# Vision based motion generation for humanoid robots

Olivier Stasse

# Habilitation thesis

A dissertation submitted to the

**PAUL SABATIER UNIVERSITY**

presented by

**Olivier STASSE**

for the degree of

**Habilitation à diriger des recherches**

**in Robotics**

---

## Vision based motion generation for humanoid robots.

---

*4th April 2013*

---

**JURY**

| | | |
|---|---|---|
| M. F. Chaumette | Directeur de recherche à l'INRIA | Examiner |
| Mme. C. Chevallereau | Directeur de recherche au CNRS | Reviewer |
| M. A. Kheddar | Directeur de Recherche au CNRS | Examiner |
| M. F. Lamiraux | Directeur de Recherche au CNRS | Examiner |
| M. F. Lerasle | Professor | Examiner |
| M. J.-P. Laumond | Directeur de Recherche au CNRS | Examiner |
| M. P. Poignet | Professor | Reviewer |
| M. G. Sandini | Professor | Reviewer |

# Contents

# Acknowledgements

I would like to thank Christine Chevallerau, Philippe Poignet and Guilio Sandini for reviewing this manuscript, François Chaumette, Abderrahmane Kheddar, Florent Lamiraux, Jean-Paul Laumond, and Frédéric Lerasle for being members of my jury.

Thanks to Abderrahmane Kheddar for making JRL such a nice place for doing research on humanoid robotics. I learned a lot from him. Thanks also to Hirosa Hirukawa for his advices and friendship. Thanks to Philippe Coiffet to whom I own all the opportunities to go to Japan. Kazuhito Yokoi and Eiichi Yoshida helped and supported me through all those wonderful years in Japan as co-directors of JRL. A warm thank to them.

Special thanks to my main 3 closest collaborators: Florent Lamiraux, Nicolas Mansard, Pierre-Brice Wieber.

During my stay in Japan, I could collaborate with many other visiting researchers to be acknowledged: Pablo Alcantarilla, Andrew Davison, Claire Dune, Nosan Kwak, Diane Larlus, François Saïdi, Ramzi Sellouati, Benoit Telle, Bram Vanderborght, Bjorn Verrelst. Special thanks to my colleagues from HRG with their unvaluable advices on humanoid robotics and Japan life: Kenji Kaneko, Shuuji Kajita, Mitsu Morisawa, Fumio Kanehiro. Thanks to Tokiko Uga for her support during the difficulties with the Japanese administration.

The PhD students with whom I had the chance to work are very important actors of the work described in this manuscript. Thanks to Torea Foissotte, Nicolas Perrin, and Sovannara Hak.

Thanks to the people at JRL with whom I had the chance to work with: Léo Baudouin, François Bleibel, Hitoshi Arisumi, Sylvain Dupitier, Adrien Escande, Paul Evrard, Pierre Gergondet, François Keith, Baptiste Lagarde, Loic Magnan, Paolo Pierro, Neo Ee Sian, Jean Semère.

The people in France at LAAS in the Gepetto team, helped me a lot also during the process: Jean-Paul Laumond, Philippe Soueres, Michel Taix, Thomas Moulard, Antonio El Khoury.

I am probably missing some many important people who helped during my time at the Joint Robotics Laboratory. Thanks to all of them, it was a nice experience.

This manuscript is dedicated to my brother Arnaud and my mother Yolande. I would like to thank my wife Christelle for her love, patience and support. My father André and my children Léa, Maëlie et Kilyan brought me the needed joy and love to go through life.

# Abstract

This manuscript present my research activities on real-time vision-based behaviors for complex robots such as humanoids. The underlying main scientific question structuring this work is the following: "What are the decisional processes which make possible for a humanoid robot to generate motion in real-time based upon visual information ?" In soccer humans can decide to kick a ball while running and when all the other players are constantly moving. When recast as an optimization problem for a humanoid robot, finding a solution for such behavior is generally computationally hard. For instance, the problem of visual search consider in this work is NP-complete.

The first part of this work is concerned about real-time motion generation. Starting from the general constraints that a humanoid robot has to fulfill to generate a feasible motion, some core problems are presented. From this several contributions allowing a humanoid robot to react to change in the environment are presented. They revolve around walking pattern generation, whole body motion for obstacle avoidance, and real-time foot-step planning in constrained environment.

The second part of this work is concerned about real-time acquisition of knowledge on the environment through computer vision. Two main behaviors are considered: visual-search and visual object model construction. They are considered as a whole taking into account the model of the sensor, the motion cost, the mechanical constraints of the robot, the geometry of the environment as well as the limitation of the vision processes. In addition contributions on coupling Self Localization and Map Building with walking, real-time foot-steps generation based on visual servoing are presented.

Finally the core technologies developed in the previous contexts were used in different applications: Human-Robot interaction, tele-operation, human behavior analysis. Based upon the feedback of several integrated demonstrators on the humanoid robot HRP-2, the last part of this thesis tries to draw some directions where innovative ideas may break some current technical locks in humanoid robotics.

# Résumé

Ce manuscrit présente mes activités de recherche sur les comportements basés vision pour des robots complexes comme les robots humanoïdes. La question scientifique sous-jacente qui structure ce travail est la suivante: “ Quels sont les processus de décisions qui permettent à un robot humanoïde de générer des mouvements en temps réel basés sur des informations visuelles ?” Au football, les êtres humains peuvent décider de frapper une balle alors qu'ils courrent et que tous les autres joueurs sont constamment en train de bouger. Reformuler comme un problème d'optimisation pour un robot humanoïde, trouver une solution pour un tel comportement est généralement très difficile du point de vue calculatoire. Par exemple, le problème de la recherche visuelle a été demontré comme étant NP-complet.

La première partie de ce travail concerne la génération de mouvements temps réel. Partant des contraintes générales qu'un robot humanoïde doit remplir pour générer un mouvement faisable, des problèmes fondamentaux sont présentés. A partir de ceux-ci, plusieurs contributions permettant à un robot humanoïde de réagir à des changements de l'environnement sont présentés. Ils concernent la génération de la marche, les mouvements corps complets pour éviter des obstacles, et la planification de pas en temps réel dans des environnements contraints.

La deuxième partie de ce travail concerne l'acquisition temps-réel de connaissance sur l'environnement à partir de la vision par ordinateur. Deux comportements principaux sont considérés: la recherche visuelle et la construction d'un modèle visuel d'un object. Ils sont considérés tout en prenant compte le modéle du capteur, le coût du mouvement, les contraintes mécaniques du robot, la géometrie de l'environnement ainsi que les limitations du processus de vision. De plus des contributions sur le couplage de l'auto-localisation basé cartes avec la marche, la génération de pas basé sur l'asservissement visuel seront présentés.

Finalement les technologies centrales développées dans les contextes précédents ont été utilisées dans différentes applications: l'interaction homme-robot, la téléopération, l'analyse de mouvement humains. Basé sur le retour d'expérience de plusieurs démonstrateurs intégrés sur le robot humanoïde HRP-2, la dernière partie de cette thèse proposent des pistes pour des idées permettant de lever les verrous technologiques actuels de la robotique humanoïde.

# Chapter 1

# Introduction

## 1.1 Context

The context of this work lies in the field of service robotics with a focus on humanoid robots autonomously performing vision-based behaviors in a human like environment. There is a very large body of work in the field. A recent survey on humanoid robot is presented in the Phd Thesis of Buschmann [Bus10]. It includes a detailed presentation of the humanoid robot Lola (see [BSF+12] for brief overview). In the following a brief update of recent relevent advances in the field of bipedal humanoid robotics is given.

### 1.1.1 Honda

In Novembre 2011, Honda release the last version of Asimo. This new version can go up to 9 $km/h$, which is currently the fastest 3D humanoid robot. For walking generation this last version is still using the ZMP based scheme described later in this manuscript. The interaction between the walking pattern generator and the balancer is described in more details in a series of papers published in IROS 2009 [TMY09a, TMYS09, TMY09b, TMY+09c]. In this lastest release, Asimo is able to run, jump on one foot, and kick strongly in a ball. The robot is able to demonstrate very advanced high-level behavior like going to take drinks and bring them back to a person. Asimo is today the most advanced mechanical humanoid robot platform.

### 1.1.2 Boston Dynamics

In IROS 2011, Marc Raibert presented the PETMAN humanoid robot which has been designed to test chemical protective clothing. The PETMAN humanoid is different from the PETProto [NSN+12]. Most notably it has a a more human like shape. One of the drawback is the decrease in walking speed from $7.2km/h$ for the PETProto to 4.8 $km/h$ for PETMAN. PETPro has a leg structure with three links, the same

mechanical structure than BigDog. From [NSN+12], it is not clear if this is the reason for a slower speed, or if it comes from the control part. The balance controller in essence is using the capture point [PCDG] to change the foot placement of the swing leg. A hierarchical swing controller is in charge of planning the trajectory of the swing leg so that the touchdown position of the swing foot is the one specified by the balance controller. The hydraulic actuation is allowing a compliant interaction with the environment. PETMAN is the base for the humanoid robot ATLAS which will be provided in the context of the DARPA Robotics challenge.

### 1.1.3 DLR

The German Aerospace Research Center (DLR) has presented in 2011 [OEE+12] a bipedal robot with Joint Torque Sensors. This robot is using electric motors but including torque sensors directly at the joint level. Robots such as Asimo and the HRP series use harmonic drives which make very difficult to model the friction forces at the joint level. This new kind of robot might be the path to the implementation of torque control, and allow safe human-robot interaction. This robot can also be used using the classical high-gain control approach.

### 1.1.4 AIST

In 2011, AIST released a new humanoid robot platform called HRP-4 [KKM+11]. This new robot main characteristics are its light-weight (40 kg) and its slim body. Due to recent new laws in Japan, the motor power in HRP-4 is lower than the ones in HRP-2 to ensure safe human-robot interaction. The robot has also been designed to have a far lower cost than HRP-2. Nonetheless it is able to walk on unknown (but relatively flat) slopes [KMM+10], and is able to perform more human like walking [MMK+11].

### 1.1.5 KAIST

The Korean Advanced Institute of Science and Technology is a leading group in building humanoid robots with a strong emphasis on the mechanical capabilities. Despite strong similarities with HRP-2, the last version HUBO-II is very complete with grasping capabilities, and is among the few humanoid robots able to run and hop [CKO11].

### 1.1.6 IIT

In the field of humanoid robotics applied to biological models validation, the Cognitive Humanoids Lab at IIT is a leading research group in the field. Their famous

humanoid robot platform iCub [MSV+08] is used by several research groups in Europe to investigate how intelligence is developing when interacting with the real world. This robot is aiming at following the developmental path of a young human without necessary walking. For this reason, the mechanical structure is based on wires, and the recent developments integrate torque sensors at the joint level [FIR+12].

