviour of the robot is related to the surrounding obstacles. When the environment is safe, the vehicle should progress forward while remaining near the taught path, with camera pointing forward ($\varphi = 0$). If avoidable obstacles are present, we apply a robot rotation for circumnavigation with an opposite camera rotation to maintain visibility. The rotation makes the robot follow the *best* tentacle in \mathcal{K} , which is selected using the strategy explained below. Finally, if collision is inevitable, the vehicle should simply stop. To select the behaviour, we assess the danger at time t with a *situation risk function* $H \in [0, 1]$, also defined below.

Stability of the desired tasks has been guaranteed in [2] by:

$$\begin{cases} v = (1 - H) v_s + H v_u \\ \omega = (1 - H) \frac{\lambda_x(x^* - x) - j_v v_s + \lambda_\varphi j_\varphi \varphi}{j_\omega} + H \kappa_b v_u \\ \dot{\varphi} = H \frac{\lambda_x(x^* - x) - (j_v + j_\omega \kappa_b) v_u}{j_\varphi} - (1 - H) \lambda_\varphi \varphi \end{cases} \quad (3)$$

In the above equations:

- H is the risk function on the best tentacle: $H = H_b$; hence, it is null if and only if the best tentacle is clear.
- $v_s > 0$ is the translational velocity in the *safe context* (i.e., when $H = 0$). It must be maximal on straight path portions, and smoothly decrease when the features quickly move in the image, i.e., at sharp robot turns (large ω), and when the camera pan angle φ is strong. Hence, we define v_s as:

$$v_s(\omega, \varphi) = v_m + \frac{v_M - v_m}{4} \sigma \quad (4)$$

with function σ defined as:

$$\sigma(\omega, \varphi) = [1 + \tanh(\pi - k_\omega |\omega|)] [1 + \tanh(\pi - k_\varphi |\varphi|)].$$

Function (4) has an upper bound $v_M > 0$ (for $\varphi = \omega = 0$), and smoothly decreases to the lower

bound $v_m > 0$, as either $|\varphi|$ or $|\omega|$ grow. Both v_M and v_m are hand-tuned variables, and the decreasing trend is determined by empirically tuned positive parameters k_ω and k_φ .

- $v_u \in [0, v_s]$ is the translational velocity in the *unsafe context* ($H = 1$). It is designed as:

$$v_u(\delta_b) = \begin{cases} v_s & \text{if } t_b^c \geq t_s^c \\ v_s \sqrt{t_b^c - t_d^c / t_s^c - t_d^c} & \text{if } t_d^c < t_b^c < t_s^c \\ 0 & \text{if } t_b^c \leq t_d^c \end{cases} \quad (5)$$

(with $t_d^c > 0$ and $t_s^c > t_d^c$ two thresholds corresponding to dangerous and safe collision times) to guarantee that the vehicle decelerates (and eventually stops) as the collision instant on the best tentacle t_b^c decreases.

- x and x^* are abscissas of the feature centroid respectively in the current and next key image.
- $\lambda_x > 0$ and $\lambda_\varphi > 0$ are empirical gains determining the convergence trend of x to x^* and of φ to 0.
- j_v , j_ω and j_φ are the components of the Jacobian relating \dot{x} and \mathbf{u} . They are:

$$\begin{aligned} j_v &= \frac{-\sin \varphi + x \cos \varphi}{\zeta} \\ j_\omega &= \frac{\rho(\cos \varphi + x \sin \varphi)}{\zeta} + 1 + x^2 \\ j_\varphi &= 1 + x^2, \end{aligned}$$

where ρ is the abscissa of the optical center in the robot frame \mathcal{F}_R , and ζ is the feature centroid depth with respect to the camera.

- κ_b is the curvature of the best tentacle. Here we detail how such a tentacle is determined. Initially, we calculate the path curvature that the robot would follow if $H = 0$:

$$\kappa = \omega/v = [\lambda_x(x^* - x) - j_v v_s + \lambda_\varphi j_\varphi \varphi] / j_\omega v_s.$$

In [2], we proved that κ is always well-defined, i.e., that $j_\omega \neq 0$. We constrain κ to the interval of feasible curvatures $[-\kappa_M, \kappa_M]$, and derive its two neighbors in \mathcal{K} : κ_n and κ_{nn} . Let κ_n be the nearest one, denoted as the *visual tentacle*³. Its situation risk function H_v is obtained by linear interpolation of the neighbours:

$$H_v = \frac{(H_{nn} - H_n) \kappa + H_n \kappa_{nn} - H_{nn} \kappa_n}{\kappa_{nn} - \kappa_n}. \quad (6)$$

If $H_v = 0$, the visual tentacle is clear and can be followed: we set $\kappa_b = \kappa_n$. Instead, if $H_v \neq 0$, we seek a clear tentacle ($H_j = 0$). First, we search among the tentacles between the visual task one and the best one at the previous iteration⁴, noted κ_{pb} . If many are present, the closest to the visual tentacle is chosen. If none of the tentacles within $[\kappa_n, \kappa_{pb}]$ is clear, we search among the others. If no tentacle in \mathcal{K} is clear, the one with minimum H_j is chosen. Ambiguities are again solved first with the distance from κ_n , then from κ_{nn} .

Let us shortly recall the main features of (3), which are detailed in [2]. When $H = 0$ (i.e., if the 2 neighbour tentacles

³We consider that intervals are defined even when the first endpoint is greater than the second: $[\kappa_n, \kappa_{nn}]$ must be read (κ_{nn}, κ_n) if $\kappa_n > \kappa_{nn}$.

⁴At the first iteration, we set $\kappa_{pb} = \kappa_n$.

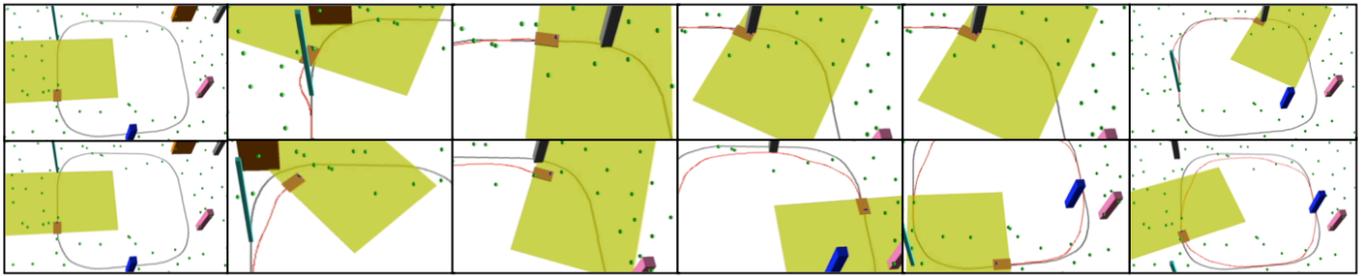


Fig. 2. Six steps of the simulations: the taught path must be followed by the robot with methods S (top) and M (bottom) and 4 moving and 1 static obstacles. Visual features (spheres), occupancy grid, and replayed paths are also shown.

are clear), the robot tracks at its best the taught path: the image error is regulated by ω , while v is set to v_s to improve tracking, and the camera is driven forward ($\varphi = 0$). When $H = 1$, $\dot{\varphi}$ ensures the visual task, and the two other inputs guarantee that the best tentacle is followed: $\omega/v = \kappa_b$. In general ($H \in [0, 1]$), the robot navigates between the taught and the best paths, and a high velocity v_s can be applied if the path is clear for future time t_s^c .

VI. EXPERIMENTS

In this section, we will detail the experiments that were used to validate our approach. These are also shown in the video attached to this paper.

All experiments have been carried out on our CyCab vehicle, set in car-like mode (i.e., using the front wheels for steering). The maximum speed attainable by the CyCab is 1.3 ms^{-1} . For preliminary simulations, we made use of Webots⁵, where we designed a virtual CyCab, and distributed random visual features, represented by spheres, as well as physical obstacles, in the environment. The robot is equipped with a coarsely calibrated 320×240 pixels 70° field of view, B&W Marlin (F-131B) camera mounted on a TRACLabs Biclops Pan/Tilt head (the tilt angle is null, to keep the optical axis parallel to the ground), and with a 2-layer, 110° scanning angle, laser SICK LD-MRS. A dedicated encoder on the TRACLabs head precisely measures the pan angle φ required in our control law (3). Since camera (10 Hz) and laser (12.5 Hz) processing are not synchronized, they are implemented on two different threads, and the control input \mathbf{u} derived from (3) is sent to the robot as soon as the visual information is available (10 Hz).

The occupancy grid is built by projecting the laser readings from the two layers on the ground, and by using: $X_M = Y_M = 10 \text{ m}$, $X_m = -2 \text{ m}$, $Y_m = -10 \text{ m}$. The cells have size $20 \times 20 \text{ cm}$. For the situation risk function, we use $t_s = 6 \text{ s}$ and $t_d = 4.5 \text{ s}$, for the unsafe translational velocity, we use $t_s^c = 5 \text{ s}$, and $t_d^c = 2 \text{ s}$, and as control gains: $\lambda_x = 1$ and $\lambda_\varphi = 0.5$. We set $\kappa_M = 0.35 \text{ m}^{-1}$ (the CyCab maximum applicable curvature, corresponding to a minimum turn radius of 2.86 m). All these values were tuned after a few simulations, and proved appropriate throughout the experiments. It is noteworthy to point out that the number of tentacles that must be processed, and correspondingly, the computational cost of the laser processing thread, increase with the context danger. For clarity, let us discuss two extreme

cases: a safe and an occupied contexts. To verify that a context is safe (i.e., that $H_v = 0$ in (6)), all the cells in the dangerous area \mathcal{D} of only the two neighbour tentacles must be explored. Instead, in a scenario where the grid is very occupied, all of the tentacles in \mathcal{K} may need to be explored. In general, this second case will be more costly than the first. The experiments showed that the computational cost of laser processing, using the chosen number of 21 tentacles, was never a critical issue with respect to that of image processing.

In all the experiments, at first, the robot is driven along a taught path. Then, moving and static objects are present on the path, while the robot replays it to follow the key images. To evaluate the experiments, we verify if the robot is able to complete the taught path until the final key image (this was the case in all experiments), and we measure its linear velocity v , averaged over the whole experiment, and denoted \bar{v} . We do not consider the 3D pose error with respect to the taught path, since our task is defined in the image space, and not in the pose space. Besides, some portions of the replayed paths, corresponding to the obstacle avoidances, are far from the taught ones. However, these deviations are indispensable to avoid collisions. In some experiments, we also compare the new approach that is presented here, and that takes into account the obstacle velocities, with the original one designed in [2]. We denote these respectively as approach M and S (for Mobile and Static).

Let us firstly describe the simulations, shown in Fig. 2. The taught visual path is a closed clockwise loop of $N = 20$ key images, and the robot must replay it, while avoiding 4 moving obstacles, with velocity norms up to 1 ms^{-1} , and a static one. Higher obstacle velocities are difficult to estimate due to the low frequency of laser processing (12.5 Hz). However, it is noteworthy to point out that 1 ms^{-1} is the walking speed of a quick pedestrian. With approach M, the vehicle is able to follow the whole path without colliding, whereas when S is used, the robot collides with the third obstacle. Let us now detail the robot behaviour in the two cases. The first obstacle (a box moving straight towards the robot) is avoided by both approaches, although with M motion prediction leads to a smoother and earlier circumnavigation. With M, the robot is faster, and reaches the brown box while it is crossing its way; but since the box is expected to leave, the robot just waits for the path to return free. With S, the robot arrives at the same point late, when the box is far. The third, grey box moves straight towards the robot, like the first one. Since it is slightly faster, this time S is not reactive enough, and a

⁵www.cyberbotics.com



Fig. 3. Ten relevant iterations of the experiment with two crossing pedestrians. For each iteration, we show the occupancy grid (left) and current image (right). In the occupancy grid, the dangerous cell sets associated with the visual tentacle and to the best tentacle (when different) are shown, and two black segments indicate the lidar amplitude. Only cells that we predict to be occupied in the next T s have been drawn in the grid. The segments link the current and next key image points.



Fig. 4. Comparison between methods S (top) and M (bottom) as a pedestrian crosses the path in front of the robot.

collision occurs. On the other hand, with M the boxes are easily avoided. The new approach also prevails in speed: the average velocity $\bar{v} = 0.67 \text{ ms}^{-1}$ with M, and $\bar{v} = 0.49 \text{ ms}^{-1}$ with S.

After the simulations, the framework has been ported on our CyCab vehicle.

In a first experiment, two pedestrians are passing during navigation: one crosses the path, and the other walks straight towards the robot. We show, in Fig. 3, relevant iterations with the corresponding occupancy grids and currently viewed images. In the occupancy grid, the propagation of cells occupied by the persons is visible at iterations 2-8. With the crossing pedestrian (iterations 2-5), since no collision is predicted, the robot keeps following the visual tentacle. Instead, with the forward walking pedestrian, a collision is predicted at iteration 7; then, the robot selects the best tentacle to avoid the person. Visual path replaying is again successful, with $\bar{v} = 0.87 \text{ ms}^{-1}$.

Then, we have compared methods S and M in an experiment, where a single pedestrian crosses the taught path in front of the robot (see Fig. 4, with control inputs plotted in Fig. 5). The smooth trend of \mathbf{u} at the beginning and end of

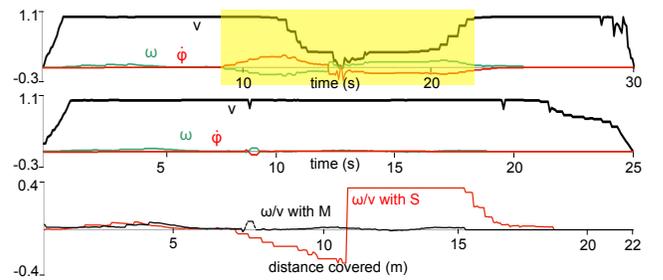


Fig. 5. Single pedestrian experiment. Top and center, respectively: control inputs using S and M, with v (ms^{-1}), ω (rads^{-1}), $\dot{\varphi}$ (rads^{-1}), and iterations with strong H highlighted. Bottom: applied curvature ω/v (in m^{-1}) using S and M.

the experiments is due to the acceleration saturation carried out at the CyCab low-level control. With controller S (top in both figures), the robot attempts avoidance on the right, since tentacles on the left are occupied by the person. This is clearly a doomed strategy, which leads the robot toward the pedestrian. Then, the robot must decelerate and almost stop ($v \approx 0$ after 15 s) when the pedestrian is very near.

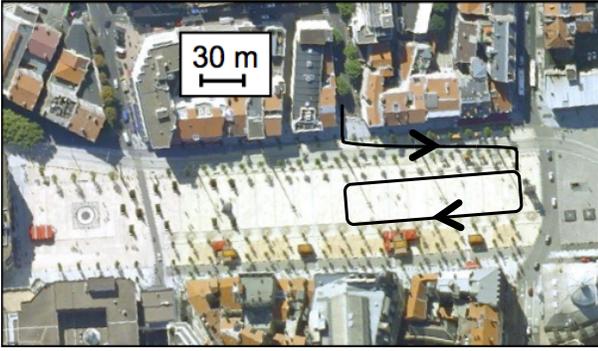


Fig. 6. City center of Clermont Ferrand, with one of the navigation paths where the urban experiments have been carried out.

Navigation is resumed only once the path is clear again. On the other hand, with controller M, as the pedestrian walks, the prediction of his future position makes him irrelevant from a safety viewpoint: risk function H (highlighted in Fig. 5), which was relevant with S, is now null. Hence, the robot does not need to reduce its speed (\bar{v} is 0.89 ms^{-1} with M, and 0.76 ms^{-1} with S) nor to deviate from the path (in the bottom of Fig. 5, the applied curvature is smaller). The image error with respect to the database $x - x^*$, averaged over the experiment is also reduced with M: 7 instead of 12 pixels.

As part of the final demonstration of the French ANR project CityVIP, we have also validated our framework in a urban context, in the city of Clermont Ferrand. The experiments have taken place in the crowded neighbourhood of the central square Place de Jaude, shown in Fig. 6. For four entire days, our Cycab has navigated autonomously, continuously replaying a set of visual paths of up to 700 m each, amidst a variety of unpredictable obstacles, such as cars, pedestrians, bicycles and scooters. In Fig. 7, we show some significant snapshots of the experiments that were carried out in Clermont Ferrand. These include photos of the Cycab, as well as images acquired by the on-board camera during autonomous navigation. These experiments are also shown in the video attached to this paper.

In Fig. 7(a-c), Cycab is moving in a busy street, crowded with pedestrians and vehicles. First, in Fig. 7(a), we show the behaviour adopted in the presence of a pedestrian (a lady with black skirt) crossing a narrow street: the Cycab brakes, since avoidance is impossible. In general, the robot would either circumnavigate the person or stop, and in four days no one has ever even closely been endangered nor touched by the vehicle. In many experiments, Cycab has navigated among fast moving vehicles (cars in Fig. 7(b), and a scooter in 7(c)), and manual security intervention was never necessary. The robot has also successfully avoided many static obstacles, including a stationing police patrol (Fig. 7(d)) and another electric vehicle (Fig. 7(e)). When all visual features are occluded by an obstacle, the robot stops, but resumes navigation as soon as the obstacle moves and the features are again visible.

Moreover, we have thoroughly tested the behaviour of our system with respect to varying light, which is an important aspect in outdoor appearance-based navigation. Varying light has been very common in the extensive Clermont Ferrand

experiments, which would last the whole day, from the first light to sunset, both with cloudy and clear sky. In some experiments, we could control the robot in different lighting conditions, using the same taught database. For instance, Fig. 7(f) shows two images acquired approximately at the same position at 5 p.m. (top) and 11 a.m. (bottom), while navigating with the same key images. However, in spite of the robustness of the image processing algorithms, which has been proved in [33], in some cases (e.g., when the camera was overexposed to sunlight), the visual features required for navigation could not be detected. Future work in adapting the camera automatic shutter settings should solve this issue.

Overall, Cycab has navigated using an average of approximately 60 visual points on each image, and some paths have even been completed using less than 30 points. Along with all the cited technical aspects, the experiments highlighted the reactions of non-robotic persons to the use of autonomous ground vehicles in everyday life. Most passer-bys had not been informed of the experiments, and responded with curiosity, surprise, enthusiasm, and - rarely - slight apprehension.

VII. CONCLUSIONS

We presented a novel framework with simultaneous laser-based moving obstacle avoidance and outdoor vision-based navigation, without any 3D model or path planning. It merges a reactive, tentacle-based technique with visual servoing, to guarantee path following, obstacle bypassing, and collision avoidance by deceleration. In particular, for the first time obstacle velocities are accounted for within a visual navigation scheme. To estimate the obstacle velocities, we have designed a Kalman-based observer. Then, we utilize the velocities to predict possible collisions between robot and obstacles. Our approach is validated in a series of experiments (including urban environments), and it is compared with a similar controller that does not consider obstacle velocities.

The results show that, by predicting the obstacle displacements within the candidate tentacles, the robot behaviour is safer and smoother, and higher velocities can be attained. The framework can be applied in realistic and challenging situations including real moving obstacles (e.g., cars and pedestrians). To our knowledge, this is the first time that outdoor visual navigation with moving obstacle avoidance is carried out in urban environments at approximately 1 ms^{-1} on over 500 m, using neither GPS nor maps.

In the future, we will investigate scenarios, where obstacles are not translating, as assumed here, and can approach the vehicle from behind. For the latter case, the current configuration (forward-looking lidar) must be modified. Perspective work also includes automatic prevention of the visual occlusions provoked by the obstacles.

REFERENCES

- [1] A. Cherubini, B. Grechanichenko, F. Spindler and F. Chaumette. "Avoiding Moving Obstacles during Visual Navigation", *IEEE Int. Conf. on Robotics and Automation*, 2013.
- [2] A. Cherubini and F. Chaumette. "Visual navigation of a mobile robot with laser-based collision avoidance", *Int. Journal of Robotics Research*, vol. 32 no. 2, 2013, pp. 189–205.

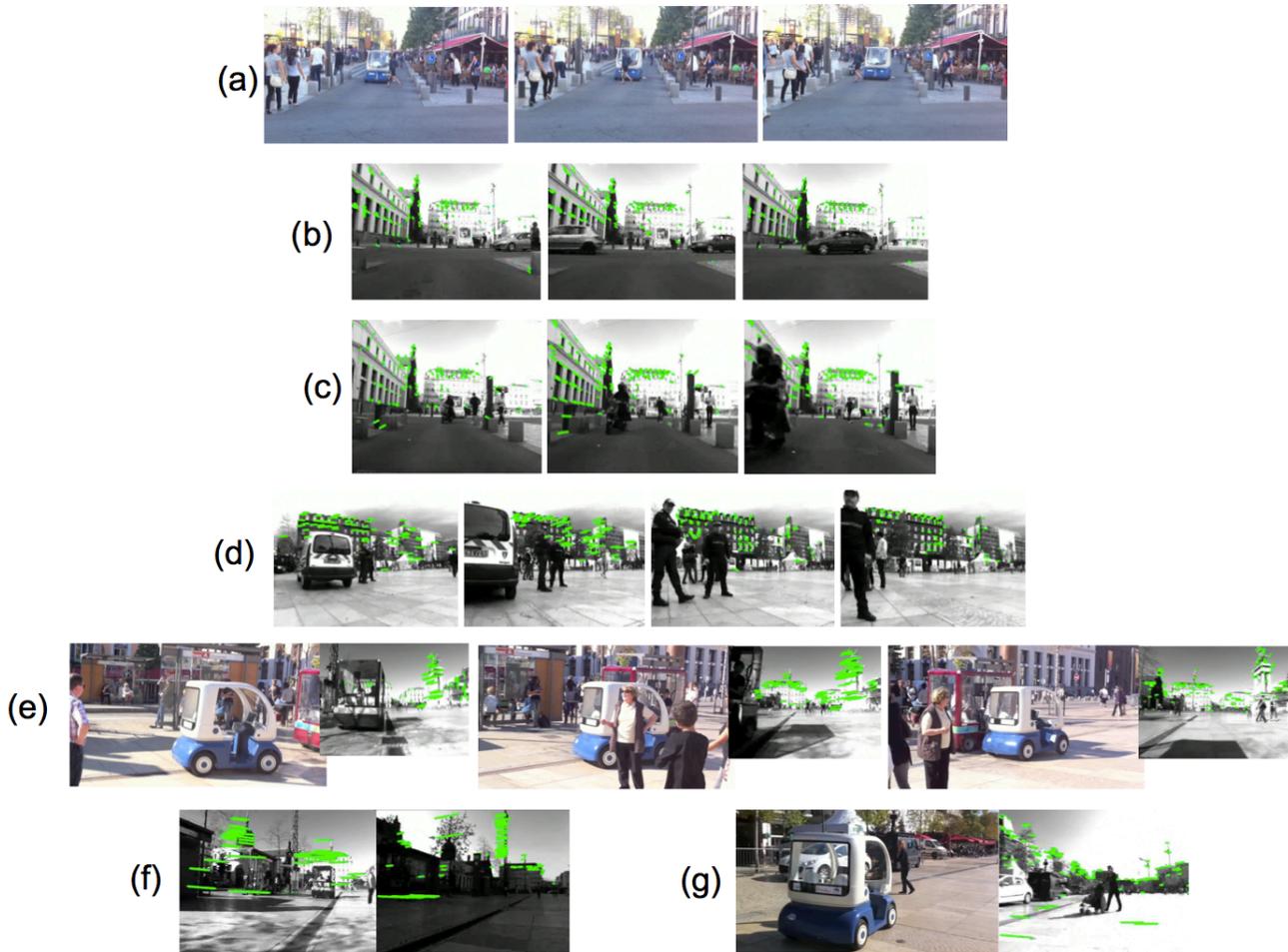


Fig. 7. Snapshots of the urban experiments. (a) Avoiding a crossing pedestrian. (b-c) Navigating close to moving cars and to a scooter, respectively. (d-e) Avoiding a stationing police patrol and a stationing vehicle, respectively. (f) Navigating with different light conditions, using the same taught database. (g) Avoiding a pedestrian with a baby pushchair.