### 1.1.7 KIT

In the field of service robotics the humanoids and Intelligence Systems Lab of the Kalsruhe Institute of Technology (KIT) is widely famous for the humanoid robot ARMAR series [ARA+06]. The robots are intended to help humans in a human centered world. One of the most famous example is the robot ARMAR-III able to perform advance vision based behaviors in a kitchen [AAV+08]. The robot is able to learn new objects using vision through the interaction with a human [PHAS09], grasp and manipulate a wide variety of human objects [MAA+06], detect the state of a washing machine and put kitchen-ware inside. The level of integration achieved by KIT make this group among the world-wide most famous in this topic, and soon, with ARMAR IV, will also adress problem related to bipedal robotics.

### 1.1.8 Context of this manuscript: the Joint Robotics Laboratory

In 2003, Philippe Coiffet (CNRS) and Kazuo Tanie (AIST) created the Joint Robotics Laboratory with two sites: one in Tsukuba Japan, with the other one located in France. I joined the Japanese site which was supervised by Abderrahmane Kheddar and Kazuhito Yokoi. During my stay at the Joint French-Japanese Laboratory, most of my work took place in the frame of a long term research plan trying to address the "Treasure hunting" problem. It can be expressed very simply in the following manner: A person shows or indicates an object to a humanoid robot. The robot builds autonomously an internal representation by moving around the object. When this is realized, the object is put in a room for which the robot has no model. The robot has to find the object in a reasonnable amount of time. This kind of problem has received a renewed interest [MFL+08] with the Semantic Robot Vision Challenge organized at CVPR.

## 1.2 Problem

Behavior realization can be formalized as the following. Considering a robot with $n$ degrees of freedom $\mathbf{q}$, with a vector of information on its internal parameters and the environment $\mathbf{v} \in \mathbb{R}^m$. For a given behavior let us assume that it exists a function $f(\mathbf{q}, \mathbf{v}, t) : \mathbb{R}^{n \times m+1} \rightarrow [0, 1]$ such as it is equal to 0 when the behavior is realized. The

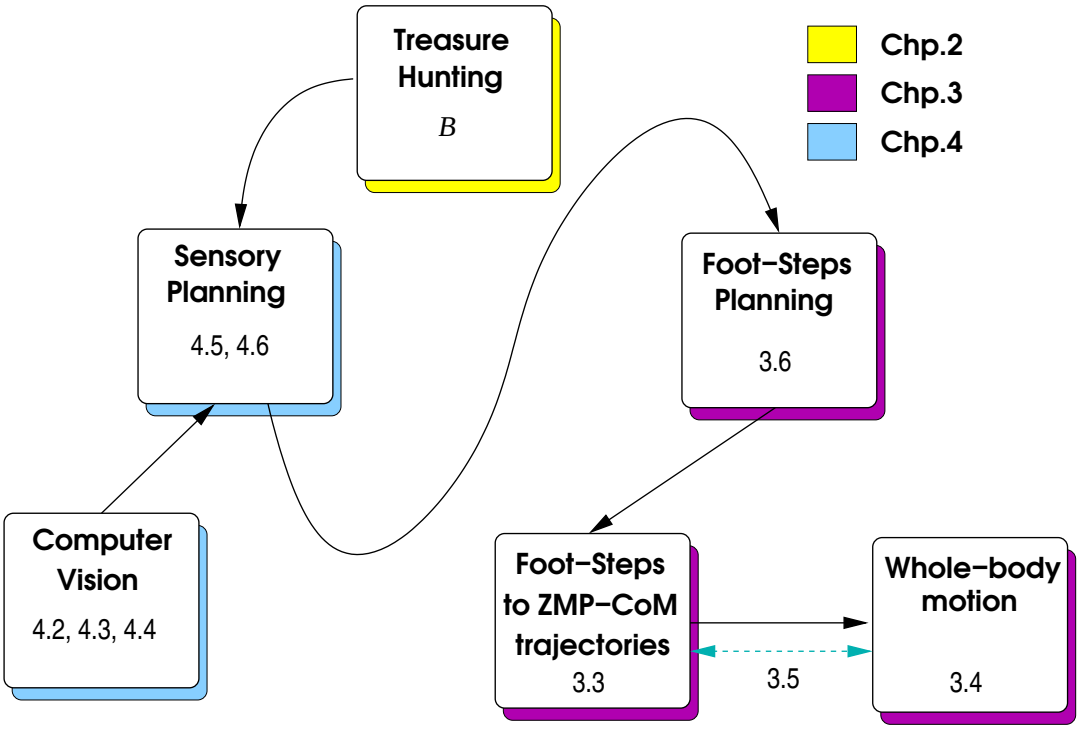



Figure 1.1: Multilayer approach in solving $B$. The number at the bottom of each box is the section number in this manuscript where explanations are provided

problem amount to find a trajectory $\mathbf{q}(t)$ such that

Behavior (B)

$$\min f(\mathbf{q}(t), \mathbf{v}(t)) \tag{1.1}$$
$$\mathbf{g}(\mathbf{q}(t), \mathbf{v}(t)) < 0 \tag{1.2}$$
$$\mathbf{h}(\mathbf{q}(t), \mathbf{v}(t)) = 0 \tag{1.3}$$

where $\mathbf{g}$ are unilateral constraints and $\mathbf{h}$ are bilateral constraints.

A common approach in robotics is to build a function $\hat{f}$ which is an approximation of $f$ based on a model of the current state of the environment $\mathbf{v}(t)$, and an estimation of the current state of the robot in this environment. A central issue is the formulation of $\hat{f}$ and $\mathbf{v}(t)$ to solve $B$ efficiently.

## 1.3 Overview

In practice we do not address $B$ directly but rather in a multilayer approach by formulating auxilaries optimization problems. Fig.1.1 depicts this decomposition and relates it to the organization of this manuscript. Generally, once a mission planner triggers the execution of the treasure hunting behavior, the system uses computer vision to

extract information from the environment. It might be 3d information regarding the object of the behavior (89) or the environment (10). From these informations, the system creates an instance of a sensory planning problem either for model building (11) or for visual search (12). When the best pose for the vision sensor is found, then a foot steps planner is used to find a feasible path (6.4). From this set of footsteps, reference trajectories for ZMP and CoM are computed to comply with the balance constraints (4). These reference trajectories are then realized by a whole-body motion controller (5).

In the context of the treasure hunting, we focus on two points: the acquisition of the object model in a known environment, and its search in an unknown environment. Both behaviors are formulated as an optimization problem similar to $B$. However the contributions differ greatly in the sense that for the visual search the work realized is mostly based on a greedy approach aiming at finding the global optimum in a reasonnable amount of time. The cost function is based upon information acquisition on unknown area, detection probability of the object, and motion cost. A video of this result is available [90]. This result has been selected as a finalist for the best paper award at ICAR 2007 [37]. For the reconstruction phase, the scheme developped is mostly finding local minima but aims to integrate more tightly the body constraints to the sensor planning problem. As those two problems are **NP**-complete [YT96] we have developped heuristics using humanoid robots specificities while considering a 6 dimensional search space.

In the course of addressing those two problems, it became apparent that whole-body motion and acyclic motion are very important in solving theoretically and practically $B$. From the theoritical viewpoint, current Bayesian approach [TBF05, MCdFB$^+$09], as well as the approach presented in this manuscript, uses model of the motion control subsystem. For instance, we used the limitation of the CoM height imposed by the Linearized Inverted Pendulum, to constraint the space of the camera in SLAM and in Visual Search. From a practical viewpoint, the necessity to realize and correct the pose of the sensor led us to develop a reactive whole-body motion generator where walking takes a great part. During the development of this motion generator, I collaborated with other researchers working on different topics but which did expand the feasible range of the robot motions. The most noticeable result is the stepping over behavior for which we received the best paper award at ICMA 2006 [48]. A video summing up the results on whole-body motion and its interaction with vision, namely SLAM and visual-servoing, was selected as a finalist for the best video award at ICRA 2007 [91]. In the context of the European Project ROBOT@CWE, the technology developped has been used to build an interactive application where the robot is moving a table cooperatively with a human.

The scientific part of this manuscript is organized in two main parts: motion generation and computer vision. In the first part the constraints related to humanoid robot motion generation are briefly stated. There are used to address the problem of finding a CoM trajectory given a set of foot-prints. From the CoM trajectory two methods can be used to find a compatible set of articular joint : generalized inverted

kinematics, or assuming a constant relationship between the waist and the CoM. For each of them, the derived applications are described. The second part deals with computer vision. At first a new camera model based on Intervals is proposed to deal with uncertainties. This result provides a guaranteed triangulation of points detected in images. Then a new model for object recognition is proposed in order to be able to detect far away candidates. After briefly state our contributions regarding the use of monocular SLAM together with walking, two high level behaviors are addressed: autonomous 3D object building, and visual search. Both are formulated as sensory planning problems submitted to humanoid robots contraints. Such constraints are the ones described in details inside the first part of the manuscript, and it is shown that they can be used to speed up the search into the space of the vision system pose. The most advanced experience realized regarding visual search has been choosen as a finalist for the best paper award at the ICAR conference in 2007 [37].

# Part I

# Motion Generation for humanoid robot

# Chapter 2

# Overview

This chapter deals with the problem of motion generation for a humanoid robot. It can be seen as a constraint satisfaction problem based on a representation of the environment. After briefly reviewing different approaches to tackle this problem, solutions allowing to take into account modifications of the environment are mostly considered. Indeed in the case of the treasure hunting problem the model of the environment can change according to the perceived visual information. More generally the robot should be able to correct its foot-steps according to sensory information. Examples such as human-humanoid robot interaction through force-feedback are given in this part (section 5.3).

This problem is addressed by a multi-layered solution: planning foot-steps, generating dynamically stable CoM trajectory from these foot-steps and whole-body control to realize the CoM trajectory. All the contributions described in this chapter revolve around the problem of generating dynamically stable CoM trajectory and its implication on the other layers.

## 2.1 CoM trajectory generation from foot-steps

This work takes its root from the preview-control proposed by Kajita [KKK+03a], which generates a CoM trajectory from a desired ZMP-trajectory. The latter one is defined by the point the most distant from all the constraints defined by the foot-steps. In [Wie06], Wieber proposed a reformulation which does not define heuristically the ZMP trajectory but simply constrains it. Using the pattern generator developed and presented in [9], I have shown that this scheme works on HRP-2. A main problem of this initial work was the time taken by the off-shelf solver to find a solution. We proposed a new algorithm to solve this specific optimization problem which is 10 times faster than the off-shelf solver [27].

## 2.2 Generalized Inverted Kinematics

### 2.2.1 Visual servoing, walking and Stack of Tasks

In this work with Nicolas Mansard, we addressed the problem of real-time motion generation once the problem of CoM trajectory generation has been tackled [38, 26]. The formalism relies on the task formulation introduced in [SLE91], and allows creating a whole control architecture by combining and prioritize controllers.

The two novelties proposed by Mansard and experimented on HRP-2 are:

- a new formulation ensuring a smooth transition between tasks,
- a mechanism dealing with constraints and the tasks activating these constraints.

### 2.2.2 Real-time (Self-)Collision avoidance

In this work, I have developed a controller to be included in the previous architecture to deal with the problem of self-collision [34]. This work relies on the results of Adrien Escande on a strictly convex representation of bodies. It provides a smooth gradient of the distance between objects. This in turn ensures that there is no discontinuity in the control law. A demonstration showing the HRP-2 robot reactively avoiding a ball has been realized.

### 2.2.3 Human-Humanoid robot real-time interaction

In the frame of the European project ROBOT@CWE, we have developed an application using the technologies described in this chapter to realize collaborative tasks with the humanoid robot HRP-2. Among the instances a direct human-humanoid robot interaction has been realized [24].

## 2.3 Whole body motion and CoM trajectory generation

In this specific work, we do not separate the problem of CoM trajectory generation and the problem of articular values trajectories generation. Instead we rely on the dynamic filter, explained in paragraph 3.3, to obtain a CoM trajectory and articular values trajectories which fit some constraints.

### 2.3.1 Toe-joint

The efficiency of walking with a simulated humanoid robot having a toe-joint is addressed in [44, 9]. More specifically together with Ramzi Sellaouti we have adapted the popular walking pattern generator described in [KKK+03a] to cope with the surface in contact of a passive toe-joint. A new desired ZMP trajectory has been designed and an efficient analytical inverse kinematics has been derived from the toe-joint model. This allows generating complete articular trajectories and faster walking speed than normal flat feet.

### 2.3.2 Dynamically stepping over large obstacles

It has been largely argued that legged robots are mechanically capable of dealing with complex environment and particulary to cross over large obstacles. However to do it in an efficient manner, humanoid robots have to deal with dynamical balance, joint limits and self-collision. The satisfaction of those three constraints is treated in a general manner without the environment in paragraph 6.4. For an obstacle enclosed in a bounding box, together with Björn Verrelst we have proposed a solution [53, 46, 48, 6] to this problem by relying on the robustness of the dynamic filter described in [KKK+03a]. The main idea is to use the same algorithm than for normal walking, but also to lower the height of the robot to avoid collision with the obstacle. Even though it appears as a violation of the constraint needed to have a linear inverted pendulum, if the acceleration of the robot CoM height is negligeable compare to the gravity, this assumption is still valid.

## 2.4 Foot-steps planning

### 2.4.1 Planning foot-steps and pattern generator

Florent Lamiraux proposed to use a non-holonomic mobile robot model to plan a trajectory for the bounding box of HRP-2 in a standard environment. Using the pattern generator presented in [9], he derived a bound on the curvature of the trajectory. The trajectory provided by this model is used to place steps. From this approach we realized an experiment where the humanoid HRP-2 is included in a collaborative working environment [7, 35]. Furthermore, in [41] I presented a model mixing a full upper humanoid robot model while the legged part is a box with a variable height. Using the dynamic filter of [KKK+03a], all the dynamical effects induced by the upper body are handled by the pattern generator and are not considered at the planning level.

### 2.4.2 A novel method for fast foot-steps feasibility tests

When planning foot-steps several constraints must be taken into account, the most important are: dynamic balance, joint limits and collision avoidance. Generating all the articular values for a given interval of time and testing these constraints on such trajectory is very time consuming. For instance using random shooting based algorithm it can takes several hours [8]. To solve this problem in a timely manner, with Nicolas Perrin, we proposed to build an approximation scheme to test the feasibility of a foot-step [24]. Although building this approximation takes several days, its evaluation is done in 9 $\mu s$.

# Chapter 3

# Problem statement, formulations and some solutions

## 3.1 Problem statement

### 3.1.1 General formulation

For a given model of the environment $\mathbf{v}(t)$ at time $t$, it is assumed that a set of contact points can be provided. They can be provided by a contact planner [Hau08, EKMG08, HMN+09], or in the case of coplanar horizontal contacts by a foot planner [KKN+02] or [4]. A motion is said to be dynamically stable and feasible for a humanoid robot if the trajectory $\mathbf{u}(t)$ over a time interval $[0, T]$ fulfills the following constraint:

Motion Constraints Satisfaction Problem ($MCSP_{\mathbf{u}}$)

$$\mathbf{M}_1(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{N}_1(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_1(\mathbf{q}) = \mathbf{T}_1(\mathbf{q})\mathbf{u} + \mathbf{C}_1^\top(\mathbf{q})\lambda \tag{3.1}$$

$$\mathbf{M}_2(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{N}_2(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_2(\mathbf{q}) = \mathbf{C}_2^\top(\mathbf{q})\lambda \tag{3.2}$$

$$\mathcal{A}\,\lambda \geq 0 \tag{3.3}$$

$$\mathbf{u}_{min} < \mathbf{u} < \mathbf{u}_{max} \tag{3.4}$$

$$\hat{\mathbf{q}}_{min} < \hat{\mathbf{q}} < \hat{\mathbf{q}}_{max} \tag{3.5}$$

$$d(\mathcal{B}_i(\mathbf{q}), \mathcal{B}_j(\mathbf{q})) > \epsilon, \forall p(i,j) \in \mathcal{P} \tag{3.6}$$

where $\mathbf{u}(t)$ are the joints torque, $\mathbf{q}(t) = [\mathbf{r}(t)^\top \hat{\mathbf{q}}(t)^\top]^\top$, $\mathbf{r}(t)$ the robot's free-flyer, $\hat{\mathbf{q}}(t)$ its joint values, $\lambda(t)$ are the forces related to the contact points at time $t$, $\mathbf{M}_i$ the inertia matrix where $i = 1$ for the actuated part of the robot, and $i = 2$ for the part of the robot without actuation. $\mathbf{G}_i$ represents the gravity, $\mathbf{N}_i$ the coriolis forces, $\mathbf{T}_1$ is the transpose of the articular Jacobian to the actuated joints, $\mathbf{C}_i$ is the articular Jacobian to the contact points. Eq.(3.1) gives the actuated dynamics, whereas eq.(3.2) gives the under-actuated one. Eq.(3.3) specifies the constraints on the force to be in the friction cone, i.e. that the robot is dynamically stable at each

time $t$. Eq.(3.4) specifies the torque limits of each joint, eq.(3.5) the joint limits, while eq.(3.6) specifies that for a pair $p(i,j)$ two bodies $\mathcal{B}_i$ and $\mathcal{B}_j$ should always be at a distance greater than $\epsilon > 0$.

### 3.1.2 Supplementary constraints

This satisfaction problem can be reformulate in an optimization problem using supplementary constraints to realize tasks [SLE91] such as looking at an object with the head or grasping [KLK+09]. A particular treatment of these additional constraints is given in section 5.1. It is also possible to take explicitly into account impact [WC06]. There are two classical ways to control a robot either through its joint torques $\mathbf{u}$, or using its articular joint values $\mathbf{q}$. Finding $\mathbf{u}(t)$ or $\mathbf{q}(t)$ is here called the Motion Constraint Satisfaction Problem ($MCSP_{\mathbf{u}}$,$MCSP_{\mathbf{q}}$). When $\mathbf{v}(t)$ is constant the contact points are also constant and a trajectory satisfying eq.(3.1-3.6) can be found off-line. One method consists in solving an associated optimization problem minimizing a cost function along an interval. A classical cost function is the integral over the trajectory of the control torques to minimize the energy consumption.

The constraint on the generalized contact forces implies to know the friction coefficient of the soil. This coefficient has to be provided by the user. Finally building constraint defined by eq.(3.3) from the Coulomb cone is not straightforward if torque limits are considered [Esc08]. However if the torque limits, eq.(3.4), are ignored the construction of the constraints related to the friction cones, eq.(3.3), can be realized in an efficient manner [BELK09].

This formulation is interesting because it includes walking. Solving $MCSP_{\mathbf{u}}$ allows walking on uneven terrain, climbing, and walking while interacting with the environment. This formulation has however several difficulties to overcome for practical implementations.

### 3.1.3 Execution on the real robot

The $\mathbf{u}(t)$ here refers to the robot torques, whereas users have access to the motors. Most human-size humanoid robot are equipped with DC motors and gear systems. Gear systems in humanoid robot's legs have usually a high reduction ratio. For instance H7 has a 180 gear reduction ratio for the harmonic drives located in the hip joints [IKN05]. Thus a precise estimation of the friction forces related to these harmonic drives is difficult. Moreover their dynamical properties change dramatically during walking because the weight of the robot is shifting constantly from one leg to the other.

A proper force control for a humanoid robot would need a precise force measurement on the joint side, and a high control bandwidth like the one implemented in Justin [ASOH07]. One difficulty lies on finding an appropriate force measurement de-

vice which is precise enough to estimate these friction forces. It must also be robust enough to handle the impact during foot landing. Moreover a force control scheme needs high speed control loop and therefore adequate CPU power which can be difficult to embed in the robot body. For these reasons, most of humanoid robots do have a position based control loop at the joint-level. It is worth noting that force control based has already been achieved on humanoid robots controlled by a position-based scheme without walking [KTYP08, AKOI06, YK09] to control a position based robot with a torque formulation. Still the last version of Asimo (November 2011) uses a position based approach for walking. Considering $MCSP_{\mathbf{q}}$, it is possible to build optimization problem minimizing the torques [SYLM07, WC06]. But the computational time necessary to find a feasible trajectory is generally quite large (few hours for [SYLM07]). It is therefore generally considered at the planning level and not at the control level.

Finally, human size robots such as ASIMO and HRP-2 embed a compliant material between the ankle and the soil to absorb impacts during foot landing [NHK+07]. Compensating the deformation of this material while walking using a force based control consists in integrating its model in eq.(3.1-3.2). When using a position-based control scheme, a controller can be designed to take into account this passivity by modifying the desired articular values to follow a desired momentum reference [KYST01, KNKH07].