- [3] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. Van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian and P. Mahoney, "Stanley: The robot that won the DARPA Grand Challenge" in *Journal of Field Robotics*, vol. 23, no. 9, 2006, pp. 661 - 692.
- [4] U. Nunes, C. Laugier and M. Trivedi, "Introducing perception, planning, and navigation for Intelligent Vehicles" in *IEEE Trans. on Intelligent Transportation Systems*, vol. 10, no. 3, 2009, pp. 375-379.
- [5] A. Broggi, L. Bombini, S. Cattani, P. Cerri and R. I. Fedriga, "Sensing requirements for a 13000 km intercontinental autonomous drive", *IEEE Intelligent Vehicles Symposium*, 2010, San Diego, USA.
- [6] M. Buehler, K. Lagnemma and S. Singh (Editors), "Special Issue on the 2007 DARPA Urban Challenge, Part I-III", in *Journal of Field Robotics*, vol. 25, no. 8-10, 2008, pp. 423-860.
- [7] J. J. Guerrero, A. C. Murillo and C. Sagüés, "Localization and Matching using the Planar Trifocal Tensor with Bearing-only Data", *IEEE Trans. on Robotics*, vol. 24, no. 2, 2008, pp. 494-501.
- [8] D. Scaramuzza and R. Siegwart, "Appearance-Guided Monocular Omnidirectional Visual Odometry for Outdoor Ground Vehicles", *IEEE Trans. on Robotics*, vol. 24, no. 5, 2008, pp. 1015-1026.
- [9] F. Bonin-Font, A. Ortiz and G. Oliver, "Visual navigation for mobile robots: a survey", *Journal of Intelligent and Robotic Systems*, vol. 53, no. 3, 2008, pp. 263-296.
- [10] G. López-Nicolás, N. R. Gans, S. Bhattacharya, C. Sagüés, J. J. Guerrero and S. Hutchinson, "An Optimal Homography-Based Control Scheme for Mobile Robots with Nonholonomic and Field-of-View Constraints", *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, vol. 40, no. 4, 2010, pp. 1115-1127.
- [11] A. Diosi, S. Segvic, A. Remazeilles and F. Chaumette, "Experimental Evaluation of Autonomous Driving Based on Visual Memory and Image Based Visual Servoing", *IEEE Trans. on Intelligent Transportation Systems*, vol. 12, no. 3, 2011, pp. 870-883.
- [12] T. Wada, S. Doi and S. Hiraoka, "A deceleration control method of automobile for collision avoidance based on driver's perceptual risk", *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009.
- [13] J. Minguez, F. Lamiraux and J.-P. Laumond, "Motion planning and obstacle avoidance", in *Springer Handbook of Robotics*, B. Siciliano, O. Khatib (Eds.), Springer, 2008, pp. 827-852.
- [14] J. C. Latombe, "Robot motion planning", 1991, *Kluwer Academic, Dordredt*.
- [15] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots", *ICRA*, 1985.
- [16] A. Ohya, A. Kosaka and A. Kak, "Vision-based navigation by a mobile robot with obstacle avoidance using a single-camera vision and ultrasonic sensing", *IEEE Trans. on Robotics and Automation*, vol. 14, no. 6, 1998, pp. 969-978.
- [17] Z. Yan, X. Xiaodong, P. Xuejun and W. Wei, "Mobile robot indoor navigation using laser range finder and monocular vision", *IEEE Int. Conf. on Robotics, Intelligent Systems and Signal Processing*, 2003.
- [18] F. Lamiraux, D. Bonnafous and O. Lefebvre, "Reactive path deformation for nonholonomic mobile robots", in *IEEE Trans. on Robotics*, vol. 20, no. 6, 2004, pp. 967-977.
- [19] T.-S. Lee, G.-H. Eoh, J. Kim and B.-H. Lee, "Mobile robot navigation with reactive free space estimation", *IROS*, 2010.
- [20] J. Borenstein and Y. Koren, "The Vector Field Histogram - Fast obstacle

- avoidance for mobile robots”, *IEEE Trans. on Robotics and Automation*, vol. 7, no. 3, 1991, pp. 278-288.
- [21] J. Minguez, “The Obstacle-Restriction Method (ORM) for robot obstacle avoidance in difficult environments”, *IROS*, 2005.
- [22] M. Mujahad, D. Fischer, B. Mertsching and H. Jaddu “Closest Gap based (CG) reactive obstacle avoidance navigation for highly cluttered environments”, *IROS*, 2010.
- [23] P. Fiorini and Z. Shiller, “Motion Planning in Dynamic Environments Using Velocity Obstacles”, in *Int. Journal of Robotics Research*, vol. 17, no. 7, 1998, pp. 760 – 772.
- [24] D. Wilkie and J. Van den Berg, D. Manocha, “Generalized Velocity Obstacles”, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009.
- [25] A. Wu and J P. How, “Guaranteed infinite horizon avoidance of unpredictable, dynamically constrained obstacles”, in *Autonomous Robots* vol. 32, no. 3, 2012, pp. 227-242.
- [26] D. Fox, W. Burgard and S. Thrun, “The Dynamic Window approach to obstacle avoidance”, in *IEEE Robotics and Automation Magazine*, vol. 4, no. 1, 1997, pp. 23–33.
- [27] B. Damas and J. Santos-Victor, “Avoiding Moving Obstacles: the Forbidden Velocity Map”, *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009.
- [28] M. Seder and I. Petrović, “Dynamic window based approach to mobile robot motion control in the presence of moving obstacles”, *IEEE Int. Conf. on Robotics and Automation*, 2007.
- [29] C. Fulgenzi, A. Spalanzani, C. Laugier, “Probabilistic motion planning among moving obstacles following typical motion patterns”, in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009.
- [30] T. Fraichard, “A Short Paper about Motion Safety”, *IEEE Int. Conf. on Robotics and Automation*, 2007.
- [31] S. Bouraine, T. Fraichard and H. Salhi, “Provably safe navigation for mobile robots with limited field-of-views in dynamic environments”, in *Autonomous Robots*, vol. 32, no. 3, 2012, pp. 267-283.
- [32] F. von Hundelshausen, M. Himmelsbach, F. Hecker, A. Mueller, and H.-J. Wunsche, “Driving with tentacles-Integral structures of sensing and motion”, in *Journal of Field Robotics*, vol. 25, no. 9, 2008, pp. 640–673.
- [33] E. Royer, M. Lhuillier, M. Dhome and J.-M. Lavest, “Monocular vision for mobile robot localization and autonomous navigation”, in *Int. Journal of Computer Vision*, vol. 74, no. 3, 2007, pp. 237–260.
- [34] D. Zeng Wang, I. Posner and P. Newman, “What Could Move? Finding Cars, Pedestrians and Bicyclists in 3D Laser Data”, in *IEEE Int. Conf. on Robotics and Automation*, 2012.
- [35] N. Wojke and M. Häselich, “Moving Vehicle Detection and Tracking in Unstructured Environments”, in *IEEE Int. Conf. on Robotics and Automation*, 2012.



applications. He is the software architect of the open-source ViSP (Visual Servoing Platform) library and is involved in the software transfer to industrial or academic partners.

Fabien Spindler Fabien Spindler graduated from ENI Brest engineer school (specialisation in electronics) in 1992, and received the Master of Science degree in Electronics and Aerospace Telecommunication from Supaero in Toulouse in 1993. Since 1994, he has been with Inria in Rennes as research engineer. He is in charge of the material and software management of several robotic experimentation platforms dedicated to researches in visual servoing. His interests include software engineering for the design of real-time computer vision and robotics



AFCET/CNRS Prize for the best French thesis in automatic control in 1991. He also received with Ezio Malis the 2002 King-Sun Fu Memorial Best IEEE Transactions on Robotics and Automation Paper Award. He has been Associate Editor of the IEEE Transactions on Robotics from 2001 to 2005 and is now in the Editorial Board of the Int. Journal of Robotics Research.

François Chaumette François Chaumette was graduated from École Nationale Supérieure de Mécanique, Nantes, France, in 1987. He received the Ph.D. degree in computer science from the University of Rennes, France, in 1990. Since 1990, he has been with Inria in Rennes where he is now Senior Research Scientist and head of the Lagadic group (<http://www.irisa.fr/lagadic>). His research interests include robotics and computer vision, especially visual servoing and active perception. Dr. Chaumette is IEEE Fellow. He received the



include: visual servoing for mobile robots, nonholonomic robot navigation, human-robot interaction, and humanoid robotics.

Andrea Cherubini Andrea Cherubini received the MSc in Mechanical Engineering in 2001 from the University of Rome La Sapienza and a second MSc in Control Systems in 2003 from the University of Sheffield, U.K. In 2008, he received the PhD degree in Control Systems from the Dipartimento di Informatica e Sistemistica, University of Rome La Sapienza. From 2008 to 2011, he worked as post-doctoral fellow at Inria Rennes. He is currently Associate Professor at Université de Montpellier 2, and researcher at LIRMM. His research interests



A unified multimodal control framework for human–robot interaction



Andrea Cherubini*, Robin Passama, Philippe Fraitse, André Crosnier

Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier LIRMM, Université Montpellier - CNRS, 161 Rue Ada, 34392 Montpellier, France

HIGHLIGHTS

- A unified multimodal sensor-based control framework is proposed.
- Pose, vision and force tasks can be realized either exclusively or simultaneously.
- Self-adapting gains and homotopies between the tasks guarantee safe operation.
- The approach is validated in an industrial task: collaborative screwing.

ARTICLE INFO

Article history:

Received 13 October 2014

Received in revised form

24 February 2015

Accepted 2 March 2015

Available online 1 April 2015

Keywords:

Reactive and sensor-based control

Human–robot interaction

Visual servoing

ABSTRACT

In human–robot interaction, the robot controller must reactively adapt to sudden changes in the environment (due to unpredictable human behaviour). This often requires operating different modes, and managing sudden signal changes from heterogeneous sensor data. In this paper, we present a multimodal sensor-based controller, enabling a robot to adapt to changes in the sensor signals (here, changes in the human collaborator behaviour). Our controller is based on a unified task formalism, and in contrast with classical hybrid vision–force–position control, it enables smooth transitions and weighted combinations of the sensor tasks. The approach is validated in a mock-up industrial scenario, where pose, vision (from both traditional camera and Kinect), and force tasks must be realized either exclusively or simultaneously, for human–robot collaboration.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

Recently, the attention of robotics researchers worldwide has turned towards the field of human–robot interaction (HRI [1–5]), to enable close collaboration between human and robot [6,7]. In this context, the robot must infer the user intention, to interact more naturally, from the human perspective [8–10]. To this end, both visual (e.g., based on Microsoft Kinect™ [11]) and force feedback, have been used [12–16]. Generally, we believe that direct sensor-based methods, such as visual servoing [17], provide better solutions, for intuitive HRI, than planning techniques, requiring a priori models of the environment and agents [18]. Moreover, force and vision should be used concurrently, since the information they provide is complementary. One pioneer work in this sense is [19], where force and visual control are used to avoid collisions, while tracking human motion during interaction.

However, the authors do not provide a unified solution for integrating the two sensing modalities. Instead, since the vision and force sensors often measure different physical phenomena, it is preferable to directly combine their data at the control level, rather than to apply multi-sensory fusion, or to design complex state machines. This idea has been initially proposed in [20,21], by adapting the *hybrid position–force control* paradigm [22]: force constrains some motion directions, while vision drives the remaining degrees of freedom. Later, the authors of [23] have presented a list of hybrid control configurations, and divided the degrees of freedom to be controlled by vision and force. An alternative is *impedance/admittance control* [24], which has been integrated with visual [25] and even tactile [26] control, to account for external forces. Although many techniques for merging vision, force and position control have been designed, the presence of the human in the robot control loop is rarely accounted for.

In our previous work [27], we have started the design of a multimodal framework for human–robot cooperation. The approach is marker-less, and has been validated in a mock-up industrial scenario. However, the following contributions are brought here, with regard to that work:

* Corresponding author.

E-mail addresses: Andrea.Cherubini@lirmm.fr (A. Cherubini), robin.passama@lirmm.fr (R. Passama), philippe.fraitse@lirmm.fr (P. Fraitse), andre.crosnier@lirmm.fr (A. Crosnier).

- a unified formalism, inspired by inverse kinematics [28,29] guarantees the controller stability, independently from the sensor modality;
- the use of smooth transitions (homotopies) between the sensor-based tasks, and of self-adapting gains, limits the robot accelerations, thus guaranteeing safer operation;
- in contrast with hybrid vision–force control, it is possible to control a same task direction using weighted combinations of different sensors, and sensor-based tasks can be expressed in different reference frames;

Other, minor improvements, with regard to [27], include the introduction of force-based control, guaranteeing safety of HRI, and better accuracy, velocity, and control smoothness. Moreover, our task-oriented approach, in contrast with similar ones, such as the stack-of-tasks [29] and constraint-based programming [30,31], is directly useable in real HRI scenarios (to our knowledge, the method presented in [31] has been used, for now, only for human collision avoidance).

The article is organized as follows. In Section 2, we present our general framework for multimodal control for HRI. In Section 3, relevant variables and sensor-based tasks are defined. Based on these preliminaries, Section 4 shows how the general framework can be instantiated for an industrial case study. Experimental results are reported in Section 5, and summarized in the Conclusion.

2. Control framework

To safely interact with the human, the designed controller must rely on the various sensing modalities present on the robot. These may include cameras (for vision), force/torque sensors, skin (for tact), or proprioception (e.g., for positioning).

A commonly used approach to merge the various sensing modalities directly at the control level is *hybrid sensors control*, e.g. hybrid force/position [22] or hybrid force/vision [20] control. This approach was recently extended in a framework integrating vision, force and tact to realize physical interaction tasks [32]. We hereby recall the formulation of that approach, and propose a more generic one, based on classic *inverse kinematics control* [28].

Let k be the dimension of the operational space associated with the end effector (e.g., $k = 3$ in the case of a planar manipulator). Consider n senses and, for each sense, the task vector $\mathbf{s}_m \in \mathbb{R}^k$, with $m = 1, \dots, n$. For example, a task associated with the sense of vision could consist in controlling an on-board camera to make it look at a target point, and a task associated with the sense of force could consist in applying a desired wrench with the end effector (e.g., $\mathbf{s}_f = [f_x, f_y, m_z]$ for a planar manipulator). In this work, since position, vision and force are used, $n = 3$. All n tasks have the same size k , and, if the sensor provides less than k measures, it will be sufficient to select the task components corresponding to the actual measures, as will be explained later.

Each task is related to the Cartesian velocity of the end effector, $\mathbf{v} \in \mathbb{R}^k$ by the $k \times k$ matrix \mathbf{L}_m (called *interaction matrix* in the case of visual servoing):

$$\dot{\mathbf{s}}_m = \mathbf{L}_m \mathbf{v}. \quad (1)$$

Stacking the n tasks yields:

$$\dot{\mathbf{s}} = \mathbf{L} \mathbf{v}, \quad \text{with } \bar{\mathbf{s}} = \begin{bmatrix} \mathbf{s}_1 \\ \vdots \\ \mathbf{s}_n \end{bmatrix} \in \mathbb{R}^{kn} \quad \text{and} \quad (2)$$

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}_1 \\ \vdots \\ \mathbf{L}_n \end{bmatrix} \in \mathbb{R}^{kn} \times \mathbb{R}^k.$$

As aforementioned, a combination of tasks defined by different senses (i.e., by components of the different \mathbf{s}_m) is realizable, as long as its size is also k . The tasks are selected thanks to n positive definite square diagonal *selection matrices* of size k , denoted \mathbf{S}_m , that activate or deactivate a given task component. Then, the k -dimensional hybrid task *to be realized*, is a linear mapping of the complete $\bar{\mathbf{s}}$:

$$\dot{\mathbf{s}} = \mathbf{S} \dot{\bar{\mathbf{s}}}, \quad \text{with } \mathbf{S} = [\mathbf{S}_1 \dots \mathbf{S}_n] \in \mathbb{R}^k \times \mathbb{R}^{kn}. \quad (3)$$

Note that, as outlined above, if the m -th sensor provides less than k measures, the missing components can be deselected by simply setting to zero the corresponding row in \mathbf{S}_m . The selection matrices can also be used, as will be shown later, to weigh outputs from different sensors and combine them into a single task.

Merging (3) and (2) gives the open-loop behaviour of the task in function of the end effector velocity:

$$\dot{\mathbf{s}} = \mathbf{S} \mathbf{L} \mathbf{v}. \quad (4)$$

Inverse kinematics control relies on the assumption that matrix $\mathbf{S} \mathbf{L}$ is invertible.¹ Then, the optimal² solution of (4) ensuring exponential convergence of \mathbf{s} to the desired constant task \mathbf{s}^* is:

$$\mathbf{v} = (\mathbf{S} \mathbf{L})^{-1} (\mathbf{s}^* - \mathbf{s}). \quad (5)$$

Indeed, replacing this into (4) yields:

$$\dot{\mathbf{s}} = \mathbf{s}^* - \mathbf{s}, \quad (6)$$

guaranteeing that $\mathbf{s} = \mathbf{s}^*$ is a stable equilibrium for the closed-loop system.

Let us now compare (5) with the *hybrid sensors control* used in numerous works [20,22,23,32]. This approach consists in assigning each sensing modality to a Cartesian direction in the operational space, and then summing the velocities associated with the selected sensors:

$$\mathbf{v} = \sum_m \mathbf{S}_m \mathbf{v}_m, \quad (7)$$

with some assumption on the selection matrices, e.g., that they are orthogonal, as in [32].

Assuming each \mathbf{L}_m is invertible, exponential convergence of \mathbf{s}_m to \mathbf{s}_m^* , according to (1), is guaranteed by applying:

$$\mathbf{v}_m = \mathbf{L}_m^{-1} (\mathbf{s}_m^* - \mathbf{s}_m). \quad (8)$$

Plugging (8) into (7), we obtain the *hybrid sensors control* expression:

$$\mathbf{v} = \tilde{\mathbf{S}} \mathbf{L} (\bar{\mathbf{s}}^* - \bar{\mathbf{s}}) \quad \text{with } \tilde{\mathbf{L}} = \begin{bmatrix} \mathbf{L}_1^{-1} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \mathbf{L}_n^{-1} \end{bmatrix} \in \mathbb{R}^{kn} \times \mathbb{R}^{kn}. \quad (9)$$

This controller is optimal for (4), if and only if (9) coincides with (5):

$$\tilde{\mathbf{S}} \mathbf{L} (\bar{\mathbf{s}}^* - \bar{\mathbf{s}}) = (\mathbf{S} \mathbf{L})^{-1} (\mathbf{s}^* - \mathbf{s}) \quad \forall (\bar{\mathbf{s}}^*, \dot{\bar{\mathbf{s}}}^*) \in \mathbb{R}^2. \quad (10)$$

This is equivalent, considering (3), to:

$$\tilde{\mathbf{S}} \mathbf{L} = (\mathbf{S} \mathbf{L})^{-1} \mathbf{S}. \quad (11)$$

In general, this is not the case, but we hereby provide two necessary conditions for it to be true.

¹ Otherwise, specific strategies for avoiding singularities, which are out of the scope of this paper, are to be devised.

² Throughout the paper, we refer to controllers as *optimal* when they provide the least squares solution to the task, i.e., they minimize the control effort.

Property. Hybrid sensors control (9) is optimal if the diagonal selection matrices \mathbf{S}_m are all binary and orthogonal, and if the sensor matrices \mathbf{L}_m are all diagonal.

Proof. Since all \mathbf{S}_m are binary (hence, idempotent) and orthogonal:

$$\sum_{m=1}^n \mathbf{S}_m = \mathbf{I}. \quad (12)$$

Moreover, binary \mathbf{S}_m imply that \mathbf{S} has full rank, so its right pseudoinverse can be derived to show, using (12), that it coincides with its transpose:

$$\mathbf{s}^\dagger = \mathbf{S}^\top (\mathbf{S}\mathbf{S}^\top)^{-1} = \mathbf{S}^\top \left(\sum_m \mathbf{S}_m^2 \right)^{-1} = \mathbf{S}^\top \sum_m \mathbf{S}_m = \mathbf{S}^\top. \quad (13)$$

Then, post-multiplying condition (11) by $\mathbf{S}^\dagger = \mathbf{S}^\top$, we obtain:

$$\tilde{\mathbf{S}}\mathbf{L}\mathbf{s}^\dagger = (\mathbf{S}\mathbf{L})^{-1} \mathbf{S}\mathbf{s}^\dagger, \quad (14)$$

which leads to:

$$\tilde{\mathbf{S}}\mathbf{L}\mathbf{s}^\top = (\mathbf{S}\mathbf{L})^{-1}. \quad (15)$$

- The first member of (15) becomes:

$$\tilde{\mathbf{S}}\mathbf{L}\mathbf{s}^\top = \sum_m \mathbf{S}_m \mathbf{L}_m^{-1} \mathbf{s}_m. \quad (16)$$

By commuting the matrix product (since all \mathbf{L}_m and \mathbf{S}_m are diagonal, and have the same size), and taking advantage of the idempotency of the \mathbf{S}_m , we obtain:

$$\tilde{\mathbf{S}}\mathbf{L}\mathbf{s}^\top = \sum_m \mathbf{S}_m^2 \mathbf{L}_m^{-1} = \sum_m \mathbf{S}_m \mathbf{L}_m^{-1}. \quad (17)$$

- The second member of (15) becomes:

$$(\mathbf{S}\mathbf{L})^{-1} = \left(\sum_m \mathbf{S}_m \mathbf{L}_m \right)^{-1}. \quad (18)$$

Noting s_{im} and l_{im} the i -th elements of \mathbf{S}_m and \mathbf{L}_m , respectively:

$$\begin{aligned} (\mathbf{S}\mathbf{L})^{-1} &= \left[\text{diag} \left(\sum_m s_{1m} l_{1m}, \dots, \sum_m s_{km} l_{km} \right) \right]^{-1} \\ &= \text{diag} \left(\frac{1}{\sum_m s_{1m} l_{1m}}, \dots, \frac{1}{\sum_m s_{km} l_{km}} \right). \end{aligned} \quad (19)$$

Since for each i , exactly one s_i is non-null and equal to 1, this equation can be rewritten:

$$(\mathbf{S}\mathbf{L})^{-1} = \sum_m \mathbf{S}_m \mathbf{L}_m^{-1}. \quad (20)$$

Eqs. (17) and (20) demonstrate that the first and second members of (15) coincide, and that the property is therefore valid. \square

To summarize, hybrid sensors control provides an optimal solution for (4) under two strong assumptions.

1. All the sensor tasks \mathbf{s}_m must be expressed in the same reference frame. This can be stated from (1), subject to the condition that the \mathbf{L}_m matrices are diagonal.
2. Only one sensor can be used to control each end effector direction. This can be stated from (7), subject to the condition that the \mathbf{S}_m are binary and orthogonal.

These assumptions are mentioned in all works that apply hybrid sensors control. However, they limit its use in practical applications. For instance, merging image-based visual servoing [17], which defines the visual task in the image frame, with force control, usually implemented in the force sensor frame, would infringe the first assumption.

On the other hand, to guarantee stability of the closed-loop system, the classical inverse control scheme (5) only requires that $\mathbf{S}\mathbf{L}$ is invertible (a weaker assumption, that is always true if the \mathbf{S}_m are binary, and the \mathbf{L}_m diagonal). Controller (5) can be applied even if the task frames associated with each sensor are different, and even if a task is defined for multiple robots [33], or as a combination of heterogeneous sensor data (as shown in many recent works by Mansard et al. [29,34,35]).

Let us now apply the previous result, by expressing the problem in the joint space, rather than in the operational space. The robot joint velocity is denoted $\dot{\mathbf{q}} \in \mathbb{R}^j$, with j the number of degrees of freedom. We assume that $j \geq k$, so that \mathbf{s} can be realized. If $j > k$, redundancy exists, and one can also minimize a scalar cost function $h(\mathbf{q}) \in \mathbb{R}$, while realizing the task \mathbf{s} .

Each task is related to the joint velocity by:

$$\dot{\mathbf{s}}_m = \mathbf{J}_m(\mathbf{q}, \mathbf{s}_m) \dot{\mathbf{q}}, \quad (21)$$

where

$$\mathbf{J}_m(\mathbf{q}, \mathbf{s}_m) = \frac{\partial \mathbf{s}_m}{\partial \mathbf{q}} \quad (22)$$

is the corresponding task Jacobian, of dimension $k \times j$, that depends on both the robot configuration and on the task. By stacking the n tasks, and using (3), we obtain:

$$\dot{\mathbf{s}} = \mathbf{S}\dot{\mathbf{s}} = \mathbf{S}\mathbf{J}(\mathbf{q}, \bar{\mathbf{s}}) \dot{\mathbf{q}}, \quad (23)$$

where:

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_n \end{bmatrix} \in \mathbb{R}^{kn} \times \mathbb{R}^j. \quad (24)$$

The multimodal controller that we propose, for driving \mathbf{s} to \mathbf{s}^* is given by:

$$\dot{\mathbf{q}} = (\mathbf{S}\mathbf{J})^\dagger \mathbf{A}(\mathbf{s}^* - \mathbf{s}) + [\mathbf{I} - (\mathbf{S}\mathbf{J})^\dagger (\mathbf{S}\mathbf{J})] \nabla \mathbf{h} \quad (25)$$

In the above equation:

- $(\mathbf{S}\mathbf{J})^\dagger$ is the $j \times k$ right pseudoinverse of $\mathbf{S}\mathbf{J}$. We assume that $\mathbf{S}\mathbf{J}$ is full rank during operation, so that the pseudoinverse can be calculated. This was the case throughout the experiments and is a common assumption in inverse kinematics control [34].
- \mathbf{A} is a positive definite square diagonal matrix of dimension k that determines the convergence rate of \mathbf{s} to \mathbf{s}^* ;
- the term $\nabla \mathbf{h} = \frac{\partial h}{\partial \mathbf{q}}$ (i.e., $\nabla \mathbf{h} = 0$ when $j = k$) is introduced in order to minimize cost function h in case of redundancy.

System (23), controlled by (25), is globally asymptotically stable with respect to the k selected tasks. Indeed, plugging (25) into (23) yields:

$$\dot{\mathbf{s}} = \mathbf{A}(\mathbf{s}^* - \mathbf{s}). \quad (26)$$

Thus, since \mathbf{A} is a positive definite diagonal matrix, $\mathbf{s} = \mathbf{s}^*$ is a stable equilibrium for the closed-loop system. Also, note that minimization of h has no effect on the convergence rate of the task.

For constant gain matrix \mathbf{A} , convergence of the task will be exponential according to (26). Thus, since (25) is a proportional feedback controller, the joint velocities will also follow an exponential trend, an unwanted behaviour which may lead to abrupt velocity changes at task transitions (i.e., when the error suddenly

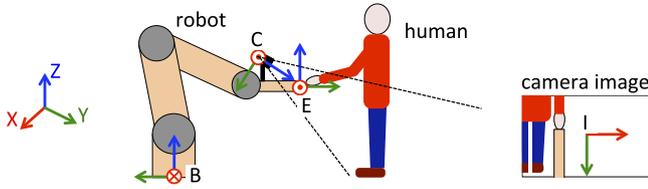


Fig. 1. Reference frames used in our multimodal framework for human–robot interaction.

increases). A simple solution to this, is the use, for each task, of an adaptive gain matrix, function of the task error $\mathbf{s}^* - \mathbf{s}$, inspired by [36]:

$$\Lambda(\mathbf{s}) = \Lambda^* \left[e^{-\alpha \|\mathbf{s}^* - \mathbf{s}\|} + \beta \left(1 - e^{-\alpha \|\mathbf{s}^* - \mathbf{s}\|} \right) \right]. \quad (27)$$

In (27), Λ^* is the diagonal gain matrix applied when \mathbf{s} is close to \mathbf{s}^* , and $\alpha \geq 0$ and $\beta \in]0, 1]$ are two scalar parameters such that, as the task error norm $\|\mathbf{s}^* - \mathbf{s}\|$ increases, Λ exponentially decreases (with slope dependent on α) to $\beta \Lambda^*$, for very large task error. This exponential trend compensates that of the error signal, thus generating a less variable control input $\dot{\mathbf{q}}$, as will be shown by the experiments. The values of α , β , and Λ^* are tuned empirically, so that the robot joint velocities stay roughly constant during operation.

In the next section, we first define the reference frames and the main variables of the framework and then, for each of three sensor-based tasks (position, vision and force), we give the expression of \mathbf{s} and that of the corresponding Jacobian \mathbf{J} .

3. Sensor-based tasks

3.1. Definitions

The reference frames used in our work are (see Fig. 1): the robot base (B), camera (C), end effector (E), and image (I) frames. Reference frame B is fixed in the world, whereas C, E and I move with the robot. The pose of A in frame B is defined as: ${}^B\mathbf{p}_A = [{}^B\mathbf{t}_A, {}^B\theta\mathbf{u}_A]^T \in \mathbb{SE}(3)$, with ${}^B\theta\mathbf{u}_A$ the angle/axis vector [37].

For the camera, we use the normalized perspective model. A 3D point with coordinates $({}^cX, {}^cY, {}^cZ)$ in the camera frame, projects in the image as a 2D point with coordinates:

$$x = \frac{{}^cX}{{}^cZ}, \quad y = \frac{{}^cY}{{}^cZ}. \quad (28)$$

We assume that the pose of the camera in the end effector, ${}^E\mathbf{p}_C$, is constant and known through a preliminary calibration step.

For human–robot collaboration, we use $n = 3$ tasks: positioning, visual, and force task. Each one has dimension $k = 6$, in order to control all 6 degrees of freedom of the end effector. We will hereby detail each task.

3.2. Positioning task

The objective of positioning is to control the end effector pose in the base frame. Hence, the positioning task is:

$$\mathbf{s}_p = {}^B\mathbf{p}_E. \quad (29)$$

This can be estimated at each iteration, by applying the robot forward kinematics to the measured articular variables, \mathbf{q} .

For this task, the Jacobian in (21) is simply:

$$\mathbf{J}_p = \frac{\partial {}^B\mathbf{p}_E}{\partial \mathbf{q}}. \quad (30)$$

This Jacobian can be computed, at run time, by applying the technique presented in [28].

3.3. Visual task

The objective of the visual task, is to drive the end effector to a desired pose with respect to a visible target. To this end, we apply the two and one-half-dimensional (2 1/2 D) visual servo paradigm originally introduced in [38]. This method combines the advantages of image-based and position-based visual servoing schemes, while trying to avoid their shortcomings [17]. In fact, the task is defined by a combination of image features and 3D characteristics:

$$\mathbf{s}_v = [x \quad y \quad \log {}^cZ \quad {}^c\theta\mathbf{u}_C]^T. \quad (31)$$

In this equation, x and y are the image coordinates of the target characterized by (28), cZ is the target depth in the camera frame, and ${}^c\theta\mathbf{u}_C$ gives the relative rotation between the current and desired poses of the camera.