### 3.1.4 The Inverted Pendulum: a reduced model for walking

When considering reactive motions, for instance while interacting with a human, modification of contact points should be taken into account in a timely manner. A rather large amount of existing work aims at simplifying $MCSP_{\mathbf{u}}$ to achieve real-time performances in the context of bipedal walking. It consists first in ignoring the last three constraints (collisions, torque and articular limits- eq.(3.4-3.6)) and in replacing the constraint of the friction cone eq.(3.3) by assuming that all contacts are coplanar and horizontal. Then this constraint can be rewritten using a stability indicator[1] based on the dynamic wrench of the robot. The problem parameterized on the trajectories of the multi-body robot joints $\mathbf{q}(t)$ becomes:

Finding $\mathbf{q}(t)$ for appropriate ZMP multibody ($ZMPMB_{\mathbf{q}}$)

$$\mathbf{Z}(\mathbf{q}) = \frac{mg\mathbf{c}(\mathbf{q}) + (\mathbf{n})_{\times}\mathbf{W}_r(\mathbf{q})}{mg + \mathbf{W}_t(\mathbf{q}).\mathbf{n}} \tag{3.7}$$

$$\mathbf{A}\mathbf{Z}(\mathbf{q}) < \mathbf{b} \tag{3.8}$$

where $\mathbf{A}, \mathbf{b}$ represents the linear constraints related to the convex envelope of the contact points, $\mathbf{n}$ a vector orthogonal to the plane where the contact lie, $\mathbf{c}$ is the Center Of Mass (CoM), $\mathbf{W}_t$ the linear part of the humanoid dynamic wrench, and $\mathbf{W}_r$ its rotational part. This problem can be further simplified if the constraints,

[1] This stability indicator is the ZMP when this constraint is satisfied

given by eq.(3.8), are replaced by a desired trajectory $\mathbf{Z}^{ref}(t)$ which leads to solve:

Finding $\mathbf{q}(t)$ for approriate ZMP multibody reference ($ZMPMB^{ref}_{\mathbf{q}}$)

$$\mathbf{Z}^{ref}(t) = \frac{mg\mathbf{c}(\mathbf{q}(t)) + (\mathbf{n})_{\times}\mathbf{W}_r(\mathbf{q}(t))}{mg + \mathbf{W}_t(\mathbf{q}(t)).\mathbf{n}} \tag{3.9}$$

Solving $ZMPMB^{ref}_{\mathbf{q}}$ amounts to solve a coupled second order differential equation with $n$ unknowns. Apart numerical methods similar to the previously optimization problems, efficient solutions are possible only by adding additional constraints. The method proposed by Nishiwaki is one of them [NKK+07]. It assumes that all the bodies of the robot perform the same sagital motion. Eq.(3.9) then can be simplified and can be formulated as a trinomial equation using boundaries conditions.

The most common simplification is to consider the robot as an inverted pendulum [KYK92]. However even in this case the obtained system is a pair of coupled differential equations. This can be solved by adding a constraint on the motion of the CoM inverted pendulum. If it is constrained in a plan, the equations can be linearized. The subsequent model is known as the Linearized Inverted Pendulum Model (3D-LIPM) or cart-model. This gives us two equivalent problems to $ZMPMB_{\mathbf{q}}$ and $ZMPMB^{ref}_{\mathbf{q}}$:

Finding $\mathbf{c}$ with constrained ZMP ($ZMPLIPM$)

$$\mathbf{Z} = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} c_x - \frac{c_z}{g}\ddot{c}_x \\ c_y - \frac{c_z}{g}\ddot{c}_y \\ 0 \end{bmatrix} \tag{3.10}$$

$$\mathbf{AZ} < \mathbf{b} \tag{3.11}$$

Finding $\mathbf{c}$ with reference ZMP ($ZMPLIPM^{ref}$)

$$\mathbf{Z}^{ref} = \begin{bmatrix} c_x - \frac{c_z}{g}\ddot{c}_x \\ c_y - \frac{c_z}{g}\ddot{c}_y \\ 0 \end{bmatrix} \tag{3.12}$$

When considering a set of steps, and assuming that the ZMP trajectory is a third order polynomial, it is possible to solve this problem analytically [Kaj05, HKKH06, MHK+07]. An equivalent approach has been used for popular humanoid robots such as QRIO [NKS+04]. These latest methods have provided numerical solution which can be computed in less than 1 $ms$ in robot such as HRP-2. Polynomial trajectories however have to be handled carefully as they might generate unnecessary motions especially when connecting two trajectories. Kajita [KKK+03a] proposed to solve this problem by associating $ZMPLIPM^{ref}$ with an optimization problem called the preview control.

Most of my contributions take their roots from the preview control and the methods to make the robot track the subsequent CoM trajectory. For this reason, the preview control is now described in more details.

## 3.2 Kajita's Preview control

The preview control approach consists in taking into account the future by integrating a preview window [KKK+03a]. $ZMPLIPM^{ref}$ is then solved by creating an optimization problem using a performance index. For a given axis of the discretized CoM trajectory, e.g. $c_x(k)$, this gives:

Preview Control($PC$)

$$\min_{\dddot{c}_x(k),\cdots,\dddot{c}_x(k+N_L)} \sum_{i=k}^{k+N_L-1} \frac{1}{2}Q(p_x(i+1)-p_x^{ref}(i+1))^2 + \frac{1}{2}R\dddot{c}_x^2(i) \tag{3.13}$$

$$\mathbf{c}_x(k+1) = \mathbf{A}\mathbf{c}_x(k) + \mathbf{B}\dddot{c}_x(k) \tag{3.14}$$

$$p(k) = \mathbf{C}\mathbf{c}_x(k), \tag{3.15}$$

where the first term of eq.(3.13) represents the difference between the actual ZMP and the desired one, and the second term is the jerk of the CoM ($\dddot{c}_x$). $Q$ and $R$ are weights influencing the importance of each part in the cost function. $N_L$ is the size of the preview window. $\dddot{c}_x$ is also the command vector of the system. Using an iterative scheme, and assuming that both CoM and ZMP trajectories are discretized by piecewise cubic polynomials over time intervals of constant length $T$, the recursive relationship described by eq.(3.14) and eq.(3.14) is obtained where

$$\begin{aligned} \mathbf{c}_x(k) &\equiv [c_x(k)\ \dot{c}_x(k)\ \ddot{c}_x(k)]^T, \\ \mathbf{A} \equiv \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{B} &\equiv \begin{bmatrix} T^3/6 \\ T^2/2 \\ T \end{bmatrix}, \mathbf{C} \equiv \begin{bmatrix} 1\ 0\ -\frac{c_z}{g} \end{bmatrix} \end{aligned} \tag{3.16}$$

Finally the optimal controller which minimizes the performance index, formulated by eq.(3.13), is given by:

$$\dddot{c}_x(k) = -K_1 \sum_{i=0}^{k} e(i) - K_2 c_x(k) - \sum_{j=1}^{N_L} K_p(j) p^{ref}(k+j) \tag{3.17}$$

where $e(i) = p_x(i) - p_x^{ref}(i)$, and $K_1, K_2$ and $K_p(j)$ are the gains calculated from the weights $Q$ and $R$ and the parameters used in the system described by eq.(3.14).

It is interesting to note the unicity of the weights $K_p(j)$ for a given control period $T$, preview window $N_L$ and constant CoM height $c_z$. The gains $K_p(j)$ gives the importance of the relationship between the future and the current state. It is important when modifying the future to take into account the current ZMP pertubations [NK07]. The graphs in Fig.3.1 indicates the variation of CoM has little effect on the relationship between the current state and the immediate future. However the control time is of paramount importance.

The most common problem with the simplification of the robot model is to keep the consistency with this model. Indeed when the robot is considered as an inverted

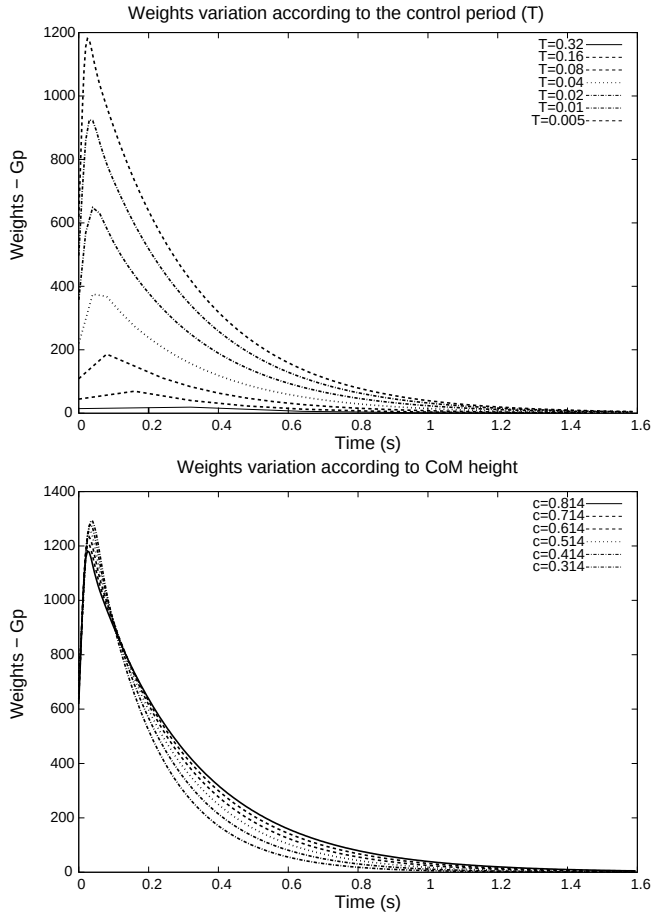



Figure 3.1: Variation of the weights according the CoM height and the control period.

pendulum, the free variables are commands on the robot CoM. Once the CoM trajectory is found by solving $ZMPLIPM^{ref}$ $\mathbf{q}(t)$ is generated using a Generalized Inverted Kinematics control scheme. In addition to ensure that the ZMP will not be pertubated, the angular momentum of the robot around its CoM should be controlled [KKK+03b]. The latter one has been used successfully by Neo [NYK+05] to achieve real-time motion in the context of teleoperation. However this might not be feasible if the legs are too heavy, and induce an important inertial effect when moving [BLB+07]. This calls generally for multiple single mass models taking into account the inertia created by the motion of the legs. Another method is to use a dynamic filter which modifies the CoM trajectory to have the robot multi-body model generates the desired ZMP trajectory [KKK+03a].

## 3.3 Dynamic filter

When the previous method generates a CoM trajectory, and a motion generator is used to generate an articular trajectory, it may happens that the resulting ZMP is not the desired one. A second preview control can be used to correct this difference. The first and second preview controllers proposed by Kajita in [KKK+03a] are generally seen as a kind of Newton-Raphson resolution of the following equation:

$$\mathbf{p}^{ref}(t) = \mathbf{f}(\mathbf{q}(t)) \tag{3.18}$$

where $\mathbf{f}$ is the function based on the multi-body robot's model to compute the ZMP. The difficulty is in computing $\mathbf{f}^{-1}$ to obtain:

$$\mathbf{q}^*(t) = \mathbf{f}^{-1}(\mathbf{p}^{ref}(t)) \tag{3.19}$$

Let us assume that $\mathbf{f}$ is a composition of two functions:

$$\mathbf{f}(\mathbf{q}(t)) = \mathbf{c}(\mathbf{g}(\mathbf{q}(t))) \tag{3.20}$$

with $\mathbf{c}$ the function giving the ZMP trajectory for a given CoM trajectory, and $\mathbf{g}$ the function giving the CoM trajectory for a given set of articular trajectories. The preview controller with the linearized single mass model give us $\tilde{\mathbf{c}}^{-1}$ which is an approximation of $\mathbf{c}^{-1}$. To approximate $\mathbf{g}^{-1}$ which in our case should follow the CoM reference given by $\tilde{\mathbf{c}}^{-1}$ we can assume that the CoM and the waist are rigidly connected. In case of HRP-2 we have 12 DOFs and 12 constraints (foot position) and uses inverse geometry to have $\tilde{\mathbf{g}}^{-1}$. Thus the first stage of the preview give us:

$$\mathbf{q}_0^*(t) = \tilde{\mathbf{g}}^{-1}(\tilde{\mathbf{c}}^{-1}(\mathbf{p}^{ref}(t))) \tag{3.21}$$

To evaluate the approximation, we can recompute the ZMP value for $\mathbf{q}_0^*(t)$ using $\mathbf{f}$ which is much easier to compute than $\mathbf{f}^{-1}$. As $\tilde{\mathbf{c}}^{-1}$ is linear, the idea of Kajita [KKK+03a] with the second preview loop is to perform a form of Newton-Raphson method on the CoM reference trajectory. Indeed the new CoM reference trajectory is given by:

$$\tilde{\mathbf{c}}^{-1}(\mathbf{p}^{ref}(t)) + \tilde{\mathbf{c}}^{-1}(\mathbf{p}^{ref}(t) - \mathbf{f}(\mathbf{q}_0^*(t))) \tag{3.22}$$

The final solution is therefore:

$$\mathbf{q}_1^*(t) = \tilde{\mathbf{g}}^{-1}(\tilde{\mathbf{c}}^{-1}(\mathbf{p}^{ref}(t)) + \tilde{\mathbf{c}}^{-1}(\mathbf{p}^{ref}(t) - \mathbf{f}(\mathbf{q}_0^*(t))) \tag{3.23}$$

Such method does not guarantee the convergence, and might suffer from numerical unstability. However several different implementations have proven its efficiency in this specific problem [KKK+03a, NK07, 9]. The second preview controller completely compensate for the use of the simplified model, because the result of the second iteration $\mathbf{q}_1^*(t)$ is much better than the first one $\mathbf{q}_0^*(t)$ using $\mathbf{f}(\mathbf{q}_0^*(t))$. Indeed Fig.6.6 bottom shows the difference between $\mathbf{f}(\mathbf{q}_0^*(t))$ and $\mathbf{f}(\mathbf{q}_1^*(t))$. It would be possible to apply this approach several times, but Fig.6.6 indicates that $\mathbf{f}(\mathbf{q}_1^*(t)) \equiv \mathbf{p}^{ref}(t)$

# Chapter 4

# CoM trajectory, Inverted Pendulum and Quadratic Program

This work is the result of a fruitful collaboration with Pierre Brice Wieber which started from his two months stay at JRL in 2005, and continued during his stay from June 2008 to June 2010 at JRL. It consists mostly in focusing on the Inverted Pendulum model detailled in the previous chapter and reformulating the Preview Control ($PC$) problem as a quadratic program with constraints. One of the immediate benefit is to guarantee the balancing. A less direct one is to allow the design of a dedicated efficient solver. In the final section a recent extension to control the CoM speed and decide foot-step position is briefly introduced [HPW10]. One of its application was to generate in real-time walking using visual information [19, 73].

## 4.1 Foundation as Quadratic Program

### 4.1.1 Quadratic reformulation of the preview control

Wieber in [Wie06] suggested to reformulate the optimization problem formulated by Kajita in [KKK+03a] using the $N_L$ versions of eq.(3.14):

$$\begin{bmatrix} p_x(k+1) \\ \vdots \\ p_x(k+N_L) \end{bmatrix} = \begin{bmatrix} 1 & T & T^2/2 - c_z/g \\ \vdots & \vdots & \vdots \\ 1 & N_L T & N_L^2 T^2 - c_z/g \end{bmatrix} \mathbf{c}_x(k) + \begin{bmatrix} T^3/6 - Tc_z/g & 0 & 0 \\ \vdots & \ddots & \vdots \\ (1 + 3N_L + 3N_L^2)T^3/6 - Tc_z/g & \dots & T^3/6 - Tc_z/g \end{bmatrix} \begin{bmatrix} \dddot{c}_x(k) \\ \vdots \\ \dddot{c}_x(k+N_L-1) \end{bmatrix} \tag{4.1}$$

The second big matrix is a $N \times N$ lower triangular Toeplitz matrix (i.e. with constant diagonals). This relation can be considered in a more compact presentation. This allows to rewrite the control scheme linked with the performance index given by eq.(3.13) as:

Preview Control as a QP ($PC$)

$$\min_{\dddot{\mathbf{C}}_x(k)} \frac{1}{2}(R\dddot{\mathbf{C}}_x^2(k) + Q(\mathbf{Z}_x(k+1) - \mathbf{Z}_x^{ref}(k+1))^2) \tag{4.2}$$

$$\mathbf{Z}_x(k+1) = \mathbf{P}_x\mathbf{c}_x(k) + \mathbf{P}_u\dddot{\mathbf{C}}_x(k) \tag{4.3}$$

where $\mathbf{Z}_x^{ref}(k+1) = [p_x(k+1)\ p_x(k+2)\ \dots\ p_x(k+NL)]^\top$. This can be solved analytically by:

$$\dddot{\mathbf{C}}_x(k) = -(\mathbf{P}_u^T\mathbf{P}_u + \frac{R}{Q}I_{N\times N})^{-1}\mathbf{P}_u^T(\mathbf{P}_x\mathbf{c}_x(k) - \mathbf{Z}_x^{ref})(k) \tag{4.4}$$

where $I_{N\times N}$ is an identity matrix. The reader is kindly invited to read [Wie06] for further details regarding stability and applications in stabilizing a humanoid against strong perturbations.

### 4.1.2 Explicitly taking into account the limits of the ZMP position

One important limitation of the initial algorithm of [KKK+03a] is the reference trajectory of the ZMP given by a heuristic. When using the HRP-2 robot, one practical reason to have this heuristic is due to the stabilizer commercially available. Indeed its role is to make sure that the ZMP is at the center of the convex hull. However, in general, there are various ZMP trajectories for the same foot-steps. Wieber proposed in [Wie06] to further simplify the performance index by solving the following quadratic program:

Preview Control as a constrained QP($PC_c$)

$$\min_{\dddot{\mathbf{C}}_x(k)} \frac{1}{2}\dddot{\mathbf{C}}_x^2(k) \tag{4.5}$$

$$\mathbf{P}_x\mathbf{c}_x(k) + \mathbf{P}_u\dddot{\mathbf{C}}_x(k) \leq \mathbf{Z}^{\max}(k) \tag{4.6}$$

$$-\mathbf{P}_x\mathbf{c}_x(k) - \mathbf{P}_u\dddot{\mathbf{C}}_x(k) \leq -\mathbf{Z}^{\min}(k) \tag{4.7}$$

Practically the generated trajectory tends to be very close to the boundary of the constraint. For this reason, we considered the following program:

Preview Control as a QP with reference ($PC_c^{ref}$)

$$\min_{\dddot{\mathbf{C}}_x(k)} \frac{1}{2}(\dddot{\mathbf{C}}_x^2(k) + \alpha\dot{\mathbf{C}}_x^2(k) + \beta(\mathbf{Z}_x(k) - \mathbf{Z}^{ref}(k))^2) \tag{4.8}$$

$$\mathbf{P}_x\mathbf{c}_x(k) + \mathbf{P}_u\dddot{\mathbf{C}}_x(k) \leq \mathbf{Z}^{\max}(k+1) \tag{4.9}$$

$$-\mathbf{P}_x\mathbf{c}_x(k) - \mathbf{P}_u\dddot{\mathbf{C}}_x(k) \leq -\mathbf{Z}^{\min}(k+1) \tag{4.10}$$

The parameters $\alpha$ and $\beta$ are respectively the weight of the speed and of the heuristic reference trajectory.

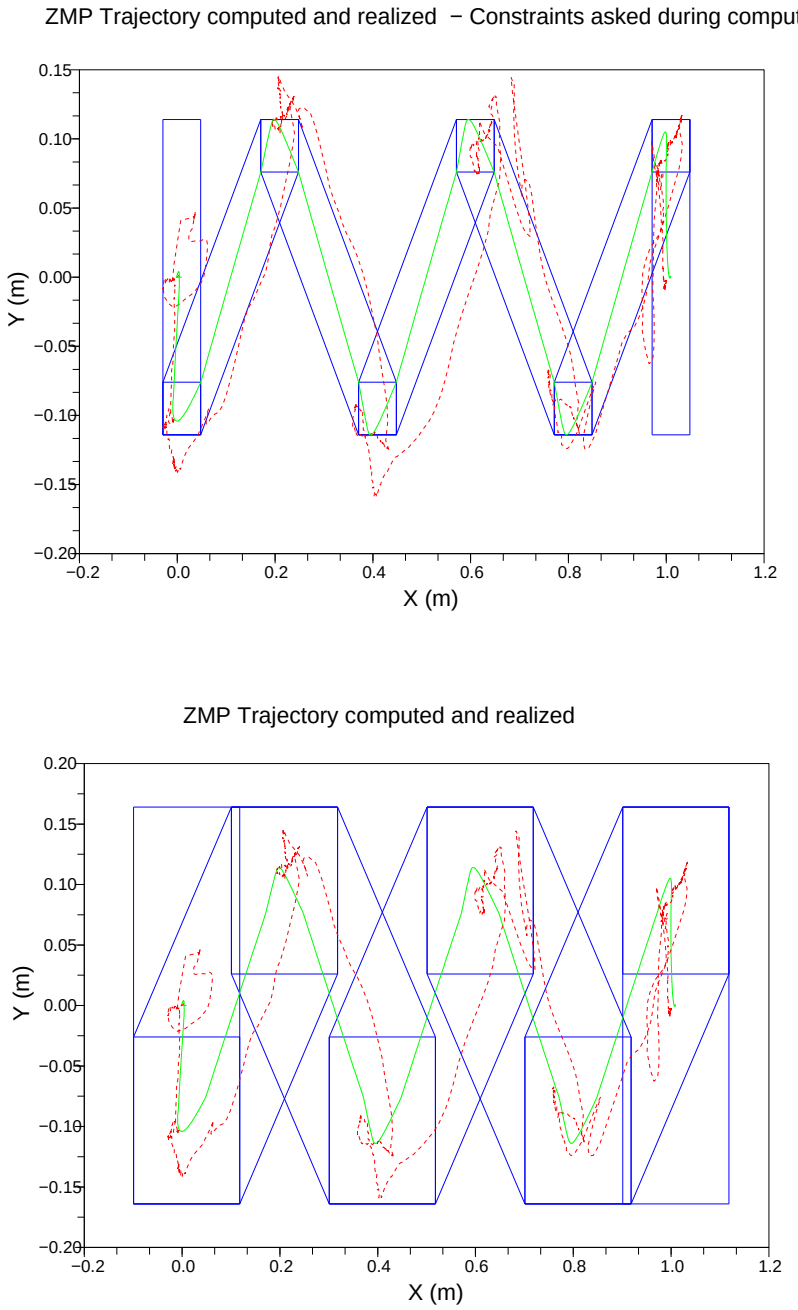



Figure 4.1: (Up) Reference ZMP, realized ZMP and convex polygons with over-constrained feet surfaces; (Bottom) Reference ZMP, realized ZMP and convex polygons when considering the full feet surface.