The Jacobian corresponding to the 2 1/2 D task is [38]:

$$\mathbf{J}_v = \mathbf{L}_s {}^c\mathbf{V}_B \frac{\partial {}^B\mathbf{p}_C}{\partial \mathbf{q}}. \quad (32)$$

In this expression, \mathbf{L}_s is the interaction matrix relating the task evolution to the camera velocity in frame C:

$$\mathbf{L}_s = \begin{bmatrix} \mathbf{L}_{11}(x, y, {}^cZ) & \mathbf{L}_{12}(x, y) \\ \mathbf{0} & \mathbf{L}_{22}({}^c\theta\mathbf{u}_C) \end{bmatrix}, \quad (33)$$

while ${}^c\mathbf{V}_B$ is the spatial motion transform matrix from frame B to frame C:

$${}^c\mathbf{V}_B = \begin{bmatrix} {}^c\mathbf{R}_B & [{}^c\mathbf{t}_B]_{\times} {}^c\mathbf{R}_B \\ \mathbf{0} & {}^c\mathbf{R}_B \end{bmatrix}. \quad (34)$$

The complete expressions of \mathbf{L}_{11} , \mathbf{L}_{12} , and \mathbf{L}_{22} are given in [17], and $[\mathbf{t}]_{\times}$ is the skew-symmetric matrix associated with vector \mathbf{t} . Jacobian \mathbf{J}_v can be calculated at each iteration, since \mathbf{L}_s depends on \mathbf{s} , ${}^c\mathbf{V}_B$ on the pose of B in C (determined via forward kinematics ${}^B\mathbf{p}_E$ plus constant known ${}^E\mathbf{T}_C$), and $\partial {}^B\mathbf{p}_C / \partial \mathbf{q}$ can be calculated again using the technique presented in [28].

3.4. Force task

The objective of force control is to regulate the external wrench \mathbf{h} (force and torque vectors \mathbf{f} and \mathbf{m}), at the contact point between robot and human, to a desired value. This is essential to guarantee safe interaction with the environment and with the human operator. Without loss of generality, in this work, such external wrench is expressed in the end effector frame E.

To realize the force task, we apply an admittance controller [24], where the deviation of the end effector motion due to the interaction with the environment is related to the contact wrench, through an equivalent mass–spring–damper system with adjustable parameters.

Here, we consider a simple spring system, with null mass and damping, and positive definite diagonal square stiffness matrix \mathbf{K} , such that:

$${}^E\mathbf{h}_E - {}^E\mathbf{h}_E^* = -\mathbf{K}({}^E\mathbf{p}_E - {}^E\mathbf{p}_E^*) = \mathbf{K} {}^E\mathbf{p}_E^*. \quad (35)$$

Then, the force task is defined as: $\mathbf{s}_f = {}^E\mathbf{h}_E$. Deriving the above equation yields the Jacobian corresponding to this task:

$$\mathbf{J}_f = -\mathbf{K} \frac{\partial {}^E\mathbf{p}_E}{\partial \mathbf{q}}. \quad (36)$$

Having defined \mathbf{K} , \mathbf{J}_f can again be calculated with the technique from [28].

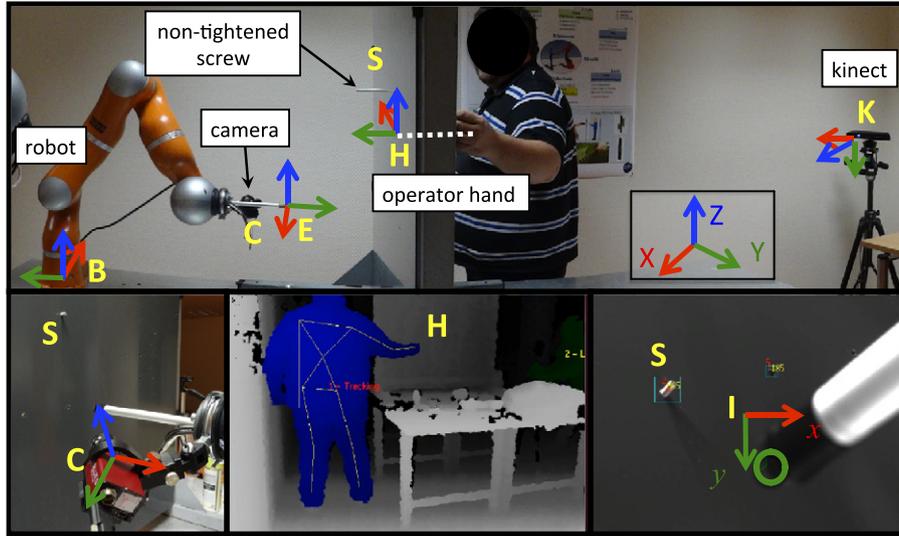


Fig. 2. Collaborative screwing case study. Top: experimental setup. Bottom left: view of the camera and end effector. Bottom centre: Kinect image. Bottom right: camera image.

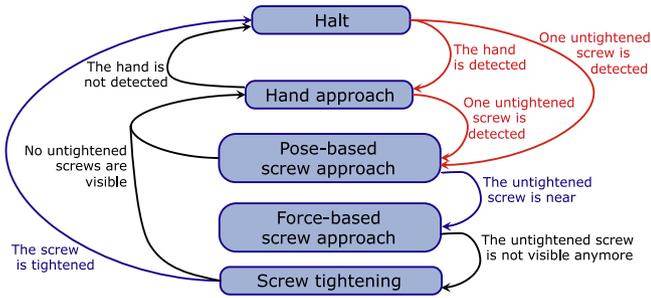


Fig. 3. Collaborative screwing state machine, selecting the appropriate control mode, according to the sensed data. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4. A case study: collaborative screwing

4.1. Experimental setup and assumptions

To validate our controller, we focus on a case study, where a robot aids a human operator in a screwing operation. In Fig. 2, we show the setup, along with the frames defined in Section 3.1; the screw is denoted S .

Human and robot operate on the opposite sides of a flank, where a series of screws must be inserted. The required operations are respectively:

- for the human: to insert the screws in the holes,
- for the robot: to tighten a bolt on each of the inserted screws, while the human maintains it on the flank.

Since the focus here is mainly on our multimodal control framework, we do not implement the physical screwing action; instead, we consider a screw to be tightened, when the end effector touches it with proper alignment. To realize the collaborative screwing operation, we utilize a Kinect, that outputs an RGB-D image of the work scene from a fixed pose, and a black and white camera mounted on the robot. These sensors are respectively dedicated to detecting and tracking the human hand motion, and to tracking newly inserted screws on the flank. Finally, to properly align end effector and screw, an estimation of the external forces is necessary. This force estimation will be explained in Section 5.

Our work assumptions are that the flank is perpendicular to the Y axis of the base frame, at known distance from B , and that the Kinect pose in the base frame has been coarsely calibrated.

To avoid luminosity variations in the image, we maintain the camera orientation with respect to the flank constant throughout operation. Since ${}^E\mathbf{p}_C$ is constant, we have decided to do this by keeping the end effector perpendicular to the flank, with the axes of frame E placed as in Fig. 2. Hence, we impose the desired rotation matrix from end effector to base to be:

$${}^B\mathbf{R}_E^* = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{bmatrix}. \quad (37)$$

In the rest of this section, we will detail the strategy that has been used to realize collaborative screwing, with controller (25).

4.2. Multimodal control strategy

To realize the collaborative screwing task, we utilize four modes, and halting, which simply consists in setting $\dot{\mathbf{q}} = \mathbf{0}$. The Jacobian used in (25) is:

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_p \\ \mathbf{J}_v \\ \mathbf{J}_f \end{bmatrix}, \quad (38)$$

with \mathbf{J}_p , \mathbf{J}_v , and \mathbf{J}_f defined respectively in (30), (32) and (36).

The modes are operated by the state machine in Fig. 3. As the figure shows, the transitions can be activated either by detection (red) or loss (black) of sensed information, or by success of the mode (blue). The detection/loss of information is determined by sensors processing. Instead, a mode is successful when:

$$\sum_{i=1}^6 w_i \|s_i^* - s_i\| < \sigma, \quad (39)$$

with $\mathbf{w} = [w_1 \dots w_6] \in \mathbb{R}^6$ a vector of positive weights, and σ a scalar threshold.

Our complete framework is summarized in Fig. 4. In the rest of this section, we will focus on each of the four modes, by specifying the selection matrices \mathbf{S}_p , \mathbf{S}_v and \mathbf{S}_f , the desired task vector \mathbf{s}^* , and the activation condition (39).

4.3. Hand approaching mode

If the human operating hand is detected by the Kinect, its position is fed to a controller that moves the robot so that the

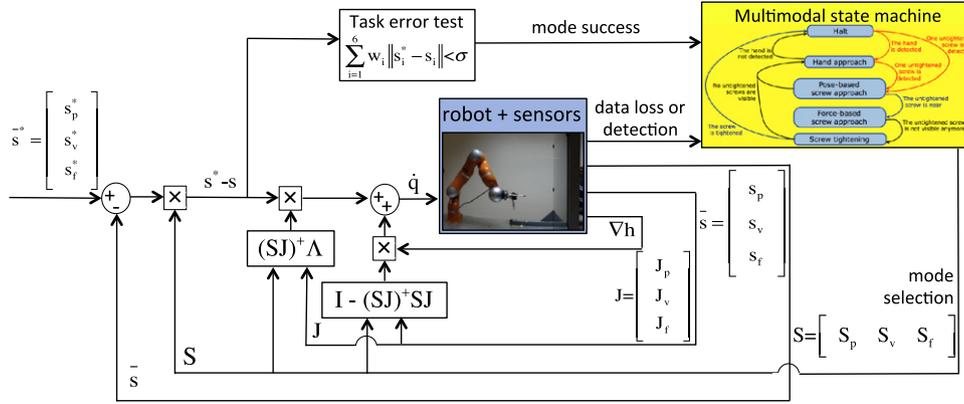


Fig. 4. Multimodal framework block diagram.

camera has a good view of the area where the human is operating. Since only the positioning task (29) is necessary, the selection matrices are:

$$\mathbf{S}_p = \mathbf{I} \quad \mathbf{S}_v = \mathbf{S}_f = \mathbf{0}. \quad (40)$$

From (3) and (29), we can infer the desired task vector:

$$\mathbf{s}^* = \mathbf{S}_p \mathbf{s}_p^* = {}^B \mathbf{p}_E^* \in \mathbb{SE}(3). \quad (41)$$

To derive \mathbf{s}_p^* , we introduce the Kinect (K), and operating hand (H) frames (see Fig. 2). The origin of H is the orthogonal projection of the operator hand on the flank, while its orientation is that of B. The hand position in the Kinect frame is estimated using OpenNI,³ and then orthogonally projected on the flank⁴ to obtain ${}^K \mathbf{X}_H$, which can then be transformed to ${}^B \mathbf{X}_H$ in the base frame. Then, our aim is to place H at a fixed desired position in the camera frame:

$${}^C \mathbf{X}_H^* = [{}^C X_H^* \quad {}^C Y_H^* \quad {}^C Z_H^* \quad 1]^\top, \quad (42)$$

to increase the chances of visualizing in the image the future inserted screw. Also, since we set ${}^H \mathbf{R}_B = \mathbf{I}$, ${}^B \mathbf{R}_E^*$ according to (37), and since ${}^C \mathbf{R}_E$ is constant and known, the desired orientation of the camera with respect to the hand, ${}^H \mathbf{R}_C^*$ can be derived. Combining ${}^C \mathbf{X}_H^*$ and ${}^H \mathbf{R}_C^*$, we can obtain the desired camera to hand transformation, ${}^H \mathbf{T}_C^*$. This can now be used to determine ${}^B \mathbf{T}_E^*$, and therefore \mathbf{s}_p^* , to be used in (25), along with:

$$\mathbf{S} = [\mathbf{I} \quad \mathbf{0} \quad \mathbf{0}]. \quad (43)$$

Since the hand approaching mode should be activated and deactivated only by perceived data, the task convergence need not be monitored, and we can set $\sigma = 0$ and $\mathbf{w} \neq \mathbf{0}$, so that (39) is never true in this mode.

4.4. Pose-based screw approaching mode

If a screw is detected in the image (see Fig. 2, bottom right), its position determines s_v according to (31), so that the end effector is driven in front of it. To this end, we exploit the screw position as viewed from the on-board camera (to infer x and y), along with the measures of the robot articular positions for forward kinematics (to infer ${}^C \mathbf{Z}$ and ${}^C \theta_{\mathbf{u}_C}$). The details on the image processing algorithms used to detect and track the screws are given in [27].

The shift from hand to screw approaching can lead to abrupt joint accelerations. This problem, which is common when switching between manipulation primitives, has been recently tackled

using on-line trajectory generation [39]. Here, we exploit homotopy to better manage the transition. We define $t > 0$ the *screw age* (i.e., the time since it has been detected). The visual task selection matrix is then designed to smoothly vary from $\mathbf{0}$ to \mathbf{I} , as t tends to a tuned scalar T :

$$\mathbf{S}_v = \begin{cases} \frac{1 - \cos(\pi t/T)}{2} \mathbf{I} & \text{if } t < T, \\ \mathbf{I} & \text{otherwise.} \end{cases} \quad (44)$$

The other task selection matrices are set to:

$$\mathbf{S}_p = \mathbf{I} - \mathbf{S}_v, \quad \mathbf{S}_f = \mathbf{0}, \quad (45)$$

so that, in controller (25):

$$\mathbf{S} = [\mathbf{I} - \mathbf{S}_v \quad \mathbf{S}_v \quad \mathbf{0}]. \quad (46)$$

In practice, the visual task is gradually activated by \mathbf{S}_v , while concurrently the hand position task is deactivated by \mathbf{S}_p .

As proved in Section 2, the advantage of our framework is that such a smooth transition can be easily implemented without compromising the controller stability, since, in contrast with hybrid sensors control, the selection matrices do not have to be binary. This is also a fundamental advantage with respect to the method proposed in [39].

From (3), we can infer the desired task vector:

$$\mathbf{s}^* = \begin{cases} (\mathbf{I} - \mathbf{S}_v) \mathbf{s}_p^* + \mathbf{S}_v \mathbf{s}_v^* & \text{if } t < T, \\ \mathbf{s}_v^* & \text{otherwise.} \end{cases} \quad (47)$$

As mentioned, \mathbf{s}^* varies from a task dependent on both hand and screw, to purely vision-based screw approaching task \mathbf{s}_v^* , that we define as:

$$\mathbf{s}_v^* = [x_S^* \quad y_S^* \quad \log {}^C Z_S^* \quad \mathbf{0}]^\top. \quad (48)$$

This \mathbf{s}_v^* corresponds to driving the screw to image position (x_S^*, y_S^*) at desired depth ${}^C Z_S^*$, while zeroing the orientation error between C and C^* .

Let us now explain how \mathbf{s}_v^* is derived. The image position (x_S^*, y_S^*) (circle in Fig. 2, bottom right) is set so that end effector and screw are aligned at the end of this mode. We set the end effector Cartesian position to have a desired translation with respect to the screw:

$${}^E \mathbf{t}_S^* = [0 \quad 0 \quad {}^E Z_S^*]^\top, \quad (49)$$

so that ${}^E Z_S^* > 0$ is as small as possible, without end effector occlusion. Then, from the known ${}^C \mathbf{T}_E$, and from ${}^E \mathbf{X}_S^*$, we can derive ${}^C \mathbf{X}_S^*$, and, from that: ${}^C Z_S^*, x_S^* = {}^C X_S^*/{}^C Z_S^*$, and $y_S^* = {}^C Y_S^*/{}^C Z_S^*$. For rotations, as usual we servo ${}^B \mathbf{R}_E^*$ according to (37). Then, ${}^C \theta_{\mathbf{u}_C}$ can be calculated from known ${}^C \mathbf{R}_E$ and ${}^E \mathbf{R}_B^*$.

³ <http://www.openni.org>

⁴ It is trivial to derive the flank plane equation in K from ${}^B \mathbf{T}_K$ and ${}^B \mathbf{Y}_H$.

The following mode (force-based screw approaching), is triggered when the visual error with respect to the screw is small enough. Hence, we set:

$$\mathbf{w} \neq \mathbf{0} \quad \forall t > 0, \quad (50)$$

$$\begin{cases} \sigma = 0 & \text{if } t < T, \\ \sigma > 0 & \text{otherwise,} \end{cases}$$

so that (39) is verified only after time T , when the hand task is deactivated.

4.5. Force-based screw approaching mode

Once the screw is near enough, force control is activated, to make the end effector compliant in case of contact, while advancing. We activate this mode just before contact, because, in the absence of external contacts, the force signal to noise ratio can lead to inaccurate positioning.

The desired wrench on the end effector, in the end effector frame, is:

$${}^E \mathbf{h}_E^* = [0 \quad 0 \quad {}^E f_{E,Z}^* \quad 0 \quad 0 \quad 0]^\top. \quad (51)$$

Through force control (35), ${}^E f_{E,Z}^* < 0$ makes the end effector progress forward. All other components are zeroed to make the end effector compliant.

As long as the screw is visible, the end effector can be driven towards it by using visual control. Then, the task selection matrices are:

$$\mathbf{S}_p = \mathbf{0} \quad \mathbf{S}_v = \begin{bmatrix} \mathbf{I}_2 & \mathbf{0}_{2 \times 4} \\ \mathbf{0}_{4 \times 2} & \mathbf{0}_4 \end{bmatrix} \quad \mathbf{S}_f = \begin{bmatrix} \mathbf{0}_2 & \mathbf{0}_{2 \times 4} \\ \mathbf{0}_{4 \times 2} & \mathbf{I}_4 \end{bmatrix}. \quad (52)$$

Therefore, in controller (25):

$$\mathbf{S} = \begin{bmatrix} \mathbf{0}_{2 \times 6} & \mathbf{I}_2 & \mathbf{0}_{2 \times 6} & \mathbf{0}_{2 \times 4} \\ \mathbf{0}_{4 \times 6} & \mathbf{0}_{4 \times 2} & \mathbf{0}_{4 \times 6} & \mathbf{I}_4 \end{bmatrix}, \quad (53)$$

and the desired task is:

$$\mathbf{s}^* = \mathbf{S}_v \mathbf{s}_v^* + \mathbf{S}_f \mathbf{s}_f^* = [x_s^* \quad y_s^* \quad {}^E f_Z^* \quad 0 \quad 0 \quad 0]^\top. \quad (54)$$

The transition from this mode to the following is triggered by the loss of the screw, when it is too near to be visible in the image. Then, we set $\sigma = 0$ and $\mathbf{w} \neq \mathbf{0}$, so that (39) is never true.

4.6. Screw tightening mode

When the screw is so near that it is not visible any more, the last mode is activated. This relies solely on force control:

$$\mathbf{S}_p = \mathbf{0} \quad \mathbf{S}_v = \mathbf{0} \quad \mathbf{S}_f = \mathbf{I}. \quad (55)$$

Therefore, in controller (25):

$$\mathbf{S} = [\mathbf{0} \quad \mathbf{0} \quad \mathbf{I}], \quad (56)$$

and the desired force task is:

$$\mathbf{s}^* = \mathbf{s}_f^* = [0 \quad 0 \quad {}^E f_{E,Z}^* \quad 0 \quad 0 \quad 0]^\top. \quad (57)$$

Clearly, if tightening was also to be realized (although this is not the case here, as mentioned in Section 4.1), the desired moment around Z , ${}^E m_{E,Z}^*$ should also be non-null. To verify that the screw is tightened, we check the force error according to (39), with tuned weights $\mathbf{w} \neq \mathbf{0}$ and threshold $\sigma > 0$.

5. Experiments

To validate our framework, we have run a series of experiments with a lightweight KUKA LWR IV robot in the scenario illustrated in Fig. 2. Since a tightening tool is not mounted on the end effector, we have used a cylindrical tool of external diameter 14 mm to verify the precision of our method. The LWR is redundant with respect to the end effector operational space dimension (it has $j = 7$ degrees of freedom, whereas $k = 6$). Thus, we use the extra degree of freedom to guarantee joint limit avoidance. To this end, in (25), we use a scalar, configuration dependent, cost function [40]:

$$h(\mathbf{q}) = \frac{1}{2} \sum_{i=1}^7 \left(\frac{q_i - q_{i,mid}}{q_{i,M} - q_{i,m}} \right)^2, \quad (58)$$

with $[q_{i,m}, q_{i,M}]$ the available range for joint i and $q_{i,mid} = (q_{i,M} + q_{i,m})/2$ its midpoint. The values of $\dot{\mathbf{q}}$ computed via (25) are fed to the Reflexxes online trajectory generation library⁵ for smoothing. To get the interaction wrench ${}^E \mathbf{h}_E$, instead of mounting a force sensor on the end effector, we have decided to use the estimated external wrench signal provided by the robot controller through the FRI Interface.⁶ The camera mounted on the end effector is a Stingray F201B from Allied Vision Technologies, with resolution 1024×768 pixels, and we used the ViSP library [41] for visualization purposes. The image processing pipeline takes approximately 60 ms. Thus, although the skeleton processing on the Kinect is slightly faster, we fix the control loop rate at 15 Hz.

To highlight the novel contributions of our framework, i.e., force control, and the use of homotopy and adaptive gains, various experiments were run. These are shown in the video attached to this paper (see Appendix A).

In a preliminary experiment, (see Fig. 5), three screws are touched with the tip of the tool, using only the hand approaching (HA) and pose-based screw approaching (SA) modes. In this experiment, the homotopy between these two modes is deactivated, and the gain matrix Λ is independent from \mathbf{s} . In Fig. 6, we have plotted the components of the error $\mathbf{e} = \mathbf{s}^* - \mathbf{s}$ (top) and of the joint velocities $\dot{\mathbf{q}}$ (bottom). The numbers correspond to the inserted screws (1 to 3). It is clear from the curves that the transitions between modes are abrupt in terms of $\dot{\mathbf{q}}$. This is because homotopy and adaptive gains are not used.

Let us now focus on the second, complete experiment (see Fig. 7), where all modes, as well as homotopy and adaptive gains, were applied. This time, we verify that the cylindrical tool successfully encircles new, non-tightened screws (see photo on the left of Fig. 7). A high accuracy is required, since the screw external diameter and tool internal diameters are respectively 5 and 9 mm. Although, the specifications are more strict than in the previous experiment, force control facilitates the insertion, by correcting slight orientation errors. Hence, we are confident that, with a tightening tool, the approach should also work. Convergence of the hand approaching and screw approaching modes has been discussed just above. Let us now focus on the final mode, when force control intervenes. In Fig. 8, we have ${}^E f_{E,Z}$ and ${}^B Y_E$ during this final mode. The plots start at time $t = 60$ s, when the tool comes into contact with the flank. Correspondingly, ${}^E f_{E,Z}$ decreases from the null value, until the desired value is reached (in (51), we set ${}^E f_{E,Z}^* = -25$ N). Then, at $t \approx 65$ s, the end effector stops (see bottom graph). After a few seconds, a user (see snapshots in Fig. 7 and video) moves by hand the robot last joint. The forces are detected (see ${}^E f_{E,Z}$ in Fig. 8), and admittance control induces the small variations of ${}^B Y_E$. This experiment shows that the framework is capable of force stabilization, for safe human–robot interaction.

⁵ www.reflexxes.com.

⁶ <http://cs.stanford.edu/people/tkr/fri/html/>.

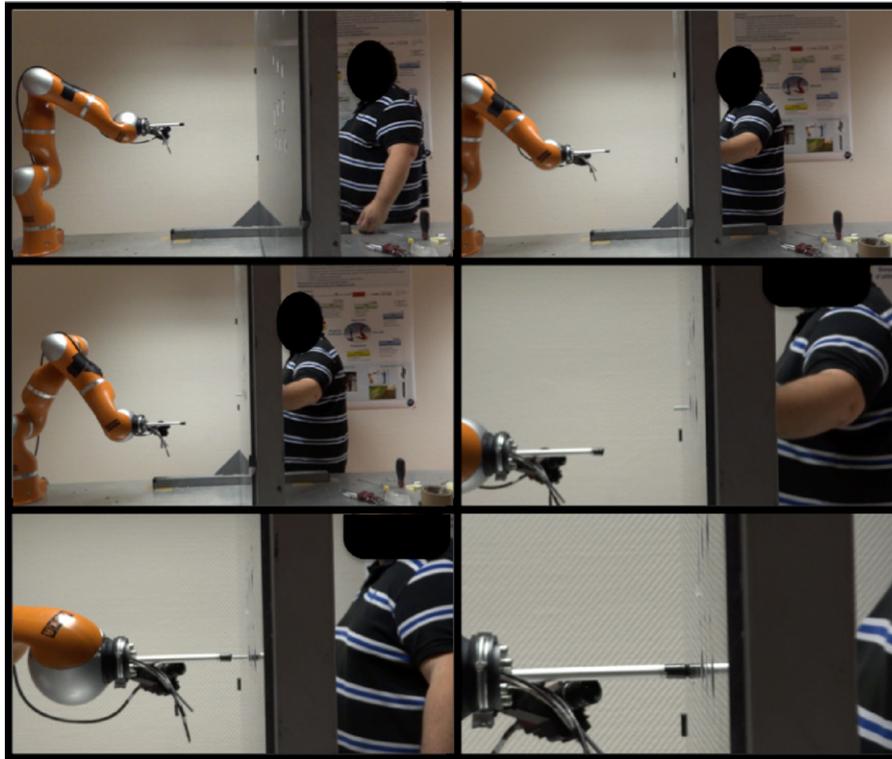


Fig. 5. Six consecutive snapshots of the first experiment of collaborative screwing.

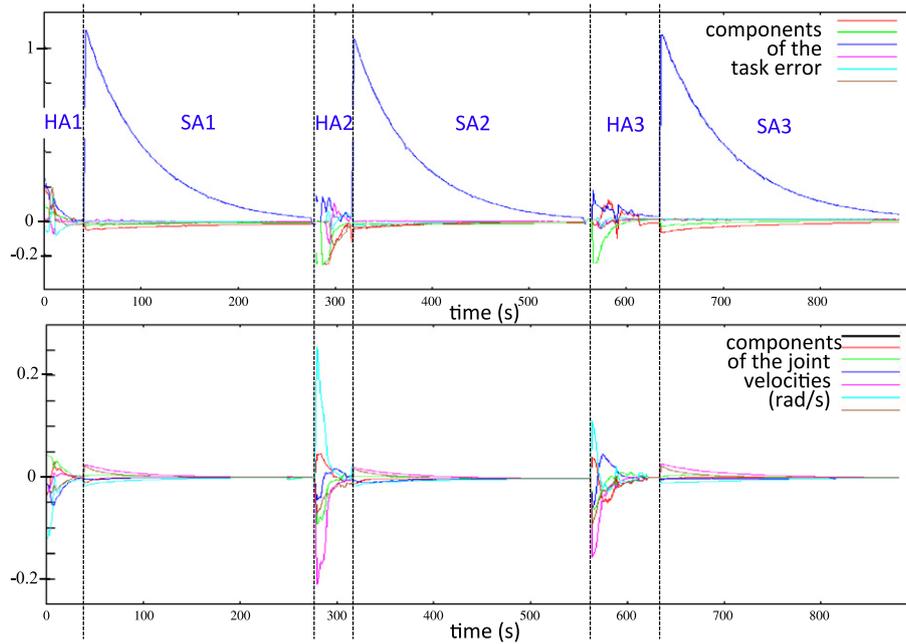


Fig. 6. The six components of the error $\mathbf{e} = \mathbf{s}^* - \mathbf{s}$ (top) and the seven components of the joint velocities $\dot{\mathbf{q}}$ (bottom) during the first experiment.



Fig. 7. Left: the tightening task. Right: consecutive snapshots of the second experiment.

Finally, we ran experiments with and without the adaptive gains for Λ (experiments noted AG and FG), and with and without the smooth homotopy from hand to screw approaching (noted

H and NH). To compare the experiments, in Fig. 9, we have plotted the norm of the joint velocity $|\dot{\mathbf{q}}|$, for three experiments: NH+FG, NH+AG, and H+AG. Since the change mainly concerns the hand

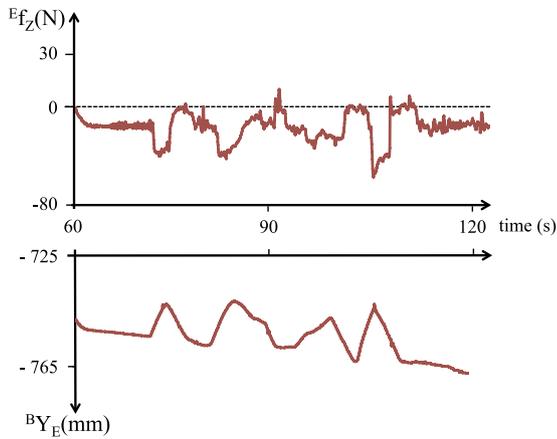


Fig. 8. Evolution of E_f^Z (top) and $^B Y_E$ (bottom) over time, during force-based screw approach and tightening.

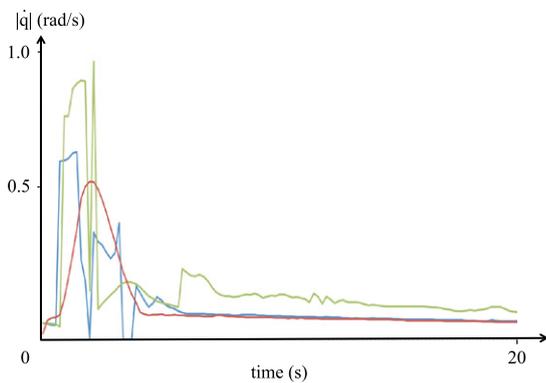


Fig. 9. Evolution of $|\dot{q}|$ over time, during hand and screw approaching, using NH+FG (green), NH+AG (cyan), and H+AG (red). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

and the beginning of screw approaching, we have only plotted the curves for these phases. The mode change is visible in the plots, and occurs after approximately 5 s. The joint velocity norm has been chosen, since it is a good indicator of the movement smoothness. In the first experiment (NH+FG, green curve), which reproduces the approach used in [27], strong variations appear during the HA mode; these come from the variability of the positioning error $\mathbf{s}_p^* - \mathbf{s}_p$, due to hand motion. Replacing the fixed gain with an adaptive one yields the cyan curve. As the curves show, the use of an adaptive gain, reduces the shaky motion. Another improvement is obtained by adding a homotopy of duration $T = 1$ second when the screw is seen. In this case (red curve), the transition from hand to screw tracking is much smoother. To further demonstrate the smooth transitions obtained thanks to the adaptive gains and homotopies, in Fig. 10, we have compared the joint velocity components obtained with NH+FG (left), and with H+AG (right). As the figure shows, the curves are much smoother, and less control effort is required, with the new approach. We have also

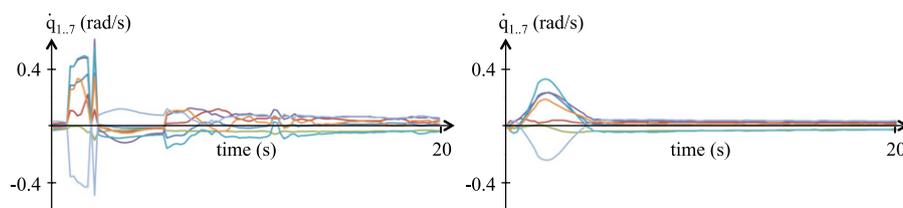


Fig. 10. Evolution of \dot{q} components over time, during hand and screw approaching, using NH+FG (left), and H+AG (right). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 1
Comparison between $|\dot{q}|$ and $|\ddot{q}|$ in the four configurations.

Configuration	FG	AG	NH	H
$ \dot{q} $ (rad s ⁻¹)	0.091	0.067	0.079	0.079
$ \ddot{q} $ (rad s ⁻²)	0.132	0.082	0.145	0.068

compared the average values of $|\dot{q}|$ and $|\ddot{q}|$ over each experiment. These, shown in Table 1, confirm the cited properties. For both metrics, AG outscores FG, by realizing the same operation in the same time with less velocity and acceleration. Also, as expected, homotopy reduces $|\ddot{q}|$ (from 0.145 to 0.068 rad s⁻²) since it realizes a smoothing effect on the joint velocities, but has no influence on $|\dot{q}|$. Reducing $|\ddot{q}|$ is crucial for safe human–robot interaction, since most robot safety metrics (see [42]), depend on accelerations measured at impacts.

In summary, the approach with both homotopy and adaptive gains (H+AG) should be selected. The advantages are numerous: less energy is required, the motion is smoother (facilitating image processing), and faster operation can be obtained. In fact, although, for the purpose of these comparisons, the gains were all tuned so that the duration of the experiments be the same, in other experiments we have fine tuned the gains of H+AG, to achieve screw approaching in 40 s, i.e., approximately 80% faster than in [27].

6. Conclusions

In this paper, we have generalized the multimodal framework for human–robot interaction originally introduced in [27]. The generalized framework can operate by activating or deactivating various tasks, according to the sensed data and to the needs of the application. This is of particular interest when numerous sensing devices are to be used for control, as is often the case in HRI. Typically, in this work, we have applied the framework to a collaborative screw tightening experiment, where vision, Kinect, position and force data must be alternatively controlled. To avoid abrupt accelerations, important features such as adaptive gains and homotopy are included in the framework.

This preliminary work opens numerous avenues for future research. In the future, we plan to use our framework for full human–robot cooperation, with direct physical interaction. It would then be possible to verify its robustness to unknown dynamic parameters, resulting from the interaction with the human body (e.g., arm and hand).

Acknowledgement

Work supported by the ANR grant no. ANR-10-CORD-0025 (French National Agency) ICARO project.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at <http://dx.doi.org/10.1016/j.robot.2015.03.002>.

References

- [1] A. Bicchi, M. Peshkin, J. Colgate, Safety for physical human–robot interaction, in: B. Siciliano, O. Khatib (Eds.), Springer Handbook of Robotics, Springer, 2008, pp. 1335–1348.
- [2] A. De Santis, B. Siciliano, A. De Luca, A. Bicchi, An atlas of physical human–robot interaction, *Mech. Mach. Theory* 43 (3) (2008) 253–270.
- [3] A. De Luca, F. Flacco, Integrated control for pHRI: Collision avoidance, detection, reaction and collaboration, in: IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics BIOROB, 2012.
- [4] L. Sentis, O. Khatib, A whole-body control framework for humanoid operating in human environments, in: IEEE Int. Conf. on Robotics and Automation ICRA, 2006.
- [5] M.S. Erden, A. Billard, End-point impedance measurements at human hand during interactive manual welding with robot, in: IEEE Int. Conf. on Robotics and Automation ICRA, 2014.
- [6] X. Li, C.C. Cheah, Human-guided robotic manipulation: theory and experiments, in: IEEE Int. Conf. on Robotics and Automation ICRA, 2014.
- [7] M.S. Erden, B. Maric, Assisting manual welding with robot, *Robot. Comput.-Integr. Manuf.* 27 (2011) 818–828.
- [8] G. Grunwald, G. Schreiber, A. Albu-Schaffer, G. Hirzinger, Touch: The direct type of human interaction with a redundant service robot, in: IEEE Int. Workshop on Robot and Human Interactive Communication, ROMAN, 2001.
- [9] M.S. Erden, T. Tomiyama, Human intent detection and physically-interactive control of a robot without force sensors, *IEEE Trans. Robot.* 26 (2) (2010) 370–382.
- [10] O. Khatib, E. Demircan, V. DeSapio, L. Sentis, T. Besier, S. Delp, Robotics-based synthesis of human motion, *J. Physiol. (Paris)* 103 (3–5) (2009) 211–219.
- [11] Microsoft Corporation, 1 Microsoft Way, Redmond, WA 98052-7329, USA, Microsoft Kinect homepage. <http://xbox.com/Kinect> (accessed: Aug. 19, 2014), Internet, 2014.
- [12] Y. Maeda, T. Hara, T. Arai, Human–robot cooperative manipulation with motion estimation, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS, 2001, vol. 4, pp. 2240–2245.
- [13] L. Villani, J. De Schutter, Force Control, in: B. Siciliano, O. Khatib (Eds.), Springer Handbook of Robotics, Springer, 2008, pp. 161–184.
- [14] V. Lippiello, B. Siciliano, L. Villani, Interaction control of robot manipulators using force and vision, *Int. J. Optomechatronics* 2 (3) (2008) 257–274.
- [15] F. Flacco, T. Kröger, A. De Luca, O. Khatib, A depth space approach to human–robot collision avoidance, in: IEEE Int. Conf. on Robotics and Automation ICRA, 2012, pp. 338–345.
- [16] J.A. Corrales, G.J. Garcia Gomez, F. Torres Medina, V. Perdereau, Cooperative tasks between humans and robots in industrial environments, *Int. J. Adv. Robot. Syst.* 9 (94) (2012) 1–10.
- [17] F. Chaumette, S. Hutchinson, Visual servo control, *IEEE Robot. Autom. Mag.* 13 (4) (2006) 82–90. vol. 14, no. 1, 2007, pp. 109–118.
- [18] S.M. La Valle, Planning Algorithms, Cambridge University Press, 2006.
- [19] A. De Santis, V. Lippiello, B. Siciliano, L. Villani, Human–robot interaction control using force and vision, *Adv. Control Theory Appl.* 353 (2007) 51–70.
- [20] B.J. Nelson, Improved force control through visual servoing, in: American Control Conference, 1995.
- [21] K. Hosoda, K. Igarashi, M. Asada, Adaptive hybrid visual servoing/force control in unknown environment, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS, 1996.
- [22] M.H. Raibert, J.J. Craig, Hybrid position/force control of manipulators, *ASME J. Dynamic Systems, Measurement, and Control* 103 (1981) 126–133.
- [23] J. Baeten, H. Bruyninckx, J. De Schutter, Integrated vision/force robotic servoing in the task space formalism, *Int. J. Robot. Res.* 22 (10–11) (2003) 941–954.
- [24] N. Hogan, Impedance control: an approach to manipulation: parts I–III, *ASME J. Dynamic Systems, Measurement, and Control* 107 (1985) 1–24.
- [25] G. Morel, E. Malis, S. Boudet, Impedance based combination of visual and force control, in: IEEE Int. Conf. on Robotics and Automation ICRA, 1998.
- [26] M. Prats, P.J. Sanz, A.P. Del Pobil, Vision-tactile-force integration and robot physical interaction, in: IEEE Int. Conf. on Robotics and Automation ICRA, 2009.
- [27] A. Cherubini, R. Passama, A. Meline, A. Crosnier, P. Fraise, Multimodal control for human–robot cooperation, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS, 2013.
- [28] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, *Robotics: Modelling, Planning and Control*, Springer, 2009.
- [29] N. Mansard, O. Stasse, P. Evrard, A. Kheddar, A versatile generalized inverted kinematics implementation for collaborative working humanoid robots: The stack of tasks, in: Int. Conf. on Advanced Robotics, ICAR, 2009.
- [30] J. De Schutter, T. De Laet, J. Rutgeerts, W. Decré, R. Smits, E. Aertbelin, K. Claes, H. Bruyninckx, Constraint-based task specification and estimation for sensor-based robot systems in the presence of geometric uncertainty, *Int. J. Robot. Res.* 26 (5) (2007) 433–455.
- [31] N.M. Ceriani, A.M. Zanchettin, P. Rocco, A. Stolt, A. Robertsson, A constraint-based strategy for task-consistent safe human–robot interaction, in: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems IROS, 2013.
- [32] M. Prats, P.J. Sanz, A.P. Del Pobil, Reliable non-prehensile door opening through the combination of vision, tactile and force feedback, *Auton. Robots* 29 (2) (2010) 201–218.
- [33] G. Antonelli, F. Arrichiello, S. Chiaverini, The NSB control: a behavior-based approach for multi-robot systems, *Paladyn J. Behav. Robot.* 1 (1) (2010) 48–56.
- [34] N. Mansard, O. Khatib, A. Kheddar, A unified approach to integrate unilateral constraints in the stack of tasks, *IEEE Trans. Robot.* 25 (3) (2009).
- [35] J. Park, J. Lee, N. Mansard, Intermediate desired value approach for task transition of robots in kinematic control, *IEEE Trans. Robot.* 28 (6) (2012) 1260–1277.
- [36] L. Saab, O. Ramos, F. Keith, N. Mansard, P. Soueres, J.-Y. Fourquet, Dynamic whole-body motion generation under rigid contacts and other unilateral constraints, *IEEE Trans. Robot.* 29 (2) (2013) 346–362.
- [37] K. Waldron, J. Schmiedeler, Kinematics, in: B. Siciliano, O. Khatib (Eds.), Springer Handbook of Robotics, Springer, 2008, pp. 9–33.
- [38] E. Malis, F. Chaumette, S. Boudet, 2-1/2D visual servoing, *IEEE Trans. Robot. Autom.* 15 (2) (1999) 238–250.
- [39] T. Kröger, B. Finkemeyer, Robot motion control during abrupt switchings between manipulation primitives, in: Workshop on Mobile Manipulation at the IEEE Int. Conf. on Robotics and Automation ICRA, 2011.
- [40] A. Liegeois, Automatic supervisory control of configurations and behavior of multibody mechanisms, *IEEE Trans. Syst. Man Cybern.* 7 (6) (1977) 868–871.
- [41] E. Marchand, F. Spindler, F. Chaumette, ViSP for visual servoing: a generic software platform with a wide class of robot control skills, *IEEE Robot. Autom. Mag.* 12 (4) (2005) 40–52. Special issue on “Software packages for vision-based control of motion”.
- [42] J. Versace, A review of severity index, in: Proceedings of Stapp Car Crash Conference, 1971, pp. 149–170.



Andrea Cherubini received the 'Laurea' degree (MSc) in Mechanical Engineering in 2001 from the University of Rome 'La Sapienza' and a second MSc in Control Systems in 2003 from the University of Sheffield, UK. From 2004 to 2007, he was a PhD student at the Dipartimento di Informatica e Sistemistica, University of Rome 'La Sapienza', under the supervision of Prof. Giuseppe Oriolo. From March to October 2007, he worked at INRIA Rennes with Dr. François Chaumette. He received the Ph.D. degree in Systems Engineering in April 2008. Then, from 2008 to 2011, he returned as post-doctoral fellow at INRIA Rennes.

He is currently Associate Professor at Université de Montpellier 2, and researcher at LIRMM.



Robin Passama is a permanent CNRS research engineer at LIRMM since December 2012. His main research theme is software engineering for the design and development of real-time robot control architectures. He received an M.Sc. in Computer Science in 2002 and then a Ph.D. in Computer Science and Robotics in 2006 from the Université Montpellier 2. After his Ph.D. he was involved in different European and national projects and developed applications in fields such as functional electrical stimulation, industrial robotic, fault-tolerant mobile robotics and medical robotics.



Philippe Fraise received M.Sc degree in Electrical Engineering from Ecole Normale Supérieure de Cachan in 1988. He received Ph.D. degree in Automatic Control in 1994. He is currently Professor at the University of Montpellier, France. He is the head of robotics department (LIRMM) and co-chair of French National Workgroup (GDR Robotique) working on Humanoid Robotics (GT7). He is also member of JRL-France scientific board (Japanese–French joint Laboratory for Robotics, AIST-JRL) and member of IEEE. His research interests include modelling and control applied to robotic and rehabilitation fields, including humanoid

robotics, robotics for rehabilitation.



André Crosnier, is currently Professor at the University of Montpellier and Research Staff member of the Robotic Department of the Laboratoire d'Informatique, de Robotique et de Micro-électronique de Montpellier. He received the Agrégation, in Electrical Engineering, from the Ecole Normale Supérieure de Cachan, in 1982 and his Ph.D. in Robotics, from the University of Montpellier, in 1987. His research interests include physical human–robot interaction, haptic and geometric modelling of the environments.



Full length Article

Collaborative manufacturing with physical human–robot interaction



Andrea Cherubini ^{a,*}, Robin Passama ^a, André Crosnier ^a, Antoine Lasnier ^b, Philippe Fraise ^a

^a Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier LIRMM, Université de Montpellier - CNRS, 161 Rue Ada, 34392 Montpellier, France

^b Department of Manufacturing Engineering, PSA Peugeot Citroën, Route de Gisy, 78140 Vélizy Villacoublay, France

ARTICLE INFO

Article history:

Received 18 June 2015

Received in revised form

13 December 2015

Accepted 17 December 2015

Keywords:

Cobots

Industrial robotics

Human–robot interaction

Reactive and sensor-based control

ABSTRACT

Although the concept of industrial *cobots* dates back to 1999, most present day hybrid human–machine assembly systems are merely weight compensators. Here, we present results on the development of a collaborative human–robot manufacturing cell for homokinetic joint assembly. The robot alternates active and passive behaviours during assembly, to lighten the burden on the operator in the first case, and to comply to his/her needs in the latter. Our approach can successfully manage direct physical contact between robot and human, and between robot and environment. Furthermore, it can be applied to standard position (and not torque) controlled robots, common in the industry. The approach is validated in a series of assembly experiments. The human workload is reduced, diminishing the risk of strain injuries. Besides, a complete risk analysis indicates that the proposed setup is compatible with the safety standards, and could be certified.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

The concept of *cobots*, i.e., robots collaborating with human workers in manufacturing assembly lines, dates back to the pioneer work [1]. In fact, cobots – designed for the assembly line worker – can reduce ergonomic concerns that arise due to on-the-job physical and cognitive loading, while improving safety, quality and productivity. This is a key issue, since according to statistics of the Occupational Safety and Health Department of the US Department of Labour,¹ more than 30% of European manufacturing workers are affected by lower back pain, leading to enormous social and economic costs. A thorough state-of-the-art on human–machine cooperation in manufacturing lines is provided in [2]. At the time of that survey (2009), the only hybrid assembly systems in manufacturing processes were weight compensators/balancers. However, the authors clearly point out the need for more advanced collaboration: although humans remain indispensable in many assembly operations, ergonomic tools assisting their duties are fundamental.

In this paper, we focus on a target application, proposed by PSA (Peugeot Citroën) in the frame of the French National Project ANR ICARO. The application is the assembly of an Rzeppa homokinetic joint, an operation that is currently done manually in the PSA line,

causing muscular pain to the workers. In this work, we propose a novel, collaborative human–robot design, of this cell.

The main contributions of this work are outlined below.

- In contrast with most existing human–machine manufacturing applications, where collision avoidance is guaranteed by a minimum security distance [2], our framework successfully manages direct physical contact between robot and human, and between robot and environment.
- In our design, the robot alternates active and passive behaviours during assembly, to lighten the burden on the operator in the first case, and to comply to his/her needs in the latter.
- In contrast with most similar works, our approach can be applied to standard position (and not torque) controlled robots, common in the industry.

From the end user (PSA) viewpoint, two aspects are particularly noteworthy. First, since the operator load is reduced by approximately 60%, the proposed assembly cell can be reclassified in the PSA ergonomics scale. Second, a complete risk analysis by PSA indicates that the proposed setup is compatible with the safety standards, and could be certified.

The article is organized as follows. Section 2 summarizes the state-of-the-art in collaborative manufacturing, and highlights our contributions in the context of current, related research. In Section 3, we present the targeted application: collaborative assembly of a homokinetic joint. The proposed framework is outlined in Section 4. The framework components (nominal trajectory generation, admittance control, and safety monitoring) are then detailed in the

* Corresponding author.

E-mail addresses: andrea.cherubini@lirmm.fr (A. Cherubini), robin.passama@lirmm.fr (R. Passama), andre.crosnier@lirmm.fr (A. Crosnier), antoine.lasnier@mpsa.com (A. Lasnier), philippe.fraise@lirmm.fr (P. Fraise).

¹ www.osha.eu.int

following sections (respectively, Sections 5–7). Experimental results are reported in Section 8, and finally summarized in the Conclusion.

2. Related work

This section summarizes the current state-of-the-art in collaborative manufacturing. We first review the more application-oriented research on human–machine cooperation (Section 2.1), and then the academic research on physical human–robot interaction (Section 2.2).

2.1. Research on human–machine cooperation in the industry

The authors of [3] provide a very rich overview of the emerging technologies in automotive assembly, including the supporting systems (mainly the information technologies). They show that mass customization requires high technological flexibility, and propose various designs to cope with this, by integrating both automated and human-based assembly. A similar perspective is that of the recent EU project ROBO-PARTNER [4], that aims at integrating assembly systems, and human capabilities. The main enablers, according to the authors, are: intuitive interfaces, safe strategies and equipment, proper methods for planning and execution, and the use of mobile robots, and of distributed computing. More recently, the U.S. Consortium for Automotive Research conducted a study on feasibility of fenceless robotic cells for automotive applications [5]. The study defines the levels of human–robot collaboration according to the cell complexity, to drive the probabilities of successful implementation. But as in the previously cited survey [2], the paper exposes the absence of high level human–robot collaboration, apart from “Intelligent Lift Assistants”.

Although some automotive manufacturers are gradually introducing robots in their human production line [6,7], a crucial question persists: how should a collaborative robotic cell be designed? Various researchers have looked into this. Papakostas et al. [8] discuss the key features of cooperating robotic cells in automotive assembly, and provide simulated comparisons of two scenarios: a conventional welding robotic cell, and one with co-operating robots. The authors of [9] assess five alternative safety designs, covering both hardware and control design, of a human–robot collaboration prototype cell for cable harness assembly. In [10], a new cell production assembly system, with human–robot cooperation is developed. The system consists of three key technologies; parts feeding by double manipulators on a mobile base, production process information support for the operator, and safety management for cooperation between operator and robot. The main target of [11] is safety of the shared work cell, in the absence of physical fences between human and robot. Since safety options provided by basic infrared sensors are limited, the authors design a network architecture of these sensors, for tracking user positions, while avoiding collisions. The authors of [12] propose a method for optimizing task distribution among workers and robots. The method is validated, using an ABB Dual Arm Concept Robot, in a PLC Input/Output module assembly scenario.

2.2. Research on physical human–robot collaboration

Recent robotics research focuses on the study and characterization of physical human–robot interaction (pHRI [13,14]). The goal is to enable close collaboration between human and robot, in all service and industrial tasks, that require the adaptability of humans to be merged with the high performance of robots in terms of precision, speed and payload [15]. In this context, it

becomes indispensable to define safety and dependability metrics [16–19]. These can contribute to the definition of standards, such as the recent ISO 10218-1:2011 “Safety requirements for industrial robots”.²

In this line of research, many solutions for realizing safe collaborative tasks have been explored in recent years. Although these solutions have not yet been transferred to the industry, we hereby list some of the most relevant theoretical works. In [20], a deformation-tracking impedance control strategy is designed to enable robot interaction with environments of unknown geometrical and mechanical properties. For successful interaction with unknown environments and operators, the robot should behave in a human-like manner. This is the target of the research in [21,22]: a human-like learning controller is designed, to minimize motion error and effort, during interaction tasks. Simulations show that this controller is a good model of human–motor adaptation, even in the absence of direct force sensing. A robust controller for a collaborative robot in the automotive industry, is extended in [23], to manage not only the interaction between an industrial robot and a stiff environment, but also human–robot–environment and human–robot–human–environment interactions.

Other researchers have focused more on industrial applications. For example, an industrial robot controller, incorporating compliance of the joints with the environment, is presented in [24]. The desired pose of the tool center point is computed from the force error. Parallel control considers a reference trajectory while allowing feedforward in force controlled directions. Although the method is designed for industrial assembly tasks, it does not take into account the presence of a human in the loop. In contrast, Erden and colleagues [25–27] have thoroughly studied an industrial task that directly involves a human operator, i.e., manual welding. In [25], a physically interactive controller is developed for a manipulator robot arm: the human applies forces on the robot, to make it behave as he/she likes. Then, a manual welding assistant robot is presented in [26]: as the human controls the welding direction and speed, the robot suppresses involuntary vibrations (e.g., caused by novice welders). The results show a considerable improvement in the welders performance when they are assisted. Finally, [27] presents a study of end-point impedance measurement at human hand, with professional and novice welders. The results support the hypothesis that impedance measurements could be used as a skill level indicator, to differentiate the welding performance levels. Although the welding assistance application targeted by these works also falls in the shared workplace paradigm evoked in [2], it differs from the one treated here, since the robot motion is driven by the human worker. Instead, in our work, the robot is active and autonomous during various phases of the assembly cycle. For the same reason, robot programming by demonstration/teaching is also out of scope here.

Other works similar to ours, but targeting manually guided robot operation, are presented in [28,29]. In [28], an operator teaches tasks to a robotic manipulator, by manually guiding its end effector. For this, the authors design a virtual tool, whose dynamics the operator should feel when interacting with the robot. An admittance controller driven by the measurements of a force/torque sensor is designed to ensure the desired virtual dynamic behaviour. The second paper addresses the problem of controlling a robot arm, executing a cooperative task with a human, who guides the robot through direct physical interaction. This problem is tackled by allowing the end effector to comply according to an impedance control law [30] defined in the Cartesian space. Redundancy ensures the stability of the coupled human–robot system, through inertial decoupling at the end effector. However, in

² www.iso.org/iso/catalogue_detail.htm?csnumber=51330

contrast with our work, the robot is torque (not position) controlled, and an impedance (instead of admittance) controller is used.

As outlined in most of the cited works, an efficient cobot must interact easily with the worker, even if s/he is non-trained. To this end, many pHRI works rely on the physical contact (touch) between robot and operator [31]. More recently, to guarantee interaction even in the absence of direct contact, researchers have proposed the use of pointing gestures [32], as well as the integration of vision with force [33–35]. Also, in our work, interaction includes both vision and force. In fact, vision stops/triggers robot operation, in case of danger, while the operator contact forces are used to start and end assembly, and to deform the robot trajectory, for collision avoidance.

3. Application: collaborative assembly of a homokinetic joint

The application that is targeted in this work is the collaborative human–robot assembly of a car homokinetic joint. Homokinetic joints allow a drive shaft to transmit power to a second shaft through a variable angle, at constant rotational speed, while limiting friction and play. They are used both in front wheel drive cars, and in rear wheel drive cars with independent rear suspension.

Specifically, the homokinetic joint that we use in this work is the Rzeppa joint, which was invented in 1926 by Alfred Hans Rzeppa (1885–1965). The joint works in a manner similar to a bevel gear, with steel balls in place of teeth. The working principle and exploded diagram of this joint are shown in Fig. 1.

The joint consists of a spherical star-shaped *inner race* with six grooves in it, and a similar enveloping outer shell (*housing*). Each groove guides one steel spherical ball. The half shaft fits in the centre of the inner race, which itself nests inside a circular *cage*. The cage is spherical but with open ends, and with six openings around the perimeter, one for each ball. Finally, the housing is attached to the second shaft.