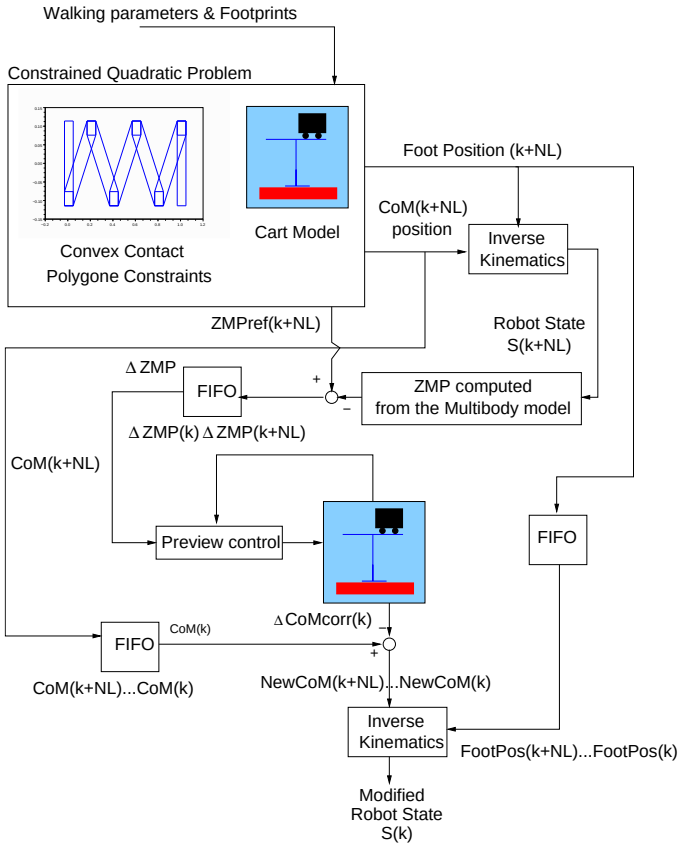



Figure 4.2: Structure of the implemented Walking Pattern Generator

### 4.1.3 Experimental results

The proposed approach was implemented on the HRP-2 robot. It was integrated in our pattern generator software. The overall structure is given in Fig.4.2. More precisely as the resolution of the quadratic problem provides the ZMP and CoM reference trajectories, it replaces the first preview control in the original scheme of Kajita. The second preview control is the Dynamical Filter described in paragraph 3.3. The experiment realized consisted of 5 forward steps with a length of 20 centimeters. To build the constrained quadratic problem we used the following parameters: $T = 10ms$ for time discretization, and $N = 150$ for the preview window. The intermediate values to generate a 5 ms command were created by linear interpolation. Fig.4.1(a) depicts in blue the convex polygons with over-constrained feet's surface. In green, the resulting ZMP reference trajectory is depicted. The over-constrained surface force the ZMP reference trajectory to be on the boundary of the feet and aim to avoid unstable situation. This is important as the realized ZMP trajectory (in red) clearly violates the constraints. However as depicted in Fig.4.1(b) it stays in the convex polygon of the full feet surfaces, and thus is stable. For comparison the standard preview control algorithm runs on the same robot with the same parameters is given in Fig.4.3.

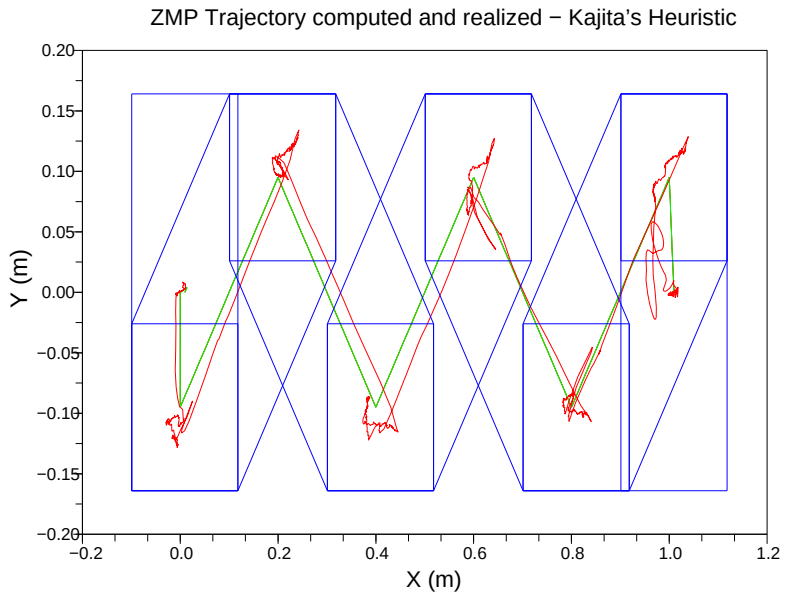



Figure 4.3: Reference ZMP, realized ZMP using Kajita's algorithm with the convex polygon of the feet.

## 4.2 An optimized QP solver

### 4.2.1 Canonical Quadratic Program

Let us reformulate $PC_c^{ref}$ as a canonical Quadratic Program (QP)

Canonical Preview Control($CPC_c^{ref}$)

$$\min_{\mathbf{u}(k)} \frac{1}{2}\mathbf{u}^T(k)\mathbf{Q}\mathbf{u}(k) + \mathbf{p}(k)^T\mathbf{u}(k) \tag{4.11}$$

$$\mathbf{b}'^{l}(k+1) \leq \mathbf{D}(k+1)\begin{pmatrix}\mathbf{P}_{zu} & 0\\ 0 & \mathbf{P}_{zu}\end{pmatrix}\mathbf{u}(k) \leq \mathbf{b}'^{u}(k+1) \tag{4.12}$$

with

$$\mathbf{u}(k) = \begin{pmatrix}\dddot{\mathbf{C}}_y(k)\\ \dddot{\mathbf{C}}_x(k)\end{pmatrix}, \tag{4.13}$$

$$\mathbf{Q} = \begin{pmatrix}\mathbf{Q}' & 0\\ 0 & \mathbf{Q}'\end{pmatrix} \tag{4.14}$$

where $\mathbf{Q}'$ is a positive definite constant matrix, and

$$\mathbf{p}(k)^T = \begin{pmatrix}\mathbf{c}_x(k)^T & \mathbf{c}_y(k)^T\end{pmatrix}\begin{pmatrix}\mathbf{P}_{su} & 0\\ 0 & \mathbf{P}_{su}\end{pmatrix} \tag{4.15}$$

where $\mathbf{P}_{su}$ is also a constant matrix which includes the non quadratic terms of eq.(4.8)(see [DDW+08] for more details).

Using eq.(3.10), the constraints eq.(3.11) on the position of the ZMP can also be represented as constraints on the jerk $u$ of the CoM as given by eq.(4.12).

Since the matrix $\mathbf{Q}$ is positive definite and the set of linear constraints eq.(4.12) forms a (polyhedral) convex set, there exists a unique global minimizer $\mathbf{u}^*(k)$ [NW00].

The number of variables in the minimization problem eq.(4.11) is equal to $n = 2N$ and the number of constraints eq.(4.12) is of the same order, $m \approx 2N$. Typical uses of this Linear Model Predictive Control (LMPC) scheme consider $N = 75$ and $T = 20\,ms$, for computations made on a time interval $NT = 1.5\,s$, approximately the time required to make 2 walking steps [DWFD08]. This leads to a QP which is typically considered as small or medium sized.

Another important measure to take into account about this QP is the number $m_a$ of *active constraints* at the minimum $u^*$, the number of inequalities in eq.(4.12) which hold as equalities. We have observed that at steady state, this number is usually very low, $m_a \leq m/10$, and even in the case of strong disturbances, we can observe that it remains low, with usually $m_a \leq m/2$ [DWFD08].

### 4.2.2 Design choices

The solver developed in this work is based on an active set method, using a primal formulation and the range space of the constraints matrix. Because quite few constraints are active when solving the problem the active set method is faster than interior point method. The primal formulation has the advantage that the algorithm can be stopped and still provide a feasible solution even if it is sub-optimal. The constraint to provide a feasible solution when starting the algorithm can be easily tackled by solving a linear problem. The range space formulation of the constraint is motivated by the fact that its complexity is directly related to the number of active constraints which is quite small. Moreover, as the related matrices are not ill-conditioned the resolution does not perform poorly.

### 4.2.3 Off-Line Change of variables

The first action of a range space active set method is usually to make a Cholesky decomposition of the matrix $\mathbf{Q} = \mathbf{L}_Q\mathbf{L}_Q^T$ and make an internal change of variable

$$\mathbf{v}(k) = \mathbf{L}_Q^T\mathbf{u}(k). \tag{4.16}$$

That way, the Quadratic Problem eq.(4.11) simplifies to a Least Distance Problem (LDP) eq.[Fle81]

$$\min_{\mathbf{v}(k)} \ \frac{1}{2}\|\mathbf{v}(k) + \mathbf{L}_Q^{-T}\mathbf{p}(k)\|^2.$$

In our case, we need to solve online a sequence of QPs eq.(4.11)-(4.12) where the matrices $\mathbf{Q}'$, $\mathbf{P}_{zu}$ and $\mathbf{P}_{su}$ are constants. We can therefore make this change of

variable completely off-line and save a lot of online computation time by directly solving online the following Least Distance Problem:

Least Distance Problem($LDP_c^{ref}$)

$$\min_{\mathbf{v}(k)} \ \frac{1}{2}\|\mathbf{v}(k) + \mathbf{p}'(k)\|^2 \tag{4.17}$$

$$\mathbf{b}'^{l}(k+1) \leq \mathbf{D}(k+1) \begin{pmatrix} \mathbf{P}_{zu}\mathbf{L}_Q^{-T} & 0 \\ 0 & \mathbf{P}_{zu}\mathbf{L}_Q^{-T} \end{pmatrix} \mathbf{v}(k) \leq \mathbf{b}'^{u}(k+1) \tag{4.18}$$

with

$$\mathbf{p}_k'^T = \begin{pmatrix} \mathbf{c}_x(k)^T & \mathbf{c}_y(k)^T \end{pmatrix} \begin{pmatrix} \mathbf{P}_{su}\mathbf{L}_Q^{-T} & 0 \\ 0 & \mathbf{P}_{su}\mathbf{L}_Q^{-T} \end{pmatrix} \tag{4.19}$$

Realizing this change of variable off-line allows saving $n^2$ *flops* at each iteration of our algorithm. Note that, we measure computational complexity in number of floating-point operations, flops. We define a flop as one multiplication/division together with an addition. Hence, a dot product $\mathbf{a}^T\mathbf{b}$ of two vectors $\mathbf{a}$, $\mathbf{b} \in \mathbb{R}^n$ requires $n$ flops.