In this work, we propose a collaborative strategy for the assembly of the Rzeppa joint. In fact, after thorough discussions with the PSA ergonomists, and analysis of the most difficult cells in the PSA process, it appears that this use-case generates more Musculoskeletal Disorders (MSDs) than other assembly cells. This is what motivated the choice of this cell, as case study. In particular, we focus on the insertion of the six balls in the joint. This task is currently done manually, using an *insertion tool* similar to the one that is shown, along with all the joint components, in the left of Fig. 2. The ball insertion operation that is currently done in the PSA production process is outlined on the right of the same figure. The ball order is predefined, since, to limit the mechanical constraints, the first three balls must be inserted, so that there is an empty groove between any two of them.

Let us hereby describe the steps of the manual assembly operation. Prior to ball insertion (i.e., in the previous work cell), the inner race, cage, and housing have been assembled, and a first ball has been inserted. This ball links the cage and inner race, leaving

some backlash, which will diminish as the other balls are placed. Hence, for simplicity, in the rest of this paragraph, the term cage will be used to refer to the linked grouping of cage and inner race. At first, the operator receives the partially assembled Rzeppa joint, and places it in an ad-hoc support. Then, by tapping on the cage with the insertion tool (see Fig. 2(1)), he slightly inclines it, until the second ball can be inserted (Fig. 2(2)). Since the backlash diminishes, for inserting the third and following balls, tapping is not sufficient anymore, and the operator must use the insertion tool to increase the lever arm needed to incline the cage (Fig. 2(3)). The insertion tool is designed so that its lower part fits precisely in the cage. The cage inclination (or *cage opening*), followed by ball insertion is repeated for the remaining balls (Figs. 2(4)–2(8)). Once all six balls have been placed, the cage is closed (i.e., reset to the flat position, Fig. 2(9)), so that the Rzeppa joint can be sent to the following cell in the production chain.

An experienced operator executes this entire operation very quickly, to guarantee the required production rate of 70 joints per hour. However, the cage opening and closing steps can provoke MSDs. These injuries are due to the forces required for cage positioning, opening, and closing, and to the repetition of such actions. Ergonomists estimate that over an eight hour shift, the operator lifts a total weight of approximately 5 tons, and realizes 18 000 upper limb movements [36]. Therefore, this part of the operation should be automated to alleviate the operator. On the other hand, the action of ball insertion by itself requires very high precision and adaptability skills, which are not attainable by present-day industrial robots. Thus, the characteristics of the Rzeppa joint ball insertion work cell make it an ideal scenario for robotics research.

In practice, one should automate the cell according to the following specifications:

- (1) The human operator must position the five balls in the corresponding cage openings, with little effort and motion, so that the fatigue and chances of injury are minimized.
- (2) The physical interaction between human and environment (specifically, between the human hand and the steel parts) must be controlled, to guarantee *safety*.
- (3) The physical interaction between insertion tool and environment (specifically, the Rzeppa joint) must be controlled, to avoid blockage.
- (4) The cobot velocity must guarantee safety, i.e., it must comply with the ISO safety standard cited in Section 2.2 (tool center point velocity limited to 0.25 ms^{-1}).

To fulfil the above requirements, we have redesigned the manufacturing cell, as shown in Fig. 3 (top). The lower part of the pre-assembled Rzeppa joint (composed of the housing, cage, inner race, and first ball), is held by a gripper placed on the end effector of a manipulator robot. The insertion tool, instead, is fixed to a support that is rigidly constrained to the robot base. In contrast with the manual insertion operation shown in Fig. 2, most of the required movements will be carried out by the robot, with the human intervening only to position the balls. The scenario is

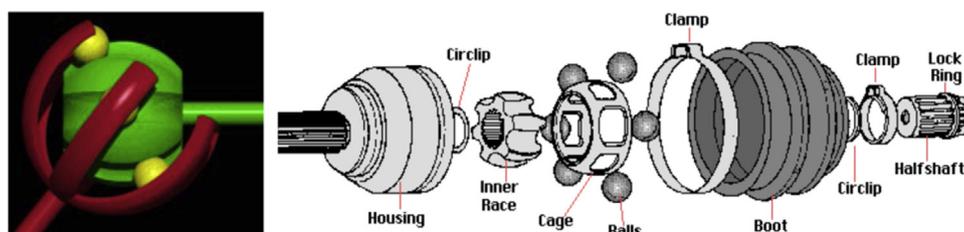


Fig. 1. Working principle (left) [https://en.wikipedia.org/wiki/Constant-velocity_joint] and exploded diagram (right) [www.aa1car.com] of the Rzeppa homokinetic joint.

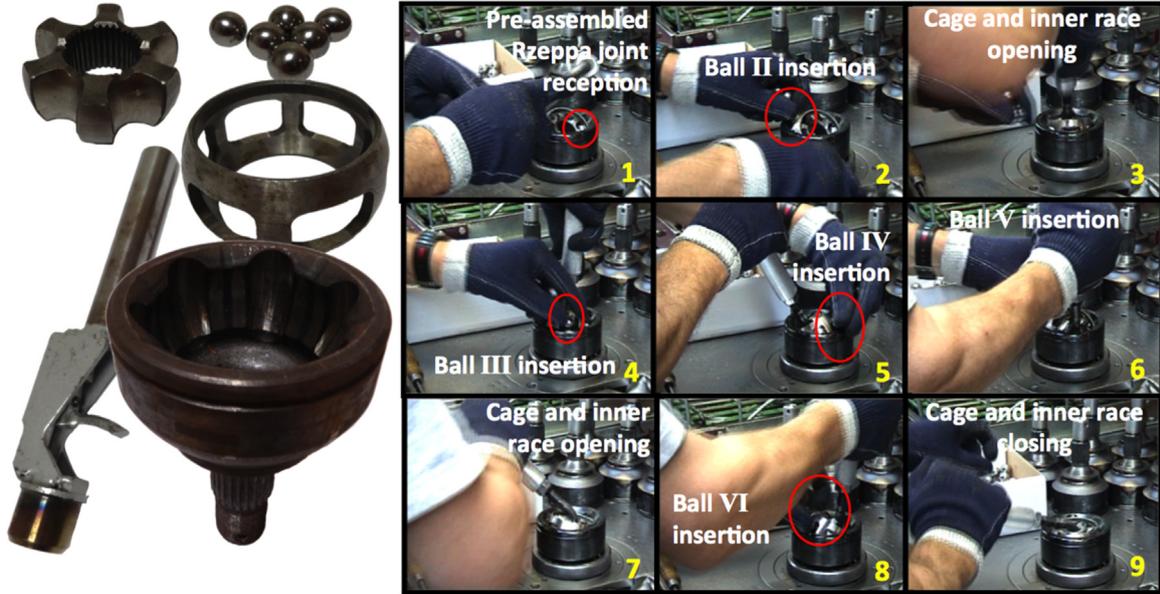


Fig. 2. Manual insertion. *Left*: concerned components (clockwise from top left: inner race, balls, cage, housing and insertion tool). *Right*: nine steps of the PSA manual insertion operation; ball 1 does not appear, since, along with the inner race, cage, and housing, it has been assembled before this operation.

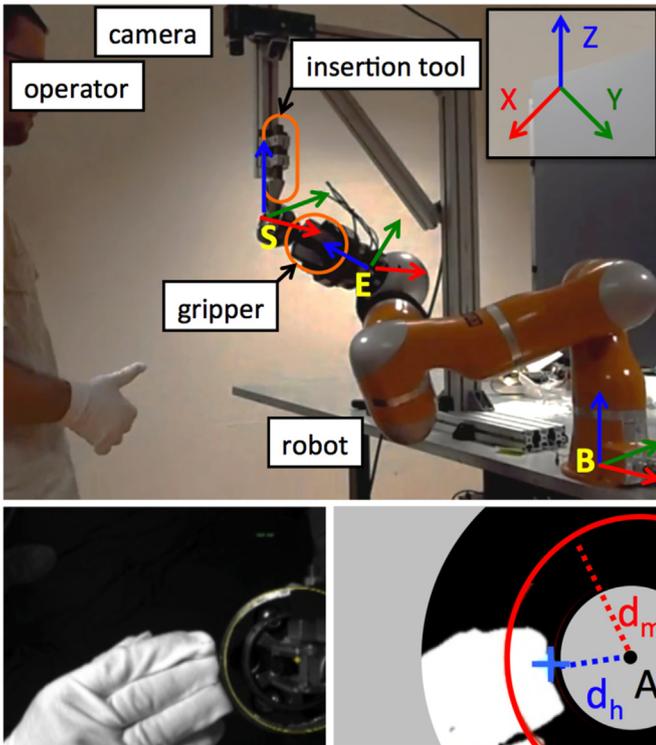


Fig. 3. *Top*: experimental setup for collaborative Rzeppa joint assembly with: insertion tool support (S), end effector (E), and robot base (B) reference frames. *Bottom left*: raw image from the camera. *Bottom right*: processed image with: locus of the tool axis in the image (A), distance between hand and tool (d_h), and minimum distance tolerated for safety (d_m). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

perfectly reversed, to make it *human-centered*: now the Rzeppa joint is displaced around the operator, instead of the opposite. To this end, we control the robot so that it opens/closes the cage – for ball insertion – by properly inclining it within the housing, and then pivots it around the vertical axis so that the operator is always facing the next desired opening. At the start (end) of the operation, the robot end effector raises (descends) to connect (disconnect) insertion tool and inner race. To realize all these motions (tool connection/

disconnection, cage opening/closing and pivoting), the robot follows nominal, pre-taught trajectories. These nominal trajectories are deformed to comply with the external force/torques from the environment and operator, using an admittance controller [30]. To further enhance safety of operation, the robot is stopped in the presence of strong force/torques, and a fixed camera, mounted on the insertion tool support, monitors the ball insertion. The camera is used both to detect ball insertion, and to stop the robot if it may endanger the operator hands. Since there is no risk of clamping fingers in the back of the joint (solid steel of the housing, without openings), the camera only monitors the front of the cage. Images from the camera are shown at the bottom of Fig. 3 (raw image on the left, and processed image on the right).

In the following, we provide more details on the control framework developed for collaborative assembly of the Rzeppa joint.

4. Definitions and characteristics of the control framework

4.1. Definitions and assumptions

The reference frames used to describe the assembly task are shown in Fig. 3. In this, and in all other figures of the paper, the RGB convention is used: axes X, Y and Z are respectively red, green and blue. The frames used in this work are: the robot base (\mathcal{F}_B), insertion tool support (\mathcal{F}_S), and end effector (\mathcal{F}_E) frames, with origins respectively B, S and E. Reference frames \mathcal{F}_B and \mathcal{F}_S are fixed in the world, whereas \mathcal{F}_E moves with the robot. The pose of \mathcal{F}_S is determined via an estimation procedure explained in Section 5. The pose of a generic frame A in frame \mathcal{F}_B is defined as: ${}^B\mathbf{p}_A = [{}^B\mathbf{t}_A, {}^B\theta\mathbf{u}_A]^\top \in \mathbb{SE}(3)$, with ${}^B\mathbf{t}_A$ the translation vector from B to A, and ${}^B\theta\mathbf{u}_A$ the angle/axis vector [37].

In this work, we consider a manipulator with $j \geq 6$ degrees of freedom (dof), and note $\mathbf{q} \in \mathbb{R}^j$ the robot joint values.

We also assume that it is possible to estimate (either through direct, or through joint torque measurement) the wrench (force ${}^E\mathbf{f}_E$ and torque ${}^E\mathbf{m}_E$) applied to the end effector, and expressed in the end effector frame: ${}^E\mathbf{h}_E = [{}^E\mathbf{f}_E, {}^E\mathbf{m}_E]^\top$. This information will be fed to an admittance controller [30], to adapt the robot motion, and avoid blockage between the Rzeppa inner race and the insertion tool.

4.2. Inverse kinematics controller

In our framework, an inverse kinematics controller is used to map end effector poses to the joint space. This is necessary for initially generating the nominal trajectory, and then for deforming it, according to the admittance controller. We hereby recall the classic inverse kinematics formulation according to [38].

The evolution of the end effector pose can be expressed as:

$${}^B\dot{\mathbf{p}}_E = \mathbf{J}\dot{\mathbf{q}}, \quad (1)$$

with $\mathbf{J} = \frac{\partial {}^B\mathbf{p}_E}{\partial \mathbf{q}}$ the *task Jacobian*, of size $6 \times j$, that is derived from the robot measured configuration $\hat{\mathbf{q}}$, and $\dot{\mathbf{q}}$ the robot joint velocities.

The end effector pose can be regulated to a desired value ${}^B\mathbf{p}_E^*$ and, if $j > 6$, redundancy [39] exists. Then, a secondary task can be realized, by exploiting the $j - 6$ extra degrees of freedom (since only 6 are required by the primary task, ${}^B\mathbf{p}_E^*$). The joint velocity $\dot{\mathbf{q}}$ for driving ${}^B\mathbf{p}_E$ to ${}^B\mathbf{p}_E^*$, is generated via:

$$\dot{\mathbf{q}} = \mathbf{J}^+ \Lambda ({}^B\mathbf{p}_E^* - {}^B\mathbf{p}_E) + (\mathbf{I} - \mathbf{J}^+ \mathbf{J}) \nabla g. \quad (2)$$

In the above equation:

- \mathbf{J}^+ is the $j \times 6$ right pseudoinverse of \mathbf{J} , such that $\mathbf{J}\mathbf{J}^+ = \mathbf{J}$. We assume that \mathbf{J} is full row rank during operation, so that this pseudoinverse can be calculated. This was the case throughout

the experiments and is a common assumptions in inverse kinematics control [40].

- Λ is a positive definite 6-dimensional square diagonal matrix that determines the convergence rate of ${}^B\mathbf{p}_E$ to ${}^B\mathbf{p}_E^*$.
- The term $\nabla g = \frac{\partial g}{\partial \mathbf{q}}$ is introduced to minimize the scalar cost function $g(\mathbf{q})$ to perform the secondary task.

It is well-known that system (1), controlled by (2), is globally asymptotically stable with respect to the pose regulation task. Indeed, plugging (2) into (1) yields:

$${}^B\dot{\mathbf{p}}_E = \Lambda ({}^B\mathbf{p}_E^* - {}^B\mathbf{p}_E), \quad (3)$$

and, since Λ is a positive definite diagonal matrix, ${}^B\mathbf{p}_E = {}^B\mathbf{p}_E^*$ is a stable equilibrium for the closed-loop system. Also, note that minimization of g has no effect on the task dynamics.

4.3. Control framework

Our framework for collaborative assembly is shown in Fig. 4. The framework is made up of two steps (shown respectively at the top and bottom of the figure): the *offline* nominal trajectory generation step, and the *online* controller for collaborative assembly. We hereby outline the framework, and will then detail its components in Sections 5–7.

A *nominal trajectory* \mathcal{T} , corresponding to the entire assembly operation, is input to the controller. This is defined by a series of N

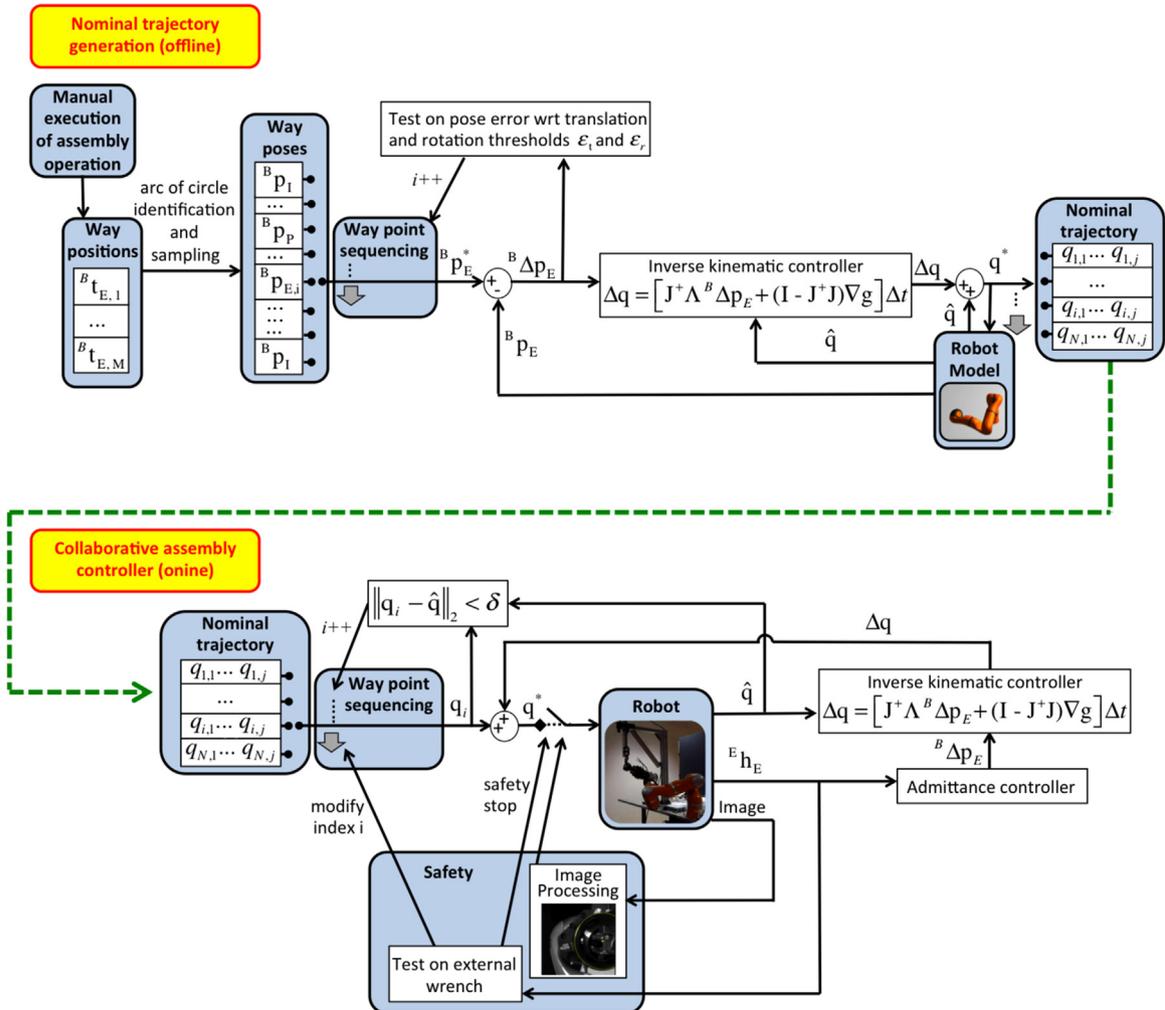


Fig. 4. Control framework for collaborative assembly. Top: nominal trajectory generation. Bottom: collaborative assembly controller.

way points in the joint space, that are to be followed by the robot:

$$\mathcal{T} = \{\mathbf{q}_1, \dots, \mathbf{q}_N\}$$

$$\mathbf{q}_i = [q_{i,1}, \dots, q_{i,j}]^T \in \mathbb{R}^j. \quad (4)$$

In (4), $q_{i,1}, \dots, q_{i,j}$ are the nominal joint angular values to be realized at iteration i . These are mapped from the operational space by the inverse kinematics controller, presented just above. Nominal trajectory (4) is identified offline through a procedure outlined at the top of Fig. 4, and detailed in Section 5.

Then, at run time, \mathcal{T} is adapted online to account for the physical interaction with the environment and operator. To this end, an *admittance controller* (see Section 6) shifts the end effector desired pose by ${}^B\Delta\mathbf{p}_E = {}^B\mathbf{p}_E^* - {}^B\mathbf{p}_E$, according to the external wrench (force and torque) on the end effector, expressed in the support frame, that we denote with ${}^S\mathbf{h}_E$. This pose variation is mapped, again by the inverse kinematics controller, to a joint correction $\Delta\mathbf{q}$, then applied to the current waypoint of \mathcal{T} , to obtain the actual joint reference \mathbf{q}^* for the robot motors. Index i is increased as soon as the reference trajectory way point \mathbf{q}_i has been reached, i.e., when $\|\mathbf{q}_i - \hat{\mathbf{q}}\|_2 < \delta$, with $\hat{\mathbf{q}}$ the joint measure, and δ an arbitrary scalar threshold (see ‘way point sequencing’ block in the bottom of Fig. 4).

Prior to this, *safety* monitoring is applied. This relies on two verifications. The first consists in checking the magnitude of the external wrench, whereas the second uses processed image data to monitor the operator hand. The safety module is detailed in Section 7.

5. Nominal trajectory generation

This section presents the generation of the nominal trajectory \mathcal{T} used by the control framework.

5.1. Characterization of the trajectory

Without loss of generality, we consider that the robot final joint value q_j is defined along the axis aligned with the end effector frame z -axis (positive q_j corresponds to a counter clockwise rotation around the z -axis). This assumption, which is consistent with the kinematics of most current day collaborative robots [41], facilitates the pivoting of the Rzeppa cage after each ball insertion. This pivoting will orient the joint, so that the groove where the next ball is to be inserted, will face the operator. The assembly operation can then be broken into five similar trajectories (one per ball), differing *only* by the orientation of the Rzeppa joint, i.e., of the end effector frame, with respect to the second-to-last joint

frame. These five orientations (shown at the top of Fig. 5) have been designed to respect the ball insertion order shown on the top left of the same figure.

The collaborative Rzeppa joint assembly operation that is presented in this paper is then composed of the following steps (shown in the bottom of Fig. 5).

1. The end effector E connects the inner race to the insertion tool. For this, E tracks a straight line segment in the operational space.
2. The robot moves the Rzeppa joint to enable insertion of each ball – the following steps are cycled 5 times (once for each ball).
 - (a) The joint is inclined so that the cage opens in front of the operator, and then the robot stays still until ball insertion (detected by the camera). For this, E tracks an arc of circle in the operational space (since the position of S and the distance between E and S are both constant).
 - (b) The joint is inclined so that the cage is closed. Again, E tracks an arc of circle in the operational space.
 - (c) The joint is brought back to the starting position of step 2, by having E track again an arc of circle in operational space.
 - (d) The last joint q_j is actuated to pivot the Rzeppa joint, so that the next cage opening will face the operator. Specifically (see Fig. 5, top), after the insertion of each ball II...VI, joint q_j must rotate, relatively to the previous value by:

$$\Delta q_j^{\text{II}} = -2\pi/3 \quad \Delta q_j^{\text{III}} = -\pi/3$$

$$\Delta q_j^{\text{IV}} = 2\pi/3 \quad \Delta q_j^{\text{V}} = 2\pi/3 \quad \Delta q_j^{\text{VI}} = -\pi/3. \quad (5)$$

The last rotation shift is necessary to realign the robot end effector with the initial configuration, to prepare the next assembly operation.

3. The end effector disconnects the inner race from the insertion tool. For this, E tracks a straight line segment in the operational space.

Each of these steps is obtained by executing a trajectory, and all are concatenated to obtain the complete nominal trajectory \mathcal{T} in (4). Defining these trajectories in the joint, rather than in the operational space, guarantees their feasibility (typically w.r.t. singularities and joint limits), and avoids errors due to inaccuracies in the encoder measurement and in the inverse kinematics computation. However, although technically defined in the joint space, the trajectories correspond to desired behaviours of the end effector in the operational space. The only exception is the trajectory in 2(d), which is best defined in the joint space, since it consists in controlling only the last joint value q_j to pivot the end effector ($q_{1,k} = q_{2,k} = \dots = q_{N,k}$, $k = 1, \dots, j - 1$). For the other steps, instead, the task is best defined in the operational space.

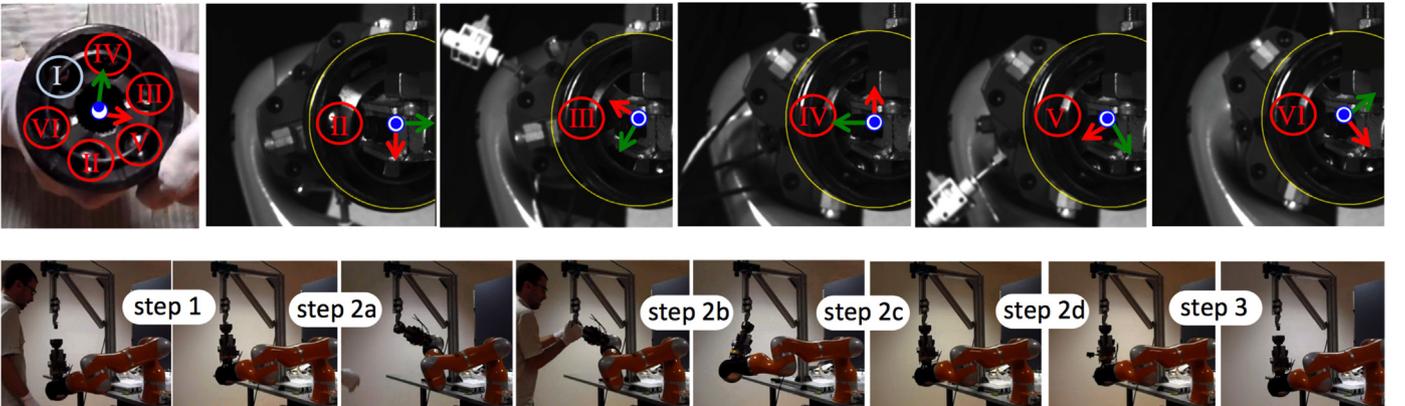


Fig. 5. Top: ball insertion order (left), and end effector frame configurations for inserting balls II...VI (right). Bottom: snapshots of the consecutive steps of the assembly operation (steps 2(a) to 2(d) are cycled 5 times, once for each ball).

5.2. Manual execution and identification in the operational space

The most delicate issue is the identification of the circular trajectory to be followed during steps 2(a)–2(c). This is crucial, since the Rzeppa joint is physically constrained to the insertion tool, during robot motion, and an inaccurate trajectory may lead to blockage. The trajectories for the other steps (1, 2(d), and 3) are derived from these circular trajectories. In fact: the trajectories for steps 1 and 3 are simple rectilinear connections to and from the arc of circle, without physical contact between the joint and the insertion tool, and the trajectory for 2(d) is constrained to the circle, with only orientation changes of the end effector. However, the equation of the circle in the robot base frame is related to the dimensions, configurations and poses of the insertion tool and of its support, which are very difficult to measure accurately. Thus, we have devised an identification procedure based on manually executing the nominal trajectory to the robot.

This procedure relies on the assumption that the robot is backdrivable (either via software or hardware). This is the case in most current day collaborative robots [41]. At first, the robot end effector is manually driven to the starting pose of step 2, noted P_{teach} . This corresponds to the insertion tool being connected to the inner race, with the Z axis of the end effector frame aligned with the insertion tool. Then, the operator moves the robot end effector back and forth a few times along the circular trajectory that is to be followed for ball insertion. The Cartesian positions of the end effector in the robot base frame are recorded throughout this operation:

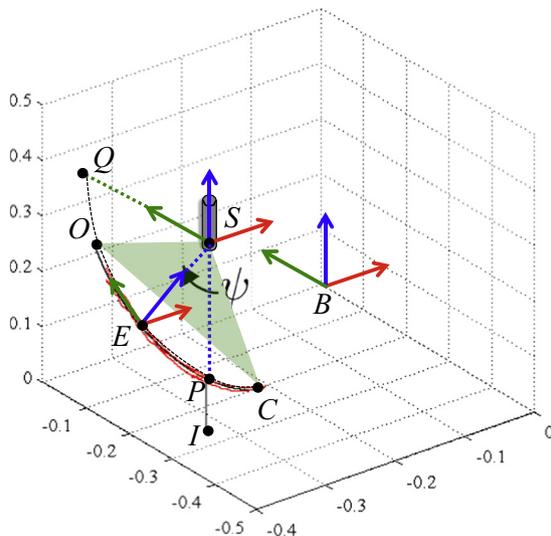
$$\{ {}^B \mathbf{t}_{E_1}, \dots, {}^B \mathbf{t}_{E_M} \}$$

$${}^B \mathbf{t}_{E_i} = \begin{bmatrix} {}^B X_{E_i} & {}^B Y_{E_i} & {}^B Z_{E_i} \end{bmatrix}^T \in \mathbb{R}^3. \quad (6)$$

This set of positions in the operational space, which is clearly noisy due to the shaky motion of the operator, is fed to an identification algorithm, coded in *Matlab*[®].