## 4.3 Constraint activation

We have observed that not removing constraints does not affect the result we obtain from our LMPC scheme in a noticeable way. From the implementation viewpoint this allows to implement very efficient updates of the Cholesky decomposition of the constraint matrix. By observing the Lagrangian multipliers we can guess which constraints will be activated for the next iteration. Our final guess for the active set when doing so is in most cases correct or includes only one, and in rare cases two unnecessarily activated constraints. This leads to slightly sub-optimal solutions, which nevertheless are feasible. Furthermore, we have observed that, this does not affect the stability of our scheme: the difference in the generated walking motions is negligible.

## 4.4 Numerical results

A first resolution of $LDP^{ref}$ relied on QL [Sch05], a state of the art QP solver implementing a dual active set method with range space linear algebra. The fact that it implements a dual strategy implies that it cannot be interrupted before reaching its last iteration since intermediary iterations are not feasible. Furthermore, no possibilities of warm starting are offered to the user. However, since it relies on a range space algebra, comparisons of computation time with our algorithm without warm starting are meaningful.

We naturally expect to gain $n^2$ flops at each iteration thanks to the off-line change of variable. Furthermore, QL does not implement double sided inequality constraints

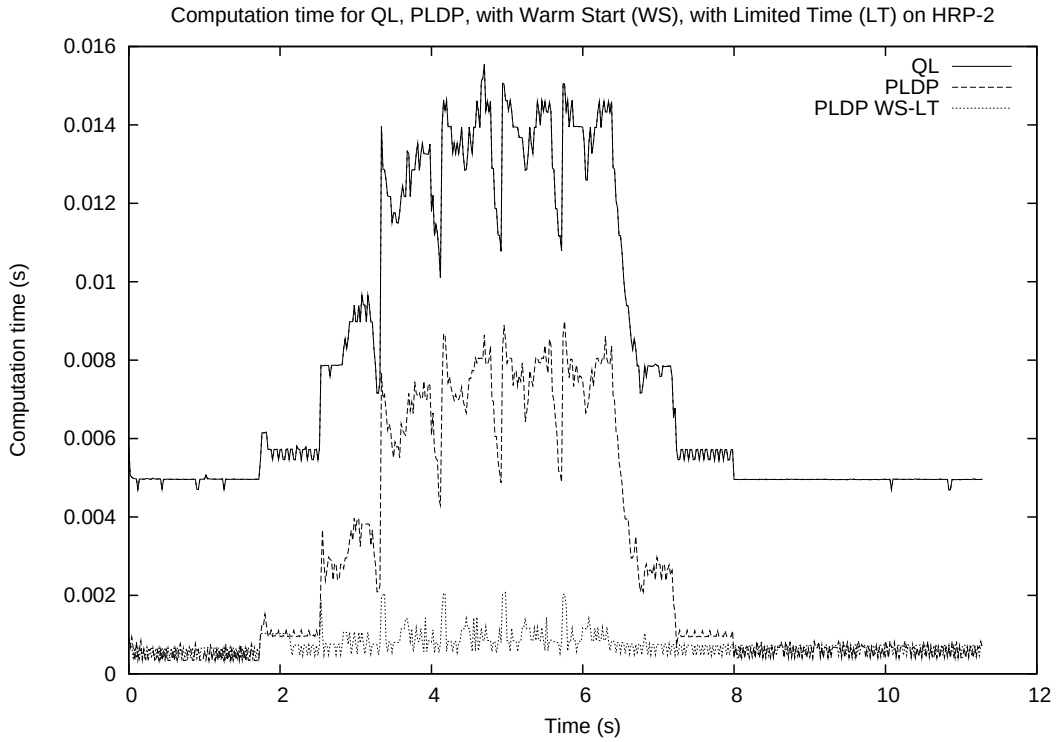



Figure 4.4: Computation time required by a state of the art generic QP solver (QL), our optimized solver (PLDP), and our optimized solver with warm start and limitation of the computation time, over 10 seconds of experiments.

like the ones we have in eq.(4.18), so we need to double artificially the number $m$ of inequality constraints. Since testing the constraints requires $nm$ flops at each iteration and $m \approx n$ in our case, that's a second $n^2$ flops which we save with our algorithm. The mean computation time when using QL is 7.86 ms on the CPU of our robot, 2.81 ms when using our Primal Least Distance Problem (PLDP) solver. Detailed time measurements can be found in Fig. 4.4.

Even more interesting is the comparison with our warm start scheme combined with a limitation to two iterations for solving each QP. This generates short periods of sub-optimality of the solutions, but with no noticeable effect on the walking motions obtained in the end: this scheme works perfectly well, with a mean computation time of only 0.74 ms and, most of all, a maximum time less than 2 ms!

A better understanding of how these three options relate can be obtained from Fig. 4.5, which shows the number of constraints activated by QL for each QP, which is the exact number of active constraints. This figure shows then the difference between this exact number and the approximate number found by PLDP, due to the fact that we decided to never check the sign of the Lagrange multipliers. Most often, the two algorithms match or PLDP activates only one constraint in excess. The difference is therefore very small.

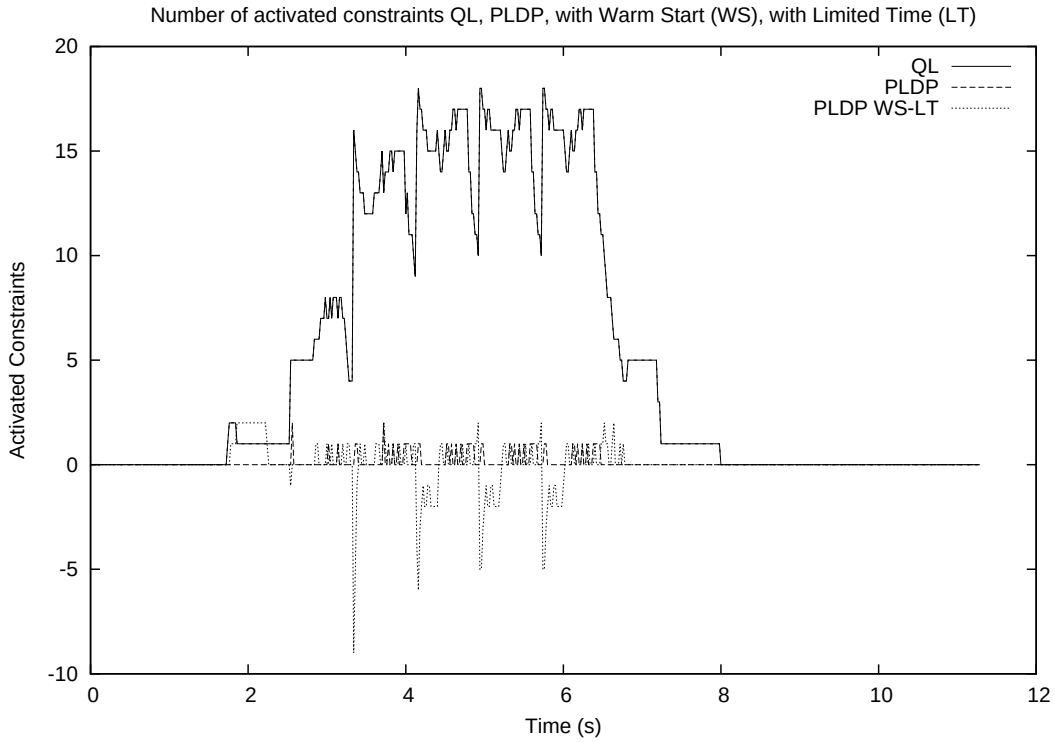



Figure 4.5: Number of active constraints detected by a state of the art solver (QL), difference with the number of active constraints approximated by our algorithm (PLDP), between 0 and 2, and difference with the approximation by our algorithm with warm start and limitation of the computation time, between -9 and 2.

This difference naturally grows when implementing a maximum of two iterations for solving each QP in our warm starting scheme: when a whole group of constraints needs to be activated at once, this algorithm can identify only two of them each time a new QP is treated. The complete identification of the active set is delayed therefore over subsequent QPs: for this reason this algorithm appears sometimes to miss identifying as many as 9 active constraints, while still activating at other times one or two constraints in excess. Note that, regardless of how far we are from the real active set, the solution obtained in the end is always feasible.

## 4.5 Extensions and applications to vision

In [HHW$^+$10], Herdt et al. propose to introduce feet positions as free variables in the problem $PC_c^{ref}$. This is realize thanks to the explicit constraints on the ZMP, and by decoupling position and orientation. Indeed the latter one is solved separately to avoid introducing non-linear terms in the problem. To deal with the issue of foot-step transition feasibility, the authors modify an initial guess given by an external process,

and limits the distance to this initial guess.

In [HPW10], Herdt el al. use the results of the work presented in 6.4 to remove the necessity to provide an initial guess. In addition the cost function of the Canonical Problem $CPC_c^{ref}$ is modified in order to let the user specify a desired CoM velocity. It is then possible to specify in real-time a given direction of motion and let the overall control system find the appropriate joint trajectories by combining the walking and whole-body methods presented so far.

In [73], together with Claire Dune, we have used this approach to implement a behavior where the function $f$ of eq.(1.1) is the difference between a desired pose of a known object in the image, and the current pose of this object in the image. This problem can be formulated as a visual-servoing problem. One difficulty is related to the constraints. Indeed if it is possible to compute the desired speed of the center of mass to decrease the function $f$, it is not guaranteed that this speed can be realized. To cope with this problem a new control law is presented in [19, 73]. It consists in introducing a virtual feature which compute the error accumulated and compensate for it. Assuming that this error has a constant integration over one period of walking, this allow us to converge towards the goal. Thanks to this method, it was possible to drive HRP-2 by simply providing the desired pose of the known object and the system was generating the motion automatically. A video is available here: http://www.youtube.com/watch?feature=player_embedded&v=5nb08c45I3k.