The results of the identification are shown on the left of Fig. 6. First, the sphere (radius R and center position ${}^B \mathbf{t}_S$ in the base frame) that is closest to the taught trajectory, is found, by solving with `lsqnonlin` the nonlinear least-squares problem:

$$\min_{R, {}^B \mathbf{t}_S} \sum_i \left[({}^B X_{E_i} - {}^B X_S)^2 + ({}^B Y_{E_i} - {}^B Y_S)^2 + ({}^B Z_{E_i} - {}^B Z_S)^2 - R^2 \right]^2. \quad (7)$$



The center of this sphere is set as the origin of the support frame, S . Second, the plane passing through S , and closest to the taught points is found, by solving, with `lsqlin`, the constrained linear least-squares problem:

$$\min_{a,b,c} \sum_i \left(a {}^B X_{E_i} + b {}^B Y_{E_i} + c {}^B Z_{E_i} - 1 \right)^2 \quad \text{subject to } a {}^B X_S + b {}^B Y_S + c {}^B Z_S = 1. \quad (8)$$

Here, a , b and c are the parameters of the plane equation $a {}^B X + b {}^B Y + c {}^B Z = 1$. Third, we define the start point of step 2, noted P for pivoting cage point. Point P is obtained by projecting P_{teach} first on the sphere (7), and then on the plane (8). This way, it is possible to define the \mathcal{F}_S frame axes: Z is parallel to line (SP) and pointing towards S , X is parallel to plane normal $[a \ b \ c]^T$ and pointing to the right of the operator, and Y completes the frame. Then, naming Q the intersection between the sphere and the Y axis of \mathcal{F}_S , the equation of the arc of circle in \mathcal{F}_B is:

$$\begin{cases} {}^B X(\psi) = {}^B X_S + {}^S X_P \cos \psi + {}^S X_Q \sin \psi \\ {}^B Y(\psi) = {}^B Y_S + {}^S Y_P \cos \psi + {}^S Y_Q \sin \psi \\ {}^B Z(\psi) = {}^B Z_S + {}^S Z_P \cos \psi + {}^S Z_Q \sin \psi, \end{cases} \quad (9)$$

with $\psi \in [\psi_m, \psi_M] \subset]-\pi, \pi]$. The two trajectory extrema ψ_m and ψ_M are manually tuned according to the setup. These correspond to points O (for open cage), and C (for closed cage). Finally, translating P away from the arc of circle, in the direction of decreasing ${}^S Z$, by the arbitrary length of the connecting trajectory, we derive the end effector initial pose, noted I . Having set a desired sampling interval on ψ , the four points I , P , O , and C , with equation (9), define the way poses of the end effector in the base frame for steps 1–3 for inserting *only* the first ball:

$$\{ {}^B \mathbf{p}_I, \dots, {}^B \mathbf{p}_P, \dots, {}^B \mathbf{p}_O, \dots, {}^B \mathbf{p}_P, \dots, {}^B \mathbf{p}_C, \dots, {}^B \mathbf{p}_P, \dots, {}^B \mathbf{p}_I \}. \quad (10)$$

For each pose, the end effector orientation ${}^B \theta_{E_i}$ is calculated so that the end effector frame Z axis passes through S , and the Y axis is tangent to the circle, so that the ball 2 cage opening is in front of the operator (as in the second snapshot of Fig. 5). The result of the described identification procedure is shown in the left of Fig. 6, where we show: in red the recorded taught positions ${}^B \mathbf{t}_{E_i}$, in gray the insertion tool, in green the identified plane (with the top vertex corresponding to S), and in black the sampled trajectory (10).

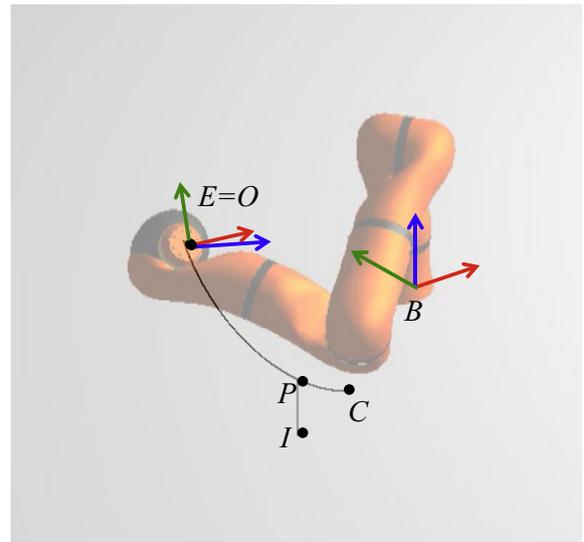


Fig. 6. Left: results of the trajectory identification with relevant way points I (initial), P (pivoting), O (open cage), and C (closed cage), and insertion tool (grey). Right: Simulations for replaying the identified path (black) using an inverse kinematics controller. In both figures, the base and end effector frames are also drawn. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

5.3. From the operational to the joint space trajectory

The following step consists in converting operational space trajectory (10) to the corresponding joint space trajectory, expressed as (4). For a general redundant robot, the inverse geometric model cannot be derived in closed form. Thus, we use the inverse kinematic pose controller (2) to track trajectory (10), as explained just below. First, we position the robot in the initial pose ${}^B\mathbf{p}_I$. Then, renaming ${}^B\mathbf{p}_{E_i}$ the ordered sample poses in (10), trajectory tracking consists in driving the end effector pose ${}^B\mathbf{p}_E$ to the next desired pose in the trajectory (${}^B\mathbf{p}_E^* = {}^B\mathbf{p}_{E_i}$), and incrementing i as soon as ${}^B\mathbf{p}_E^*$ is reached (i.e., as soon as $\|{}^B\mathbf{t}_E - {}^B\mathbf{t}_E^*\|_2 < \epsilon_t$ and $\|{}^E\mathbf{u}_E \sin {}^E\theta_E\|_2 < \epsilon_r$, with ϵ_t and ϵ_r two arbitrary scalar thresholds). The trajectory terminates when pose ${}^B\mathbf{p}_E^* = {}^B\mathbf{p}_C$ is reached, as in (10).

The joint values recorded while tracking (10), i.e.,

$$\{\mathbf{q}_I, \dots, \mathbf{q}_P, \dots, \mathbf{q}_O, \dots, \mathbf{q}_P, \dots, \mathbf{q}_C, \dots, \mathbf{q}_P, \dots, \mathbf{q}_I\}, \quad (11)$$

are then split in the following five primitive joint trajectories:

$$\begin{aligned} \mathcal{T}_1 &= \{\mathbf{q}_I, \dots, \mathbf{q}_P\}, \\ \mathcal{T}_{2a}^1 &= \{\mathbf{q}_P, \dots, \mathbf{q}_O\}, \\ \mathcal{T}_{2b}^1 &= \{\mathbf{q}_O, \dots, \mathbf{q}_P, \dots, \mathbf{q}_C\}, \\ \mathcal{T}_{2c}^1 &= \{\mathbf{q}_C, \dots, \mathbf{q}_P\}, \\ \mathcal{T}_3 &= \{\mathbf{q}_P, \dots, \mathbf{q}_I\}. \end{aligned} \quad (12)$$

The index of each trajectory coincides with the corresponding step, as listed at the beginning of this Section. For the five ball insertions, the last joint value q_j of the trajectories (12) must be shifted by the Δq values in (5). This corresponds to applying the following iterative rule:

$$\begin{cases} \mathcal{T}_{\text{STEP}}^{\text{BALL}+1} = \mathcal{T}_{\text{STEP}}^{\text{BALL}} \\ \quad + \{\mathbf{q}^{\text{BALL}}, \dots, \mathbf{q}^{\text{BALL}}\} \\ \mathcal{T}_{2d}^{\text{BALL}+1} = \{\mathbf{0}, \dots, \mathbf{q}^{\text{BALL}+1}\} & \text{STEP} = \{2a, 2b, 2c\} \\ \mathbf{q}^{\text{BALL}} = [0 \dots 0 \Delta q_j^{\text{BALL}}] & \text{BALL} = \{\text{I, II, III, IV, V}\}, \end{cases} \quad (13)$$

with, in addition, Δq_j^I set to 0. In practice, the three trajectories of step 2 are identical for all five balls, except for the value of the last joint angle, which is shifted by a constant value at each new ball insertion. To guarantee smooth transitions between the trajectories, the Rzeppa joint is pivoted during step 2(d) with the linear interpolation indicated by the second equation in (13).

To summarize, the complete nominal trajectory for assembling an Rzeppa joint, is given by the following concatenation of trajectories:

$$\begin{aligned} \mathcal{T} &= \mathcal{T}_1 \cup \mathcal{T}_{2a}^{\text{II}} \cup \mathcal{T}_{2b}^{\text{II}} \cup \mathcal{T}_{2c}^{\text{II}} \cup \mathcal{T}_{2d}^{\text{II}} \cup \mathcal{T}_{2a}^{\text{III}} \cup \mathcal{T}_{2b}^{\text{III}} \cup \mathcal{T}_{2c}^{\text{III}} \cup \mathcal{T}_{2d}^{\text{III}} \cup \\ &\quad \cup \mathcal{T}_{2a}^{\text{IV}} \cup \mathcal{T}_{2b}^{\text{IV}} \cup \mathcal{T}_{2c}^{\text{IV}} \cup \mathcal{T}_{2d}^{\text{IV}} \cup \mathcal{T}_{2a}^{\text{V}} \cup \mathcal{T}_{2b}^{\text{V}} \cup \mathcal{T}_{2c}^{\text{V}} \cup \mathcal{T}_{2d}^{\text{V}} \cup \\ &\quad \cup \mathcal{T}_{2a}^{\text{VI}} \cup \mathcal{T}_{2b}^{\text{VI}} \cup \mathcal{T}_{2c}^{\text{VI}} \cup \mathcal{T}_{2d}^{\text{VI}} \cup \mathcal{T}_3. \end{aligned} \quad (14)$$

In the next section, we explain how this nominal trajectory \mathcal{T} is used to control the robot, during the Rzeppa joint assembly operation, when the interaction forces with the environment and operator are also to be taken into account.

6. Admittance control

Admittance and impedance are reciprocal concepts [30]: while impedance control produces forces/torques in response to velocities, admittance control produces velocities in response to forces and torques, at a given interaction port. Here, we apply an admittance

controller, to map the contact wrench to a desired deviation of the end effector motion in the base frame, ${}^B\Delta\mathbf{p}_E$, through an equivalent spring-damper system. This guarantees motion safety, since the end effector trajectory is 'naturally' deformed to avoid blockage. It also makes the robot compliant during step 1, so that the operator can manually guide the inner race in the insertion tool.

The contact between robot and environment is located at S , and the frame \mathcal{F}_S is immobile during operation. For these reasons, the admittance controller will rely on the wrench applied on the end effector and expressed in the support frame, ${}^S\mathbf{h}_E$. It is straightforward to obtain this from ${}^E\mathbf{h}_E$, through:

$${}^S\mathbf{h}_E = {}^S\mathbf{W}_E {}^E\mathbf{h}_E, \quad (15)$$

with:

$${}^S\mathbf{W}_E = \begin{bmatrix} {}^S\mathbf{R}_E & \mathbf{0} \\ [{}^S\mathbf{t}_E]_{\times} & {}^S\mathbf{R}_E \end{bmatrix}, \quad (16)$$

and ${}^S\mathbf{R}_E$ the rotation matrix corresponding to angle/axis vector ${}^S\theta\mathbf{u}_E$. The relative pose of the effector in the support frame required in (16), can be derived at each iteration from the robot end effector pose ${}^B\mathbf{p}_E(\mathbf{q})$, and from the pose of the support in the base frame, ${}^B\mathbf{p}_S$ (computed as explained in 5).

To ensure that the admittance controller is activated only for sufficiently high values, and filter out measurement noise, each component of ${}^S\mathbf{h}_E$ is passed through the following deadband filter:

$$h_{adm,i} = \begin{cases} {}^S h_{E,i} + h_{m,i} & \text{if } {}^S h_{E,i} < -h_{m,i} \\ 0 & \text{if } |{}^S h_{E,i}| < h_{m,i} \\ {}^S h_{E,i} - h_{m,i} & \text{if } {}^S h_{E,i} > h_{m,i} \quad i = 1 \dots 6, \end{cases} \quad (17)$$

to generate the components of wrench vector \mathbf{h}_{adm} , that is the one actually input to the admittance controller. The sizes of the six deadband regions, $h_{m,i}$, are manually tuned, as will be explained just below.

We consider a spring-damper system, with null mass, and adjustable parameters (stiffness matrix \mathbf{K} , and damping matrix \mathbf{B} , both positive definite and diagonal):

$$\mathbf{h}_{adm} = \mathbf{K}^S \Delta\mathbf{p}_E + \mathbf{B}^S \dot{\Delta}\mathbf{p}_E. \quad (18)$$

Numerically differentiating on a sampling period Δt , and rearranging terms, yields the expression of ${}^S\Delta\mathbf{p}_E$ at time t :

$${}^S\Delta\mathbf{p}_E(t) = \Delta t (\Delta t \mathbf{K} + \mathbf{B})^{-1} \mathbf{h}_{adm}(t) + (\Delta t \mathbf{K} + \mathbf{B})^{-1} \mathbf{B}^S \Delta\mathbf{p}_E(t - \Delta t). \quad (19)$$

This controller relies on its output at the previous iteration ${}^S\Delta\mathbf{p}_E(t - \Delta t)$. Finally, a spatial motion transform matrix is used to convert ${}^S\Delta\mathbf{p}_E$ to ${}^B\Delta\mathbf{p}_E$, as required by the inverse kinematics controller (2).

The whole admittance control pipeline is outlined in Fig. 7. The procedure for tuning all the parameters ($h_{m,i}$, and the diagonal components of \mathbf{K} and \mathbf{B} , noted respectively $k_{1 \dots 6}$ and $b_{1 \dots 6}$) is the following.

- (1) The robot is manipulated away from the insertion tool support, with the end effector not in contact with the environment.
- (2) The damping is set to zero in (19), so the ${}^B\Delta\mathbf{p}_E$, to be realized by the robot is a linear function of \mathbf{h}_{adm} (pure spring model).
- (3) The operator applies external torques/forces to the end effector along each of the six dof, and tunes the corresponding values of the deadband range $h_{m,1 \dots 6}$ and of the stiffness $k_{1 \dots 6}$, to obtain the desired deadband tolerance and static gain.
- (4) \mathbf{B} is introduced to remove the oscillations: the operator applies an external wrench, but this time the robot is controlled

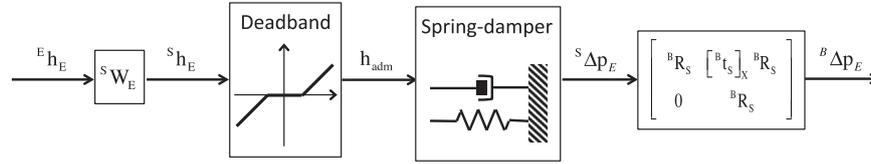


Fig. 7. Complete pipeline of the admittance controller, from measured external wrench ${}^E\mathbf{h}_E$ to desired pose variation ${}^B\Delta\mathbf{p}_E$.

with (19), and the $b_{1\dots 6}$ are tuned to obtain the desired response time for each component.

The values used in the experiments are given in Section 8.

7. Safety

This section presents the *safety* module. This relies on two tests. The first consists in checking whether any component of ${}^S\mathbf{h}_E$ is greater than the corresponding one in a threshold vector. Then, according to the current phase of the experiment, the robot either stops ('safety stop') or changes waypoint, using a strategy that will be detailed in Section 7.1. Second, the external camera is used to monitor safety of the insertion. If the operator hand is detected in the scene during robot motion, a *safety stop* is triggered. Details on the algorithm used for this are given in Section 7.2.

7.1. Dealing with strong external wrench

The external wrench on the end effector is used not only as input to the admittance controller presented just above, but also to monitor state changes or blocking situations during the assembly operation. Here also, we use its expression in the support frame, ${}^S\mathbf{h}_E$, derived with (15). The components of ${}^S\mathbf{h}_E$ are monitored either to check if a step has been completed, or to stop the motion in case of blockage. These two cases are indicated respectively by the left and right arrows exiting block *Test on external wrench* in Fig. 4.

In practice, steps 1 and 2(b) are considered completed if any component of ${}^S\mathbf{h}_E$ is greater than the corresponding one in threshold vectors τ_1 and τ_{2b} . If this is the case, even if the end effector has not yet reached the end point (respectively, P and C), the nominal reference is shifted to the start of the next trajectory (respectively, \mathcal{T}_{2a} and \mathcal{T}_{2c}). During the connecting trajectory \mathcal{T}_1 , we only monitor component ${}^S\mathbf{f}_{E,Z}$:

$$\tau_1 = [+ \infty \quad + \infty \quad \tau_{1,z} \quad + \infty \quad + \infty \quad + \infty]. \quad (20)$$

Since this force is due to the first contact between the Rzeppa joint and the insertion tool, if it passes the threshold, we consider point P to be reached, and start the first inclination step, \mathcal{T}_{2a}^1 . During the cage closing phase, \mathcal{T}_{2b} , a strong component of ${}^S\mathbf{h}_E$ will indicate that the closure is complete, and that the ball has been properly inserted in the inner race. Then, we consider that point C has been reached, and reverse the motion to return towards P , by following \mathcal{T}_{2c} .

Throughout operation, a safety stop is triggered if any component of ${}^S\mathbf{h}_E$ is greater than the corresponding one in threshold vector τ_s . The robot is blocked to avoid damaging the parts (or the robot itself), and the operator must manually unblock it, to resume operation. The image processing module (explained just below) ensures that motion is not resumed while the operator hand is in contact with the end effector.

7.2. Image processing

To safely interact with the human and with the environment, the designed controller relies on various sensed data. Along with the force/torque measures that are monitored as explained in the

previous section, we utilise a camera to observe the scene around the insertion tool. Specifically, the camera should detect the operator hand, to trigger a safety stop, if the hand is near the robot, and restart motion, once a ball has been inserted.

The camera is placed with its optical axis parallel to the insertion tool, to maximize scene saliency. The type of image seen from the camera is shown in the bottom left of Fig. 3. Only the image greyscale is considered, and we assume that the operator hand is the lighter gray area in the scene. To facilitate detection, the operator wears white gloves, and the ground floor, below the end effector, is black. These are reasonable constraints, since the industrial cell can be partly structured, and since image processing is not the main contribution of this work. More sophisticated approaches, such as [42,43], could be applied to relax these constraints.

A simple image processing algorithm is used to derive the distance d_h between the hand and the insertion tool axis (see bottom right of Fig. 3). For this, two geometric parameters are predefined according to the relative pose of the camera and tool. These are: the locus of the tool axis (point A), and an annulus surrounding the tool, which is considered relevant for safety. Only the pixels in this annulus are taken into account (the grey pixels in the figure are excluded, and in the following paragraph, we refer to this annulus as the *image*).

The algorithm then consists in the following steps:

1. A two-dimensional Gaussian filter is applied to the image to improve the signal-to-noise ratio.
2. The image is binarized: pixels are turned black or white, by comparing their luminosity with a fixed threshold.
3. One erosion and two dilations are applied to the image, to connect regions of similar pixels (blobs).
4. The size of each white blob in the image is evaluated by counting the neighbour pixels.
5. The greatest white blob is selected; if it is sufficiently large, it is assumed to be the hand, and d_h is calculated. Otherwise, d_h is undefined.

If d_h is defined, and if it is below a minimum tolerated distance for safety, denoted d_m , a safety stop is triggered (see Fig. 4). The stop is only active while $d_h < d_m$. For example, if the operator has to manually unblock the robot as explained in Section 7.1, the robot will resume motion only when: it has been unblocked (all components of ${}^S\mathbf{h}_E$ are smaller than those of τ_s) and the operator hand is in safety (not in the camera field of view, or with $d_h > d_m$). This method is also used to close the Rzeppa cage: as soon as the hand exits the dangerous area, the ball is considered inserted, and the robot starts the closing phase (step 2(b)).

8. Experiments

8.1. Setup and nominal trajectory generation

To validate our framework, we have run a series of experiments with a 7 dof lightweight KUKA LWR IV³ robot in the scenario

³ www.kuka-labs.com/en

illustrated in Fig. 3 (top). This robot is well known for its flexibility, which has fostered its use in pHRI applications [44]. The controller sampling time is set to $\Delta t = 20$ ms (a specification imposed by the ICARO project software architecture). The \mathbf{q}^* computed by the controller (see Fig. 4) are smoothed by the Reflexxes online trajectory generation library⁴ before being sent to the robot joints. To get the interaction wrench ${}^E\mathbf{h}_E$, instead of mounting a force sensor on the end effector, we average, over a window of 4 samples, the wrench that is estimated by the robot controller through the FRI Interface⁵ every 5 ms. This signal, which is derived from the applied and measured joint torques, proved to be accurate enough for our application. The camera used to monitor the human hand is a B & W Stingray F201B from Allied Vision Technologies, with resolution 1024×768 pixels. As secondary task in (2), we impose joint limit avoidance, via the following scalar cost function:

$$g(\mathbf{q}) = \frac{1}{2} \sum_{k=1}^7 \left(\frac{q_k - q_{k,\text{mid}}}{q_{k,M} - q_{k,m}} \right)^2, \quad (21)$$

with $[q_{k,m}, q_{k,M}]$ the available range for joint k and $q_{k,\text{mid}} = (q_{k,M} + q_{k,m})/2$ its midpoint.

Two compliant controllers (in the operational and joint space) are already embedded in the KUKA LWR. However, we decided to use our own admittance controller (described in Section 6) for three reasons. First, we must control the robot in the joint space (to verify its configuration w.r.t. the environment and w.r.t. self-collisions), whilst defining the admittance characteristics (wrench references, damping and stiffness) in the operational space, and this is not feasible with the KUKA controller. Second, the admittance parameters must be varied online at the various assembly steps, and the KUKA controller can only operate with prefixed values. Finally, our method can be applied to any position-based industrial robot with wrench measurement, making the framework generic, and not limited to the LWR.

The manually taught trajectory is identified (see Fig. 6, left), as an arc of circle of radius $R=0.241$ m, center position ${}^B\mathbf{t}_S = [-0.432 \ -0.421 \ 0.448]^T$ m, and amplitude $\psi_M - \psi_m = 1.70$ radians. Then, to generate the corresponding articular joint trajectories, we apply (2) to track operational space trajectory (10), on a KUKA LWR IV, simulated in Webots⁶ (see Fig. 6, right). The simulations are used to record the joint values (11), without endangering the real robot. We set $\Lambda = \mathbf{I}_6$, $\epsilon_t = 0.02$ and $\epsilon_r = 0.005$, to obtain the five primitive joint trajectories (12) for each of the seven joints, plotted in Fig. 8.

8.2. Control parameters

These trajectories are deformed online in case of external contacts, using the admittance controller described in Section 6. For the deadband filter, we apply the following threshold: $\mathbf{h}_m = [8 \ 8 \ 8 \ 3 \ 3 \ 3]^T$, (i.e., force components below 8 N and torque components below 3 Nm are not taken into account). The stiffness and damping matrix components are set to:

$$\begin{aligned} k_{1,2} &= 250 \text{ N/mm} & k_3 &= 500 \text{ N/mm} & k_{4,5,6} &= 170 \text{ Nm/mrad} \\ b_{1,2} &= 0.005 \text{ Ns/m} & b_3 &= 0.01 \text{ Ns/m} & b_{4,5,6} &= 0.003 \text{ Nms/rad}. \end{aligned} \quad (22)$$

The threshold vectors that are used to check if a step has been completed, or to stop the motion (see Section 7.1), are as follows (forces expressed in N, moments in Nm).

- To verify that P has been reached (i.e., that the tool has been

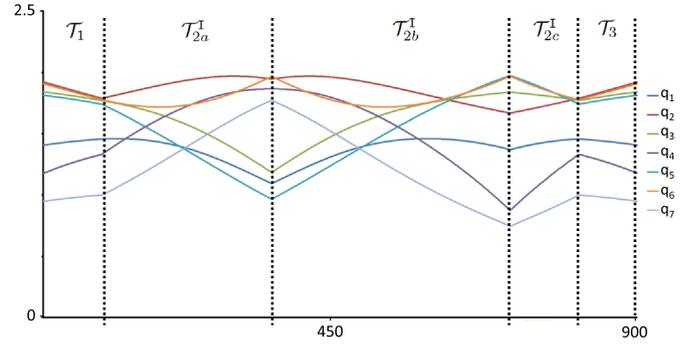


Fig. 8. The five primitive trajectories in joint space, obtained by inverse kinematics in Webots (abscissa: iterations, ordinate: joint angles in rad).

inserted), we only monitor ${}^S\mathbf{f}_{E,Z}$, according to (20): $\tau_{1,z} = 40$ N.

- To verify that the closure is complete, we compare the wrench to the threshold:

$$\tau_{2b} = [70 \ 40 \ 70 \ 15 \ 15 \ 15]^T. \quad (23)$$

The reason for setting the threshold on ${}^S\mathbf{f}_{E,Y}$ smaller than ${}^S\mathbf{f}_{E,X}$ and ${}^S\mathbf{f}_{E,Z}$ is that, in our setup, when the closure is completed, the most relevant forces appear along the support frame Y -axis (see Fig. 3).

- To block the robot:

$$\tau_s = [70 \ 70 \ 70 \ 25 \ 25 \ 25]^T. \quad (24)$$

In this case, all directions are considered equally relevant.

Although we used the ViSP library [45] for visualization, the image processing algorithms for detecting and tracking the operator hand were developed from scratch, as explained in Section 7.2. The threshold for considering the hand dangerous is set to $d_m=300$ pixels, while the relevant annulus minimum and maximum radii are respectively 180 and 750 pixels. The whole image processing pipeline takes only 5 ms, well below both the controller (20 ms) and camera framegrabber (33 ms) sampling times.

8.3. Results and discussion

The experiments consisted in having various users insert the five balls in the Rzeppa joint. We hereby discuss the results of these experiments, which are shown in Fig. 9, and in the video available at <https://youtu.be/3KWduKKSyy8>.

On the top of Fig. 9, the relevant steps of a complete successful experiment are shown. These include: tool connection/disconnection (1, 8), ball insertion (2, 3, 7), joint pivoting (4, 5), vision triggered emergency stop (6), and joint manual unmounting (9, 10).

The joint trajectories obtained during this experiment are plotted in Fig. 10. In this figure, the emergency stops activated by vision are highlighted in yellow, and the insertion phases for each ball are indicated with the black rectangles. Correspondingly, the robot motion is stopped. As the curves show, the values of $q_{1..6}$ are the same for each of the five balls, whereas q_7 changes, to pivot the cage before the start of the next ball insertion (just before the vertical dashed lines). In the current version of the software, the end of the insertion phase is notified by keyboard pressing, to resume robot motion. The reason is that it is very difficult, and unnecessary, to determine this state automatically. In future work, this notification could be given via a pedal, to free the operator hands.

The admittance controller is indispensable to avoid blockage between robot and tool, which systematically occurs when it is deactivated. Its effect can be seen in Fig. 11, where we have plotted the components of the external wrench ${}^S\mathbf{h}_E$, along with the

⁴ www.reflexxes.com

⁵ <http://cs.stanford.edu/people/tkr/fri/html/>

⁶ www.cyberbotics.com

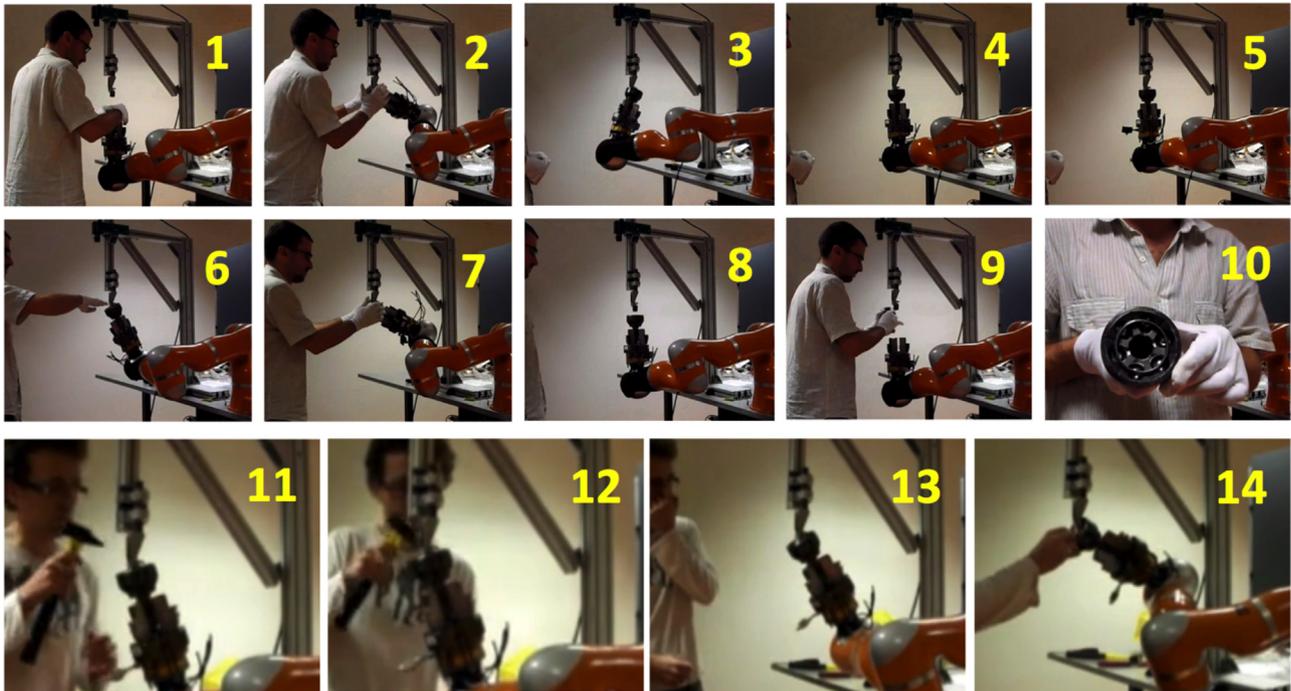


Fig. 9. Top: first experiment. (1) Tool connection. (2, 3) Second ball insertion. (4, 5) Joint pivoting. (6) Vision triggered emergency stop. (7) Sixth ball insertion. (8) Tool disconnection. (9, 10) Unmounting the assembled Rzeppa joint. Bottom: second experiment. (11, 12). The operator unblocks the joint using a hammer. (13, 14) Motion resumes and a new ball can be inserted.

corresponding variation of the articular trajectories. Apart from the peaks on ${}^S\mathbf{f}_{E,Y}$ (red curves in the top figure), which occur at each of the five insertion phases, it can be seen that throughout the assembly, the admittance controller induces variations of the order of magnitude of 0.02 rad (1 degree) to the articular joint values.

Although the whole assembly takes approximately 120 s, i.e., almost 4 times more than the manual one, the following aspects must be considered.

- Part of the delays are due to the current cell configuration. In particular, when the Rzeppa joint opens, the cage is horizontal. This complicates the ball insertion task, and makes, in this experiment, the operator drop the second ball to the ground, causing a 25 s insertion phase (first black rectangle in Fig. 10). To solve this problem, in future work, we plan to incline the insertion tool w.r.t. the ground, so that the housing is horizontal during ball insertion.

- We have deliberately decided to operate at 40% of the maximum speed allowed in physical Human–Robot Interaction (0.25 ms^{-1}). In fact, throughout the experiments, the tool center point velocity is always below 0.1 ms^{-1} . However, given the low image processing sampling time (5 ms), we are confident that increasing the robot speed towards the 0.25 ms^{-1} limit would not pose problems.
- Note that, even if the collaborative assembly was executed at the maximum speed allowed in physical Human–Robot Interaction (0.25 ms^{-1}), it would still be slower than the manual one. However, since the operator activity over one cycle time is smaller, it could be possible to have more robots working with a single operator, in order to leverage his/her production time.
- Another gain can be obtained by pivoting the Rzeppa joint during step 2(c). This would suppress step 2(d), which currently takes approximately 10% of the entire cycle time (2.5 s per ball).

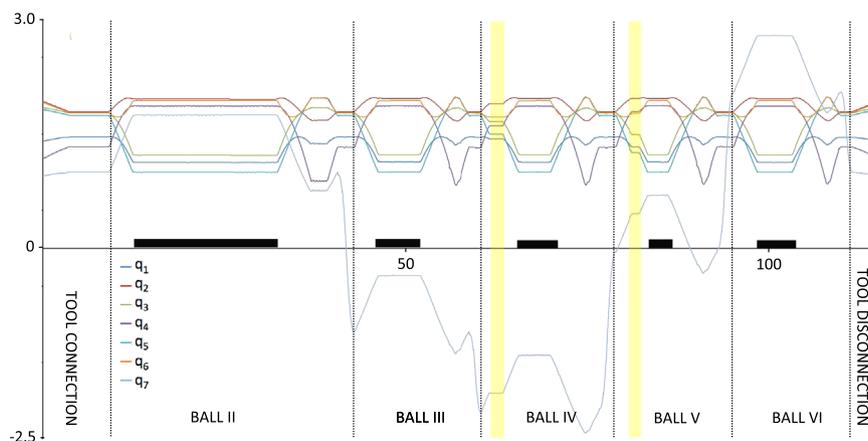


Fig. 10. The joint trajectories, during the first experiment (abscissa: time in s, ordinate: joint angles in rad). The emergency stops activated by vision are highlighted in yellow, and the insertion phases for each ball are indicated with the black rectangles. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

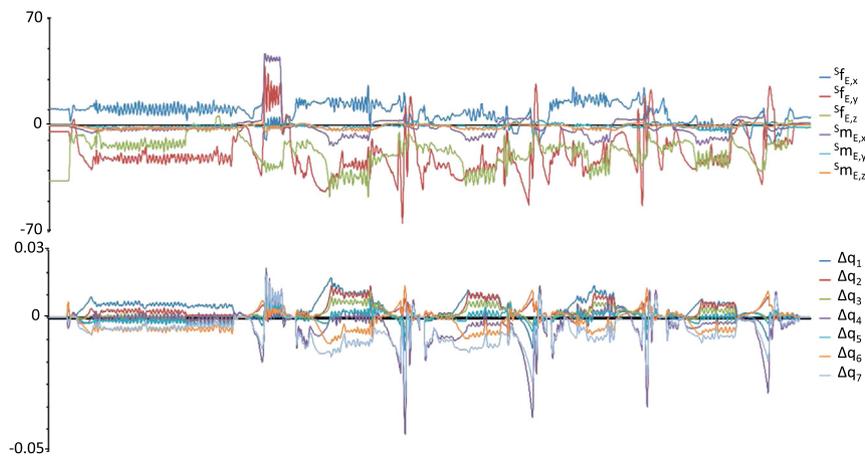


Fig. 11. Six components of the external wrench ${}^S\mathbf{h}_E$ (top, in N and Nm), and corresponding variation of the joint trajectories, $\Delta\mathbf{q}$, produced by the admittance controller (bottom, in rad). The abscissa timescale is identical to that of Fig. 10. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

- The collaborative robot assembly substantially reduces the risk of strain injuries with respect to the manual assembly, since most of the physical effort is realized by the robot. In fact, the operator load is reduced by approximately 60%, and, following this work, the Rzeppa assembly cell was reclassified in the PSA ergonomics scale (from 'red' to 'medium' level).

The experiments have been repeated by two users, to show the simplicity and robustness of the system. The video shows four experiments, where the entire assembly is completed, twice by each user. The last experiment (shown in the bottom of Fig. 9) is of particular interest, since it shows a case where the robot motion is blocked due to excessive external wrench (above threshold (24)). In this case, the user has to manually unblock the cage, using a hammer, until motion resumes (when the wrench returns within the safety range). It is noteworthy to point out that this also often occurs during manual assembly by experienced users. Typically, a slight misplacement of the third ball can jam the inner race and housing, requiring intervention with a hammer. The advantage of our framework is that these situations are automatically managed at a high level, avoiding permanent blockage of the robot.

9. Conclusions

This paper presents preliminary results on the development of a collaborative human–robot manufacturing cell for Rzeppa joint assembling. In contrast with most human–machine manufacturing applications, this one requires direct physical contact between robot and human, as well as between robot and environment.

The proposed framework integrates many state-of-the-art robotics components, seldom applied in real industrial scenarios. These include: trajectory optimization, admittance control, and image processing. The approach is validated in a series of experiments, with different users. With the proposed visual gesture monitoring and intrinsic collision detection, the robot setup is compatible with the safety standards, and could be certified. In addition, PSA has conducted a complete risk analysis of the application, to ensure that there is no obstacle in the deployment of such technology. Although the cycle time is lower than that of manual assembly, the collaborative cell lightens the operator burden, and substantially reduces the risk of strain injuries. Furthermore, the causes of the assembly delays have been identified, and we are confident that slight adjustments in this sense would radically reduce the cycle time, and spur the transfer of the proposed technology.

If successfully deployed in automotive factories, the proposed robotic assistant would have a very quick return on investment. Indeed, in spite of the high price of the hardware, the expected reduction of MSDs and associated costs, is tremendous. It has been estimated, by PSA, that the savings in terms of activity limitations and disability could cover the hardware cost of the setup in approximately a year.

Following this work, the two groups (LIRMM and PSA) are pursuing research in this field, in the context of other French National Projects. Future work will consist in deploying the proposed methodologies on mobile manipulator robots, which represent key technologies in flexible manufacturing.

Acknowledgements

This work has been supported by the ANR (French National Agency) ICARO project, grant number ANR-10-CORD-0025.

Appendix A. Supplementary material

Supplementary data associated with this article can be found in the online version at <http://dx.doi.org/10.1016/j.rcim.2015.12.007>.

References

- [1] P. Akella, M. Peshkin, E. Colgate, W. Wannasupphasit, N. Nagesh, J. Wells, S. Holland, T. Pearson, B. Peacock, Cobots for the automobile assembly line, in: IEEE International Conference on Robotics and Automation, ICRA, 1999.
- [2] J. Kruger, T.K. Lien, A. Verl, Cooperation of human and machines in assembly lines, *CIRP Ann. – Manuf. Technol.* 58 (2009) 628–646.
- [3] G. Michalos, S. Makris, N. Papakostas, D. Mourtzis, G. Chryssolouris, Automotive assembly technologies review: challenges and outlook for a flexible and adaptive approach, *CIRP J. Manuf. Sci. Technol.* 2 (2) (2010) 81–91.
- [4] G. Michalos, S. Makris, J. Spiliotopoulos, I. Misios, P. Tsarouchi, G. Chryssolouris, ROBO-PARTNER: Seamless Human–Robot cooperation for intelligent, flexible and safe operations in the assembly factories of the future, in: The Fifth CIRP Conference on Assembly Technologies and Systems (CATS), 2014.
- [5] J. Shi, G. Jimmerson, T. Pearson, R. Menassa, Levels of human and robot collaboration for automotive manufacturing, in: Workshop on Performance Metrics for Intelligent Systems, PerMIS, 2012, pp. 95–100.
- [6] W. Knight, How Human–Robot teamwork will upend manufacturing - MIT Technology Review, (www.technologyreview.com).
- [7] New human–robot cooperation in Audi's production processes - Feb 16, 2015, (www.audi.com).
- [8] N. Papakostas, G. Michalos, S. Makris, D. Zouzas, G. Chryssolouris, Industrial applications with cooperating robots for the flexible assembly, *Int. J. Comput. Integr. Manuf.* 24 (7) (2011) 650–660.
- [9] J. Tan, D. Feng, Y. Zhang, K. Watanabe, R. Kato, T. Arai, Human–robot

- collaboration in cellular manufacturing: Design and development, in: IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 2009.
- [10] M. Morioka, S. Sakakibara, A new cell production assembly system with human–robot cooperation, *CIRP J. Manuf. Sci. Technol.* 59 (2010) 9–12.
- [11] N. Pedrocchi, F. Vicentini, M. Malosio, L.M. Tosatti, Safe human–robot cooperation in an industrial environment, *Int. J. Adv. Robot. Syst., IJARS* 10 (27) (2012).
- [12] H. Ding, M. Schipper, M. Bjoern, Optimized task distribution for industrial assembly in mixed human–robot environments – case study on IO module assembly, in: IEEE International Conference on Automation Science and Engineering, CASE, 2014.
- [13] A. Bicchi, M. Peshkin, J. Colgate, Safety for physical human–robot interaction, in: B. Siciliano, O. Khatib (Eds.), *Springer Handbook of Robotics*, Springer, 2008, pp. 1335–1348.
- [14] A. De Luca, F. Flacco, Integrated control for phri: collision avoidance, detection, reaction and collaboration, in: IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics, BIOROB, 2012.
- [15] C. Heyer, Human–robot interaction and future industrial robotics applications, in: IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 2010, pp. 4749–4754.
- [16] A. De Santis, B. Siciliano, A. De Luca, A. Bicchi, An atlas of physical human–robot interaction, *Mech. Mach. Theory* 43 (3) (2007) 253–270.
- [17] S. Haddadin, A. Albu-Schäffer, G. Hirzinger, Requirements for safe robots: measurements, analysis and new insights, *Int. J. Robot. Res., IJRR*, 2009.
- [18] F. Flacco, A. De Luca, Safe physical human–robot collaboration, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2013.
- [19] J. Fryman, M. Bjoern, Safety of industrial robots: From conventional to collaborative applications, in: The Seventh German Conference on Robotics ROBOTIK, 2012.
- [20] L. Roveda, F. Vicentini, L.M. Tosatti, Deformation-tracking impedance control in interaction with uncertain environments, in: IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 2013.
- [21] G. Ganesh, A. Albu-Schäffer, M. Haruno, M. Kawato, E. Burdet, Biomimetic motor behavior for simultaneous adaptation of force, impedance and trajectory in interaction tasks, in: IEEE International Conference on Robotics and Automation, ICRA, 2010.
- [22] C. Yang, G. Ganesh, S. Haddadin, S. Parusel, A. Albu-Schäffer, E. Burdet, Human-like adaptation of force and impedance in stable and unstable interactions, *IEEE Trans. Robot.* (2011). <http://dx.doi.org/10.1109/TRO.2011.2158251>.
- [23] D. Surdilovic, G. Schreck, U. Schmidt, Development of collaborative robots (cobots) for flexible human-integrated assembly automation, in: 41st International Symposium on Robotics, ISR, 2010.
- [24] F. Lange, W. Bertleff, M. Suppa, Force and trajectory control of industrial robots in stiff contact, in: IEEE International Conference on Robotics and Automation, ICRA, 2013.
- [25] M. Suphi Edren, T. Tomiyama, Human intent detection and physically interactive control of a robot without force sensors, *IEEE Trans. Robot.* 26 (2) (2010) 370–382.
- [26] M. Suphi Edren, B. Maric, Assisting manual welding with robot, *Robot. Comput. Integr. Manuf.* 27 (2011) 818–828.
- [27] M. Suphi Edren, A. Billard, End-point impedance measurements at human hand during interactive manual welding with robot, in: IEEE International Conference on Robotics and Automation, ICRA, 2014.
- [28] G. Ferretti, G. Magnani, P. Rocco, Assigning virtual tool dynamics to an industrial robot through an admittance controller, in: International Conference on Advanced Robotics, ICAR, 2009.
- [29] F. Ficuciello, A. Romano, L. Villani, B. Siciliano, Cartesian impedance control of redundant manipulators for human–robot co-manipulation, in: IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 2013.
- [30] N. Hogan, Impedance control: an approach to manipulation: parts I–III, *ASME J. Dyn. Syst. Meas. Control* 107 (1985) 1–24.
- [31] G. Grunwald, A. Albu-Schäffer, G. Hirzinger, Touch: the direct type of human interaction with a redundant service robot, in: IEEE International Workshop on Robot and Human Interactive Communication, ROMAN, 2001.
- [32] C. Perez Quintero, R. Tatsambon Fomena, A. Shademan, N. Wolleb, T. Dick, M. Jagersand, Sepo: Selecting by pointing as an intuitive human–robot command interface, in: IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 2013.
- [33] V. Lippiello, B. Siciliano, L. Villani, Interaction control of robot manipulators using force and vision, *Int. J. Optomechatron.* 2 (3) (2008) 257–274.
- [34] A. De Santis, V. Lippiello, B. Siciliano, L. Villani, Human–robot interaction control using force and vision, *Adv. Control Theory Appl.* 353 (2007) 51–70.
- [35] A. Cherubini, R. Passama, A. Meline, A. Crosnier, P. Fraisse, Multimodal control for human–robot cooperation, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2013.
- [36] A. Lasnier, ANR ICARO D2.1 - Description détaillée des besoins, PSA, ANR ICARO Project Confidential Deliverable (in French), 2012.
- [37] K. Waldron, J. Schmiedeler, Kinematics, in: B. Siciliano, O. Khatib (Eds.), *Springer Handbook of Robotics*, Springer, 2008, pp. 9–33.
- [38] B. Siciliano, L. Sciacivco, L. Villani, G. Oriolo, *Robotics: Modelling, Planning and Control*, Springer, 2009.
- [39] A. Liegeois, Automatic supervisory control of configurations and behavior of multibody mechanisms, *IEEE Trans. Syst. Man Cybern.* 7 (6) (1977) 868–871.
- [40] N. Mansard, O. Khatib, A. Kheddar, A unified approach to integrate unilateral constraints in the stack of tasks, *IEEE Trans. Robot.* 25 (3) (2009).
- [41] Robotiq, Collaborative robot e-book – (<http://blog.robotiq.com>).
- [42] I. Oikonomidis, N. Kyriazis, A.A. Argyros, Full dof tracking of a hand interacting with an object by modeling occlusions and physical constraints, in: International Conference on Computer Vision ICCV, 2011.
- [43] S.S. Rautaray, A. Agrawal, A real time hand tracking system for interactive applications, *Int. J. Comput. Appl.* 18 (March (6)) (2011) 28–33.
- [44] R. Bischoff, J. Kurth, G. Schreiber, R. Koeppel, A. Albu-Schaeffer, A. Beyer, O. Eiberger, S. Haddadin, A. Stemmer, G. Grunwald, G. Hirzinger, The KUKA-DLR Lightweight Robot Arm – A new reference platform for robotics research and manufacturing, in: the 41st International Symposium on Robotics (ISR) and the Sixth German Conference on Robotics (ROBOTIK), 2010.
- [45] E. Marchand, F. Spindler, F. Chaumette, Visp for visual servoing: a generic software platform with a wide class of robot control skills, *IEEE Robot. Autom. Mag., Special Issue Softw. Packages Vision-Based Control Motion* 12 (4) (2005) 40–52.

Bibliography

- [1] A. Bicchi, M. Peshkin, and J. Colgate, “Safety for physical human-robot interaction,” *Springer Handbook of Robotics*, B. Siciliano and O. Khatib (Eds.), Springer, pp. 1335-1348, 2008.
- [2] A. De Luca and F. Flacco, “Integrated control for pHRI: collision avoidance, detection, reaction and collaboration,” in *IEEE RAS/EMBS Int. Conf. on Biomedical Robotics and Biomechatronics, BIOROB*, 2012.
- [3] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi, “An atlas of physical human-robot interaction,” *Mechanism and Machine Theory*, vol. 43, no. 3, pp. 253–270, 2007.
- [4] “ISO 13482:2014 Robots and robotic devices - Safety requirements for personal care robots,” International Organization for Standardization, Geneva, Switzerland, Tech. Rep., 2014.
- [5] Y. Maeda, T. Hara, and T. Arai, “Human-robot cooperative manipulation with motion estimation,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, vol. 4, 2001, pp. 2240–2245.
- [6] E. Gribovskaya, A. Kheddar, and A. Billard, “Motion learning and adaptive impedance for robot control during physical interaction with humans,” in *IEEE Int. Conf. on Robotics and Automation*, 2011, pp. 4326–4332.
- [7] A. Bussy, A. Kheddar, A. Crosnier, and F. Keith, “Human-humanoid haptic joint object transportation case study,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2012, pp. 3633–3638.
- [8] G. Grunwald, A. Albu-Schäffer, and G. Hirzinger, “Touch: the direct type of human interaction with a redundant service robot,” in *IEEE Int. Symposium on Robot and Human Interactive Communication*, 2001.
- [9] M. Suphi Erden and T. Tomiyama, “Human intent detection and physically interactive control of a robot without force sensors,” *IEEE Trans. on Robotics*, vol. 26, no. 2, pp. 370–382, 2010.
- [10] S. M. La Valle, *Planning Algorithms*. Cambridge University Press, 2006.

- [11] F. Chaumette and S. Hutchinson, “Visual servo control, Part I: Basic approaches,” *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006.
- [12] M. H. Raibert and J. J. Craig, “Hybrid position/force control of manipulators,” *ASME J. Dyn. Syst. Meas. Control*, no. 103, pp. 126–133, 1981.
- [13] U. Proske and S. C. Gandevia, “The proprioceptive senses: their roles in signaling body shape, body position and movement, and muscle force,” *Physiol Rev.*, vol. 92, no. 4, pp. 1651–1697, 2012.
- [14] A. De Luca and R. Mattone, “Sensorless robot collision detection and hybrid force/motion control.” in *IEEE Int. Conf. on Robotics and Automation*, 2005, pp. 999–1004.
- [15] N. Hogan, “Impedance control: an approach to manipulation: parts I-III,” *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 107, pp. 1–24, 1985.
- [16] A. Cherubini and F. Chaumette, “Visual navigation of a mobile robot with laser-based collision avoidance.” *Int. Journal of Robotics Research*, vol. 32, no. 2, pp. 189–209, 2013.
- [17] A. Cherubini, F. Spindler, and F. Chaumette, “Autonomous visual navigation and laser-based moving obstacle avoidance,” *IEEE Trans. on Int. Transportation Systems*, vol. 15, no. 5, pp. 2101–2110, 2014.
- [18] A. Cherubini, M. Colafrancesco, G. Oriolo, L. Freda, and F. Chaumette, “Comparing appearance-based controllers for nonholonomic navigation from a visual memory,” in *IEEE ICRA 2009 Workshop on safe navigation in open and dynamic environments: application to autonomous vehicles*, 2009.
- [19] A. Cherubini and F. Chaumette, “Visual navigation with a time-independent varying reference,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2009.
- [20] ———, “A redundancy-based approach for obstacle avoidance in mobile robot navigation,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2010.
- [21] A. Cherubini, F. Spindler, and F. Chaumette, “A redundancy-based approach for visual navigation with collision avoidance,” in *IEEE Symp. on Computational Intelligence in Vehicles and Transportation Systems, CIVTS*, 2011.
- [22] A. Cherubini and F. Chaumette, “Visual navigation with obstacle avoidance,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2011.

- [23] A. Cherubini, F. Spindler, and F. Chaumette, “A new tentacles-based technique for avoiding obstacles during visual navigation,” in *IEEE Int. Conf. on Robotics and Automation*, May 2012.
- [24] A. Cherubini, B. Grechanichenko, F. Spindler, and F. Chaumette, “Avoiding moving obstacles during visual navigation,” in *IEEE Int. Conf. on Robotics and Automation*, May 2013.
- [25] C. Sprunk, G. D. Tipaldi, A. Cherubini, and W. Burgard, “Lidar-based teach-and-repeat of mobile robot trajectories,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2013.
- [26] A. Cherubini, R. Passama, P. Fraitse, and A. Crosnier, “A unified multimodal control framework for human-robot interaction,” *Robotics and Autonomous Systems*, vol. 70, pp. 106–115, 2015.
- [27] A. Cherubini, R. Passama, A. Crosnier, A. Lasnier, and P. Fraitse, “Collaborative manufacturing with physical human-robot interaction,” *Robotics and Computer-Integrated Manufacturing*, vol. 40, pp. 1–13, August 2016.
- [28] A. Cherubini, R. Passama, A. Meline, and P. Fraitse, “Multimodal control for human-robot cooperation,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, nov JTCF Novel Technology Paper Award for Amusement culture finalist, 2013.
- [29] B. Navarro, A. Cherubini, A. Fonte, R. Passama, G. Poisson, and P. Fraitse, “An ISO10218-compliant adaptive damping controller for safe physical human-robot interaction,” in *IEEE Int. Conf. on Robotics and Automation*, May 2016.
- [30] D. J. Agravante, A. Cherubini, A. Bussy, and A. Kheddar, “Human-humanoid joint haptic table carrying task with height stabilization using vision,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2013.
- [31] D. J. Agravante, A. Cherubini, and A. Kheddar, “Using vision and haptic sensing for human-humanoid haptic joint actions,” in *6th IEEE International Conference on Robotics, Automation and Mechatronics, RAM*, 2013.
- [32] D. J. Agravante, A. Cherubini, A. Bussy, P. Gergondet, and A. Kheddar, “Collaborative human-humanoid carrying using vision and haptic sensing,” in *IEEE Int. Conf. on Robotics and Automation*, 2014.
- [33] A. Paolillo, A. Cherubini, F. Keith, A. Kheddar, and M. Vendittelli, “Toward autonomous car driving by a humanoid robot: A sensor-based framework,” in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2014.

- [34] D. J. Agravante, A. Sherikov, P.-B. Wieber, A. Cherubini, and A. Kheddar, "Walking pattern generators designed for physical collaboration," in *IEEE Int. Conf. on Robotics and Automation*, May 2016.
- [35] D. J. Agravante, A. Cherubini, A. Sherikov, P.-B. Wieber, and A. Kheddar, "Human-humanoid collaborative carrying," *Int. Journal of Robotics Research (submitted)*, 2016.
- [36] P. Gergondet, D. Petit, M. Meilland, A. Kheddar, A. I. Comport, and A. Cherubini, "Combining 3D slam and visual tracking to reach and retrieve objects in daily-life indoor environments," in *11th Int. Conf. on Ubiquitous Robots and Ambient Intelligence, URAI*, 2014.
- [37] D. Petit, P. Gergondet, A. Cherubini, M. Meilland, A. I. Comport, and A. Kheddar, "Navigation assistance for a BCI-controlled humanoid robot," in *IEEE 4th Annual Int. Conf. on Cyber Technology in Automation, Control, and Intelligent Systems, CYBER*, 2014.
- [38] D. Petit, P. Gergondet, A. Cherubini, and A. Kheddar, "An integrated framework for humanoid embodiment with a BCI," in *IEEE Int. Conf. on Robotics and Automation*, May 2015.
- [39] W. Tigra, B. Navarro, A. Cherubini, X. Gorron, A. Gelis, C. Fattal, D. Guiraud, and C. Azevedo Coste, "A novel EMG interface for individuals with quadriplegia to pilot robot hand grasping," *IEEE Transactions on Neural Systems and Rehabilitation (to appear)*, 2016.
- [40] A. Paolillo, P. Gergondet, A. Cherubini, V. M., and A. Kheddar, "Autonomous car driving by a humanoid robot," *Journal of Field Robotics (under revision)*, 2016.
- [41] L. Villani and J. De Schutter, "Force Control," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2008, ch. 7, pp. 161–185.
- [42] T. B. Kazerooni H., Sheridan and P. K. Houpt, "Robust compliant motion for manipulators. Part I: the fundamental concepts of compliant motion," *IEEE J. Robot. Autom.*, no. 2, pp. 83–92, 1986.
- [43] B. J. Nelson, J. D. Morrow, and P. K. Khosla, "Improved force control through visual servoing," in *Proc. of the American Control Conference*, vol. 1. IEEE, 1995, pp. 380–386.
- [44] K. Hosoda, K. Igarashi, and M. Asada, "Adaptive hybrid visual servoing/force control in unknown environment," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, vol. 3, 1996, pp. 1097–1103 vol.3.

- [45] R. Smits, T. De Laet, K. Claes, H. Bruyninckx, and J. De Schutter, “iTASC: a tool for multi-sensor integration in robot manipulation,” in *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2008, pp. 426–433.
- [46] G. Taylor and L. Kleeman, *Visual Perception and Robotic Manipulation: 3D Object Recognition, Tracking and Hand-Eye Coordination*, ser. Springer Tracts in Advanced Robotics. Springer, 2006.
- [47] S. Wieland, D. Gonzalez-Aguirre, N. Vahrenkamp, T. Asfour, and R. Dillmann, “Combining force and visual feedback for physical interaction tasks in humanoid robots,” in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2009, pp. 439–446.
- [48] M. O. Ernst and M. S. Banks, “Humans integrate visual and haptic information in a statistically optimal fashion,” *Nature*, vol. 415, no. 6870, pp. 429–433, 2002.
- [49] M. O. Ernst and H. H. Bühlhoff, “Merging the senses into a robust percept,” *Trends in cognitive sciences*, vol. 8, no. 4, pp. 162–169, 2004.
- [50] B. J. Nelson and P. K. Khosla, “Force and vision resolvability for assimilating disparate sensory feedback,” *IEEE Trans. on Robotics and Automation*, vol. 12, no. 5, pp. 714–731, 1996.
- [51] R. Featherstone, *Rigid body dynamics algorithms*. Springer, 2008.
- [52] M. T. Mason, “Compliance and force control for computer controlled manipulators,” *IEEE Trans. on Systems, Man and Cybernetics*, vol. 11, no. 6, pp. 418–432, 1981.
- [53] J. Baeten, H. Bruyninckx, and J. De Schutter, “Integrated vision/force robotic servoing in the task frame formalism,” *Int. Journal of Robotics Research*, vol. 22, no. 10-11, pp. 941–954, 2003.
- [54] M. Buehler and S. E. Lagnemma, K. and Singh, “Special issue on the 2007 DARPA Urban Challenge, Part I-III,” *Journal of Field Robotics*, vol. 25, no. 8–10, pp. 423–860, 2008.
- [55] Y. Matsumoto, M. Inaba, and H. Inoue, “Visual navigation using view-sequenced route representation,” in *IEEE Int. Conf. on Robotics and Automation*, 1996.
- [56] E. Royer, M. Lhuillier, M. Dhome, and J.-M. Lavest, “Monocular vision for mobile robot localization and autonomous navigation,” *Int. Journal of Computer Vision*, vol. 74, no. 3, pp. 237–260, 2007.

- [57] A. A. Argyros, K. E. Bekris, S. C. Orphanoudakis, and L. E. Kavraki, "Robot homing by exploiting panoramic vision," *Autonomous Robots*, vol. 19, no. 1, pp. 7–25, 2005.
- [58] A. Kelly, B. Nagy, D. Stager, and R. Unnikrishnan, "An infrastructure-free automated guided vehicle based on computer vision," *IEEE Robotics and Automation Magazine*, 2007.
- [59] H. Mäkelä and T. von Numers, "Development of a navigation and control system for an autonomous outdoor vehicle in a steel plant," *Control Engineering Practice*, vol. 9, no. 5, pp. 573–583, 2001.
- [60] M. Vitus and C. Tomlin, "Sensor placement for improved robotic navigation," in *Robotics: Science and Systems (RSS)*, 2010.
- [61] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. V. Gool, "Omnidirectional vision based topological navigation," *Int. Journal of Computer Vision*, vol. 74, no. 3, pp. 219–236, 2007.
- [62] A. Diosi, S. Segvic, A. Remazeilles, and F. Chaumette, "Experimental evaluation of autonomous driving based on visual memory and image based visual servoing," *IEEE Trans. on Int. Transportation Systems*, vol. 12, no. 3, pp. 870–883, 2011.
- [63] G. L. Mariottini, G. Oriolo, and D. Prattichizzo, "Image-based visual servoing for nonholonomic mobile robots using epipolar geometry," *IEEE Trans. on Robotics*, vol. 23, no. 1, pp. 87–100, 2007.
- [64] P. Zanne, G. Morel, and F. Plestan, "Sensor-based robot control in the presence of uncertainties: bounding the task function tracking errors," in *IEEE Int. Conf. on Robotics and Automation*, 2002.
- [65] F. Mezouar, Y. and Chaumette, "Path planning for robust image-based control," *IEEE Trans. on Robotics and Automation*, vol. 18, no. 4, pp. 534–549, 2002.
- [66] J. Pomares and F. Torres, "Time independent tracking using 2-d movement flow based visual servoing," in *IEEE Int. Conf. on Robotics and Automation*, 2005.
- [67] L. Lapierre, R. Zapata, and P. Lepinay, "Simultaneous path following and obstacle avoidance control of a unicycle-type robot," in *IEEE Int. Conf. on Robotics and Automation*, 2007.
- [68] T.-S. Lee, G.-H. Eoh, and B.-H. Kim, J.and Lee, "Mobile robot navigation with reactive free space estimation," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2010.

- [69] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *IEEE Int. Conf. on Robotics and Automation*, 1985.
- [70] F. Von Hundelshausen, M. Himmelsbach, F. Hecker, A. Mueller, and H.-J. Wuensche, "Driving with tentacles-integral structures of sensing and motion," *Journal of Field Robotics*, vol. 25, no. 9, pp. 640–673, 2008.
- [71] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *Int. Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.
- [72] J. P. Wu, A. and How, "Guaranteed infinite horizon avoidance of unpredictable, dynamically constrained obstacles," *Autonomous Robots*, vol. 32, no. 3, pp. 227–242, 2012.
- [73] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to obstacle avoidance," *IEEE Robotics and Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.
- [74] J. Damas, B. and Santos-Victor, "Avoiding moving obstacles: the forbidden velocity map," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2009.
- [75] D. Folio and V. Cadenat, "A redundancy-based scheme to perform safe vision-based tasks amidst obstacles," in *IEEE Int. Conf. on Robotics and Biomimetics*, 2006.
- [76] P. Akella, M. Peshkin, E. Colgate, W. Wannasuphoprasit, N. Nagesh, J. Wells, S. Holland, T. Pearson, and B. Peacock, "Cobots for the automobile assembly line," in *IEEE Int. Conf. on Robotics and Automation*, 1999.
- [77] J. Kruger, T. K. Lien, and A. Verl, "Cooperation of human and machines in assembly lines," *CIRP Annals - Manufacturing Technology*, vol. 58, pp. 628–646, 2009.
- [78] J. Shi, G. Jimmerson, T. Pearson, and R. Menassa, "Levels of human and robot collaboration for automotive manufacturing," in *Workshop on Performance Metrics for Intelligent Systems, PerMIS*, 2012, pp. 95–100.
- [79] W. Knight, "How Human-Robot teamwork will upend manufacturing - MIT Technology Review - www.technologyreview.com."
- [80] "New human-robot cooperation in Audi's production processes" - Feb 16, 2015 - www.audi.com.
- [81] C. Heyer, "Human-robot interaction and future industrial robotics applications," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2010, pp. 4749–4754.

- [82] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, “Requirements for safe robots: Measurements, analysis and new insights,” *Int. Journal of Robotics Research*, 2009.
- [83] J. Fryman and M. Bjoern, “Safety of industrial robots: From conventional to collaborative applications,” in *7th German Conference on Robotics ROBOTIK*, 2012.
- [84] F. Flacco and A. De Luca, “Safe physical human-robot collaboration,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2013.
- [85] “ISO 10218-1:2011 Robot for industrial environments - Safety requirements - Part 1 : Robot,” International Organization for Standardization, Geneva, Switzerland, Tech. Rep., 2006.
- [86] F. Flacco, T. Kroger, A. De Luca, and O. Khatib, “A Depth Space Approach to Human-Robot Collision Avoidance,” in *IEEE Int. Conf. on Robotics and Automation*, 2012.
- [87] D. Song, N. Kyriazis, I. Oikonomidis, C. Papazov, A. Argyros, D. Burschka, and D. Kragic, “Predicting human intention in visual observations of hand/object interactions,” in *IEEE Int. Conf. on Robotics and Automation*, 2013, pp. 1608–1615.
- [88] L. Roveda, F. Vicentini, and L. M. Tosatti, “Deformation-tracking impedance control in interaction with uncertain environments,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2013.
- [89] G. Ganesh, A. Albu-Schäffer, M. Haruno, M. Kawato, and E. Burdet, “Biomimetic motor behavior for simultaneous adaptation of force impedance and trajectory in interaction tasks,” in *IEEE Int. Conf. on Robotics and Automation*, 2010.
- [90] C. Yang, G. Ganesh, S. Haddadin, S. Parusel, A. Albu-Schäffer, and E. Burdet, “Human-like adaptation of force and impedance in stable and unstable interactions,” *IEEE Trans. on Robotics*, 2011.
- [91] D. Surdilovic, G. Schreck, and U. Schmidt, “Development of collaborative robots (cobots) for flexible human-integrated assembly automation,” in *41st International Symposium on Robotics, ISR*, 2010.
- [92] F. Lange, W. Bertleff, and M. Suppa, “Force and trajectory control of industrial robots in stiff contact,” in *IEEE Int. Conf. on Robotics and Automation*, 2013.
- [93] M. Suphi Erden and B. Maric, “Assisting manual welding with robot,” *Robotics and Computer Integrated Manufacturing*, vol. 27, pp. 818–828, 2011.

- [94] M. Suphi Erden and A. Billard, “End-point impedance measurements at human hand during interactive manual welding with robot,” in *IEEE Int. Conf. on Robotics and Automation*, 2014.
- [95] G. Ferretti, G. Magnani, and P. Rocco, “Assigning virtual tool dynamics to an industrial robot through an admittance controller,” in *Int. Conf. on Advanced Robotics, ICAR*, 2009.
- [96] F. Ficuciello, A. Romano, L. Villani, and B. Siciliano, “Cartesian impedance control of redundant manipulators for human-robot co-manipulation,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2013.
- [97] Robotiq, “Collaborative robot ebook - 3rd edition - <http://blog.robotiq.com>.”
- [98] M. Zinn, O. Khatib, B. Roth, and J. Salisbury, “Playing it safe,” *IEEE Robotics and Automation Magazine*, vol. 11, no. 2, pp. 12–21, 2004.
- [99] A. Albu-Schaffer et al., “Soft robotics,” *IEEE Robotics and Automation Magazine*, vol. 15, no. 3, pp. 20–30, 2008.
- [100] R. Schiavi, G. Grioli, S. Sen, and A. Bicchi, “VSA-II: A novel prototype of variable stiffness actuator for safe and performing robots interacting with humans,” in *IEEE Int. Conf. on Robotics and Automation*, 2008.
- [101] A. Vick, D. Surdilovic, and J. Krüger, “Safe physical human-robot interaction with industrial dual-arm robots,” in *9th IEEE Workshop on Robot Motion and Control (RoMoCo)*, 2013, pp. 264–269.
- [102] S. Jung, T. Hsia, and R. Bonitz, “Force tracking impedance control of robot manipulators under unknown environment,” *IEEE Trans. on Control Systems Technology*, vol. 12, no. 3, pp. 474–483, May 2004.
- [103] F. Almeida, A. Lopes, and P. Abreu, “Force-impedance control: A new control strategy of robotic manipulators,” in *Recent Advances in Mechatronics, Springer, Singapore*, 1999, pp. 126–137.
- [104] V. Duchaine and C. Gosselin, “Safe, stable and intuitive control for physical human-robot interaction,” in *IEEE Int. Conf. on Robotics and Automation, ICRA*, 2009, pp. 3383–3388.
- [105] H. Sadeghian, L. Villani, M. Keshmiri, and B. Siciliano, “Experimental study on task space control during physical human robot interaction,” in *2nd RSI/ISM Int. Conf. on Robotics and Mechatronics (ICRoM)*, Oct 2014, pp. 125–130.
- [106] R. Ikeura and H. Inooka, “Variable impedance control of a robot for cooperation with a human,” in *IEEE Int. Conf. on Robotics and Automation*, vol. 3, 1995, pp. 3097–3102 vol.3.

- [107] R. Ikeura, T. Moriguchi, and K. Mizutani, "Optimal variable impedance control for a robot and its application to lifting an object with a human," in *IEEE Int. Symposium on Robot and Human Interactive Communication*, 2002, pp. 500–505.
- [108] E. Burdet, R. Osu, D. W. Franklin, T. E. Milner, and M. Kawato, "The central nervous system stabilizes unstable dynamics by learning optimal impedance," *Nature*, vol. 414, no. 6862, pp. 446–449, 2001.
- [109] V. Duchaine and C. M. Gosselin, "General model of human-robot cooperation using a novel velocity based variable impedance control," in *EuroHaptics Conf. and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*. IEEE, 2007, pp. 446–451.
- [110] B. Corteville, E. Aertbeliën, H. Bruyninckx, J. De Schutter, and H. Van Brussel, "Human-inspired robot assistant for fast point-to-point movements," in *IEEE Int. Conf. on Robotics and Automation*, 2007, pp. 3639–3644.
- [111] A. De Santis, V. Lippiello, B. Siciliano, and L. Villani, "Human-robot interaction control using force and vision," *Advances in Control Theory and Applications*, vol. 353, pp. 51–70, 2007.
- [112] J. A. Corrales, G. J. García Gómez, F. Torres, and V. Perdereau, "Cooperative tasks between humans and robots in industrial environments," *Int. Journal of Advanced Robotic Systems, IJARS*, 2012.
- [113] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer, 2009.
- [114] N. Mansard, O. Stasse, P. Evrard, and A. Kheddar, "A versatile generalized inverted kinematics implementation for collaborative working humanoid robots: The stack of tasks," in *Int. Conf. on Advanced Robotics, ICAR*, 2009.
- [115] M. Prats, P. J. Sanz, and A. P. Del Pobil, "Reliable non-prehensile door opening through the combination of vision, tactile and force feedback," *Autonomous Robots*, vol. 29, no. 2, pp. 201–218, 2010.
- [116] V. Mariano Gonçalves, P. Fraise, A. Crosnier, and B. V. Adorno, "Parsimonious Kinematic Control of Highly Redundant Robots," *IEEE Robotics and Automation Letters*, no. 1, pp. 65–72, 2016.
- [117] H. Hasunuma, K. Nakashima, M. Kobayashi, F. Mifune, Y. Yanagihara, T. Ueno, K. Ohya, and K. Yokoi, "A tele-operated humanoid robot drives a backhoe," in *IEEE Int. Conf. on Robotics and Automation*, 2003, pp. 2998–3004.

- [118] K. Yokoi, K. Nakashima, M. Kobayashi, H. Mihune, H. Hasunuma, Y. Yanagihara, T. Ueno, T. Gokyu, and K. Endou, “A tele-operated humanoid operator,” *Int. Journal of Robotics Research*, vol. 5-6, pp. 593–602, 2006.
- [119] K. Kosuge, M. Sato, and N. Kazamura, “Mobile robot helper,” in *IEEE Int. Conf. on Robotics and Automation*, vol. 1, 2000, pp. 583–588 vol.1.
- [120] M. Lawitzky, A. Mortl, and S. Hirche, “Load sharing in human-robot cooperative manipulation,” in *IEEE Int. Symposium on Robot and Human Interactive Communication*, Sept 2010, pp. 185–191.
- [121] J. Stückler and S. Behnke, “Following human guidance to cooperatively carry a large object,” in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2011, pp. 218–223.
- [122] G. Morel, E. Malis, and S. Boudet, “Impedance based combination of visual and force control,” in *IEEE Int. Conf. on Robotics and Automation*, vol. 2, 1998, pp. 1743–1748.
- [123] M. Prats, P. Martinet, A. P. del Pobil, and S. Lee, “Vision force control in task-oriented grasping and manipulation,” in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2007, pp. 1320–1325.
- [124] V. Perdereau and M. Drouin, “A new scheme for hybrid force-position control,” *Robotica*, vol. 11, pp. 453–464, 1993.
- [125] E. Malis, G. Morel, and F. Chaumette, “Robot control using disparate multiple sensors,” *Int. Journal of Robotics Research*, vol. 20, no. 5, pp. 364–377, 2001.
- [126] E. Berger, D. Vogt, N. Haji-Ghassemi, B. Jung, and H. B. Amor, “Inferring guidance information in cooperative human-robot tasks,” in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2013.
- [127] M. Bellaccini, L. Lanari, A. Paolillo, and M. Vendittelli, “Manual guidance of humanoid robots without force sensors: Preliminary experiments with NAO,” in *IEEE Int. Conf. on Robotics and Automation*, May 2014, pp. 1184–1189.
- [128] S. McGill and D. Lee, “Cooperative humanoid stretcher manipulation and locomotion,” in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2011, pp. 429–433.
- [129] T. Koolen, T. de Boer, J. Rebula, A. Goswami, and J. Pratt, “Capturability-based analysis and control of legged locomotion, Part 1: Theory and application to three simple gait models,” *Int. Journal of Robotics Research*, vol. 31, no. 9, pp. 1094–1113, 2012.

- [130] J. Pratt, T. Koolen, T. de Boer, J. Rebula, S. Cotton, J. Carff, M. Johnson, and P. Neuhaus, "Capturability-based analysis and control of legged locomotion, Part 2: Application to M2V2, a lower-body humanoid," *Int. Journal of Robotics Research*, vol. 31, no. 10, pp. 1117–1133, 2012.
- [131] K. Yokoyama, H. Handa, T. Isozumi, Y. Fukase, K. Kaneko, F. Kanehiro, Y. Kawai, F. Tomita, and H. Hirukawa, "Cooperative works by a human and a humanoid robot," in *IEEE Int. Conf. on Robotics and Automation*, vol. 3, 2003, pp. 2985–2991.
- [132] A. Bussy, P. Gergondet, A. Kheddar, F. Keith, and A. Crosnier, "Proactive behavior of a humanoid robot in a haptic transportation task with a human partner," in *IEEE Int. Symposium on Robot and Human Interactive Communication*, 2012, pp. 962–967.
- [133] W. W. Newman, Z.-H. Chong, C. Du, R. Hung, K.-H. Lee, L. Ma, T. Ng, C. Swetenham, K. Tjoeng, and W. Wang, "Autonomous valve turning with an atlas humanoid robot," in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2009, pp. 439–446.
- [134] A. Herdt, H. Diedam, P.-B. Wieber, D. Dimitrov, K. Mombaur, and M. Diehl, "Online walking motion generation with automatic footstep placement," *Advanced Robotics*, vol. 24, no. 5-6, pp. 719–737, 2010.
- [135] C. Billian and P. H. Gorman, "Upper extremity applications of functional neuromuscular stimulation," *Assistive Technology*, vol. 4, no. 1, pp. 31–39, 1992.
- [136] J. Weisz, C. Elvezio, and P. K. Allen, "A user interface for assistive grasping," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2013.
- [137] K. Farry, I. Walker, and R. Baraniuk, "Myoelectric teleoperation of a complex robotic hand," *IEEE Trans. on Robotics and Automation*, vol. 12, no. 5, pp. 775–788, Oct 1996.
- [138] C. Cipriani, F. Zacccone, S. Micera, and M. Carrozza, "On the shared control of an EMG-controlled prosthetic hand: Analysis of user," *IEEE Trans. on Robotics*, vol. 24, no. 1, pp. 170–184, Feb 2008.
- [139] M. Cutkosky, "On grasp choice, grasp models, and the design of hands for manufacturing tasks," *IEEE Trans. on Robotics and Automation*, vol. 5, no. 3, pp. 269–279, Jun 1989.
- [140] L. R. Hochberg, D. Bacher, B. Jarosiewicz, N. Y. Masse, J. D. Simeral, J. Vogel, S. Haddadin, J. Liu, S. S. Cash, P. Van der Smagt, and J. P. Donoghue, "Reach and grasp by people with tetraplegia using a neurally controlled robotic arm," *Nature*, vol. 485, no. 7398, pp. 372–375, 2012.

- [141] C. W. Moss, K. L. Kilgore, and P. H. Peckham, "A novel command signal for motor neuroprosthetic control," *Neurorehabilitation and neural repair*, vol. 25, no. 9, pp. 847–854, 2011.
- [142] A. S. Royer and B. He, "Goal selection versus process control in a brain-computer interface based on sensorimotor rhythms." *Journal of Neural Engineering*, vol. 6, no. 1, p. 016005, Mar. 2009.
- [143] J. del R. Millán, R. Rupp, G. R. Müller-Putz, R. Murray-Smith, C. Giugliemma, M. Tangermann, C. Vidaurre, F. Cincotti, A. Kübler, R. Leeb, C. Neuper, K.-R. Müller, and D. Mattia, "Combining Brain-Computer Interfaces and Assistive Technologies: State-of-the-Art and Challenges." *Frontiers in Neuroscience*, vol. 4, pp. 1–33, Jan. 2010.
- [144] F. Galán, M. Nuttin, E. Lew, P. W. Ferrez, G. Vanacker, J. Philips, and J. del R. Millán, "A Brain-Actuated Wheelchair: Asynchronous and Non-Invasive Brain-Computer Interfaces for Continuous Control of Robots," *Clinical Neurophysiology*, vol. 119, no. 2007, pp. 2159–2169, 2008.
- [145] C. J. Bell, P. Shenoy, R. Chalodhorn, and R. P. N. Rao, "Control of a humanoid robot by a noninvasive brain-computer interface in humans," *Journal of Neural Engineering*, vol. 5, no. 2, pp. 214–20, Jun. 2008.
- [146] A. Cherubini, G. Oriolo, F. Macrì, F. Aloise, F. Cincotti, and D. Mattia, "A multimode navigation system for an assistive robotics project," *Autonomous Robots*, vol. 25, no. 4, pp. 383–404, 2008.
- [147] F. Cincotti, F. Aloise, F. Bufalari, G. Schalk, G. Oriolo, A. Cherubini, F. Davide, F. Babiloni, M. Marciiani, and D. Mattia, "Non invasive brain-computer interface system to operate assistive devices," in *Engineering in Medicine and Biology Society (EMBC), 2007 IEEE International Conference of the*, 2007.
- [148] F. Aloise, F. Cincotti, M. Marciiani, F. Babiloni, D. Morelli, S. Paolucci, G. Oriolo, A. Cherubini, F. Sciarra, F. Mangiola, A. Melpignano, F. Davide, and D. Mattia, "The ASPICE project inclusive design for the motor disabled," in *Int. Working Conf. on Advanced Visual Interfaces*, 2006.
- [149] S. Kim, M. Kim, J. Lee, S. Hwang, J. Chae, B. Park, H. Cho, J. Sim, J. Jung, H. Lee, S. Shin, M. Kim, N. Kwak, Y. Lee, S. Lee, M. Lee, S. Yi, K.-S. K. Chang, and J. Park, "Approach of team SNU to the DARPA Robotics Challenge finals," in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2015, pp. 777–784.
- [150] S. McGill, S.-J. Yi, and D. D. Lee, "Team THOR's adaptive autonomy for disaster response humanoids," in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2015, pp. 453–460.

- [151] H. Jeong, J. Oh, M. Kim, K. Joo, I. S. Kweon, and J.-H. Oh, "Control strategies for a humanoid robot to drive and then egress a utility vehicle for remote approach," in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2015, pp. 811–816.
- [152] C. Rasmussen, K. Sohn, Q. Wang, and P. Oh, "Perception and control strategies for driving utility vehicles with a humanoid robot," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2014, pp. 973–980.
- [153] M. Meilland and A. I. Comport, "On unifying key-frame and voxel-based dense visual SLAM at large scales," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2013.
- [154] N. Mansard, O. Khatib, and A. Kheddar, "A unified approach to integrate unilateral constraints in the stack of tasks," *IEEE Trans. on Robotics*, vol. 25, no. 3, 2009.
- [155] F. Keith, P.-B. Wieber, and A. Mansard, N.and Kheddar, "Analysis of the discontinuities in prioritized tasks-space control under discrete task scheduling operations," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2011.
- [156] N. Mansard and F. Chaumette, "Task sequencing for high-level sensor-based control," *IEEE Trans. on Robotics*, vol. 23, no. 1, pp. 60 – 72, 2007.
- [157] M. Sorour, A. Cherubini, R. Passama, and P. Fraitse, "Kinematic modeling and singularity treatment of steerable wheeled mobile robots with joint acceleration limits," in *IEEE Int. Conf. on Robotics and Automation*, May 2016.
- [158] B. V. Adorno, P. Fraitse, and S. Druon, "Dual position control strategies using the cooperative dual task-space framework," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, 2010.
- [159] D. Navarro-Alarcon, Y.-H. Liu, J. G. Romero, and P. Li, "Model-free visually servoed deformation control of elastic objects by robot manipulators." *IEEE Trans. on Robotics*, vol. 29, no. 6, pp. 1457–1468, 2013.
- [160] M. Iosa, G. Morone, A. Cherubini, and S. Paolucci, "The three laws of neuro-robotics: a review on what neurorehabilitation robots should do for patients and clinicians," *Journal of Medical and Biological Engineering*, p. First Online, February 2016.
- [161] F. Cincotti, F. Aloise, F. Babiloni, M. G. Marciani, D. Morelli, S. Paolucci, G. Oriolo, A. Cherubini, S. Brusino, F. Sciarra, F. Mangiola, A. Melpignano, F. Davide, and D. Mattia, "Brain-operated assistive devices: the AS-PICE project," in *IEEE RAS/EMBS Int. Conf. on Biomedical Robotics and Biomechatronics, BIOROB*, 2006.

- [162] A. Cherubini, F. Giannone, L. Iocchi, M. Lombardo, and G. Oriolo, “Policy gradient learning for a humanoid soccer robot,” *Robotics and Autonomous Systems*, vol. 57, no. 8, pp. 808–818, 2009.