# Chapter 5

# Whole-body motion

In this chapter we described the framework used to generate reactive motion based on sensory information.

The following work is a fruitful collaboration with Nicolas Mansard which started in 2006 during a two months stay in the frame of a JSPS summer scholarship. After his PhD defense, he joined our group for 6 months during which he wrote a generic library implementing the framework called the Stack Of Tasks [26]. I participated to the writing of this library, and my main contribution lies in the interaction with walking.

In this section, we describe a general framework for building complex whole-body control for highly redundant robot, and we propose various applications illustrating its implementation. The key idea is to divide the control into several sensor-based control tasks that are simultaneously executed by a general structure called *stack of tasks*. This structure enables a very simple access for task sequencing, and can be used for task-level control. This framework was applied for a visual servoing task, reactive walk, and human-humanoid robot interaction.

The Generalized Inverted Kinematics (GIK) introduced by Nakamura et al. [NH86] to control redundant robots is widely used in humanoid robotics [GJG05, NYK+05], as well as its counter part in the force domain the Operational Space approach [Kha87, SK06]. Based on the notion of task [SLE91], priority between tasks is introduced by projecting the tasks with lower priority in the kernel of tasks having a higher priority. Initially considering only two tasks the work of Nakumura has been extended by Siciliano [SS91] to an iterative scheme shortly presented below.

The Stack of Tasks as a software component currently mostly implements the GIK formalism in an efficient manner. However the Stack of Tasks as a framework can be extended to other formalisms. Software projects to control robots already exist, such as ViSP [MSC05] dedicated to provide all the tools necessary to realize visual servoing. Other projects such as Orocos [Bru01] provides a framework and tools to

build robots controllers.

## 5.1 Stack of Tasks

### 5.1.1 Goals

The stack of tasks is a structure that orders the tasks currently active. Only the tasks in the stack are taken into account in the control law. The task at the bottom level has priority over all the others, and the priority decreases as the stack level increases. The control law is computed from the tasks in the stack, in accordance with three rules:

- any new task added in the stack does not disturb the tasks already in the stack.
- the control law is continuous, even when a task is added or removed from the stack. The robot is controlled through the articular velocity $\dot{\mathbf{q}}$. A break of continuity would mean an infinite acceleration during a short period of time, which would imply that the control is not correctly applied.
- if possible, the additional constraints should be added to the control law, but without disturbing the tasks in the stack.

### 5.1.2 Task definition

A task $\mathbf{e_i}$ is defined as a difference between a desired feature $\mathbf{s_i^*}$ and its current value $\mathbf{s_i}$:

$$\mathbf{e_i} = \mathbf{s_i} - \mathbf{s_i^*} \tag{5.1}$$

The current value of the feature and the velocity $\mathbf{v}$ of a point on the robot are usually related by the equation:

$$\dot{\mathbf{s_i}} = \mathbf{L_{s_i}}\mathbf{v} \tag{5.2}$$

where $\mathbf{L_{s_i}}$ is called the interaction matrix in the field of visual servoing. Let $\mathbf{q}$ be the vector of the robot articular positions. The Jacobian of $\mathbf{e_i}$ is noted $\mathbf{J_i}$ and is defined by:

$$\dot{\mathbf{e}}_\mathbf{i} = \frac{\partial \mathbf{e_i}}{\partial \mathbf{q}} = \mathbf{J_i}\dot{\mathbf{q}} \tag{5.3}$$

$\mathbf{J_i}$ is assumed to be of full rank. Assuming that the robot is controlled using $\dot{\mathbf{q}}$, we can compute its value using:

$$\dot{\mathbf{q}}_i = \mathbf{J_i^+}\dot{\mathbf{e}}_\mathbf{i}{}^* \tag{5.4}$$

where $\dot{\mathbf{e}}_\mathbf{i}{}^*$ is the desired motion in the task space, and where $\mathbf{J_i^+}$ is the pseudo-inverse of $\mathbf{J_i}$. The motion is usually constrained to follow a differential equation:

$$\dot{\mathbf{e}}_\mathbf{i}{}^* = -\lambda \mathbf{e_i} \tag{5.5}$$

Thus the control law is:

$$\dot{\mathbf{q}}_i = -\lambda \mathbf{J}_\mathbf{i}^+ \mathbf{e_i} \tag{5.6}$$

We finally get the task Jacobian $\mathbf{J_i}$ according to the robot articular Jacobian $\mathbf{J_q}$, and the interaction matrix:

$$\mathbf{J_i} = \mathbf{L}_{\mathbf{s_i}} \mathbf{M} \mathbf{J_q} \tag{5.7}$$

where $\mathbf{M}$ is the matrix expressing the velocity $\mathbf{v}$ from $\mathbf{J_q}$.

One can remark that according to eq. (5.6) and eq.(5.1) a task is mostly defined by the feature it is handling $(\mathbf{s_i}, \mathbf{s_i^*})$ and its gain $\lambda$. The Jacobian eq.(5.3) is then simply computed from the interaction matrix provided by the feature and the articular Jacobian of the robot.

### 5.1.3 Ensuring the priority

Let $(\mathbf{e_1}, \mathbf{J_1})$ ... $(\mathbf{e_n}, \mathbf{J_n})$ be $n$ tasks. The control law computed from these $n$ tasks should ensure the priority, that is to say the task $\mathbf{e_i}$ should not disturb the task $\mathbf{e_j}$ if $i > j$. A recursive computation of the articular velocity is proposed in [SS91]:

$$\begin{cases} \dot{\mathbf{q}}_\mathbf{0} = 0 \\ \dot{\mathbf{q}}_\mathbf{i} = \dot{\mathbf{q}}_{\mathbf{i-1}} + (\mathbf{J_i} \mathbf{P}_{\mathbf{i-1}}^\mathbf{A})^+ (\dot{\mathbf{e}}_\mathbf{i} - \mathbf{J_i} \dot{\mathbf{q}}_{\mathbf{i-1}}), \quad i = 1..n \end{cases} \tag{5.8}$$

where $\mathbf{P_i^A}$ is the projector onto the null-space of the augmented Jacobian $\mathbf{J_i^A} = (\mathbf{J_1}, \dots \mathbf{J_i})$ and $\widetilde{\mathbf{J}}_\mathbf{i} = \mathbf{J_i} \mathbf{P}_{\mathbf{i-1}}^\mathbf{A}$ is the limited Jacobian of the task $i$ . The robot articular velocity realizing all the tasks in the stack is $\dot{\mathbf{q}} = \dot{\mathbf{q}}_\mathbf{n}$.

### 5.1.4 Ensuring the continuity

From (5.8), the control law is obtained by imposing a reference velocity $\dot{\mathbf{e}}_\mathbf{i}$ for each task in the stack. Generally, an exponential decrease is required by imposing the first order differential equation $\dot{\mathbf{e}}_\mathbf{i} = -\lambda_i \mathbf{e_i}$. However, this equation does not ensure the continuity of the robot velocity when the stack is changed. In [MC04], Mansard has proposed a solution to properly smooth the robot velocity at the transition, by imposing a specific second order equation:

$$\ddot{\mathbf{e}}_\mathbf{i} + (\lambda_i + \mu)\ \dot{\mathbf{e}}_\mathbf{i} + (\lambda_i \mu)\ \mathbf{e_i} = 0 \tag{5.9}$$

where $\lambda_i$ is the gain that tunes the convergence speed of task $\mathbf{e_i}$, and $\mu$ sets the transition smoothness of the global control law. The control law is obtained by introducing (5.9) in (5.8):

$$\begin{cases} \dot{\mathbf{q}}_\mathbf{i} = \dot{\mathbf{q}}_{\mathbf{i-1}} + (\mathbf{J_i} \mathbf{P}_{\mathbf{i-1}}^\mathbf{A})^+ (-\lambda_i \mathbf{e_i} - \mathbf{J_i} \dot{\mathbf{q}}_{\mathbf{i-1}}) \\ \dot{\mathbf{q}} = \dot{\mathbf{q}}_\mathbf{n} + e^{-\mu(t-\tau)} \left( \dot{\mathbf{e}}(\tau) + \Lambda \mathbf{e}(\tau) \right) \end{cases} \tag{5.10}$$

where $\tau$ is the time of the last modification of the stack and $\Lambda$ is diagonal matrix for which each element $\Lambda(i,i) = \lambda_i$.

### 5.1.5 Adding the secondary constraints

The constraints are added using the Gradient Projection Method [Lie77, Kha86]. The constraints are described by a cost function $V$. The gradient $\mathbf{g}(\mathbf{q})$ of this cost function can be considered as an artificial force, pushing the robot away from the undesirable configurations. It is introduced as the last task of the stack. It has thus to be projected onto the null space of each task into the stack. Using (5.10), the complete control law is finally

$$\dot{\mathbf{q}} = \dot{\mathbf{q}}_{\mathbf{n}} + e^{-\mu(t-\tau)} \left( \dot{\mathbf{e}}(\tau) + \Lambda \mathbf{e}(\tau) \right) - \kappa \mathbf{P}_{\mathbf{n}}^{\mathbf{A}} \mathbf{g} \tag{5.11}$$

The reader is invited to refer to [MC04, MC07] for more details.

### 5.1.6 Application to Visual servoing for grasping

The presented experiment is a typical execution of the stack of tasks. An object is placed in the workspace, and is moved randomly. The robot is walking along a planned trajectory that passes close to the object. While walking, the robot has to grasp the object.

The tasks used are the following:

- At each iteration, the pattern generator produces the next reference position that should be reached by the robot. The walking behavior can thus be written as a task function:

  $$\mathbf{e}_{\mathbf{walk}} = \mathbf{q}_{\mathbf{leg}} - \mathbf{q}^*_{\mathbf{leg}} \tag{5.12}$$

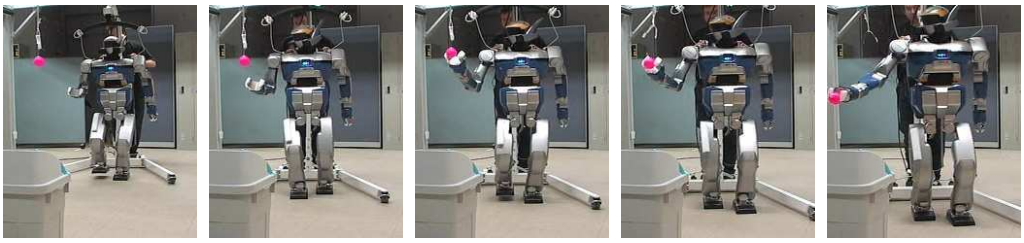
  where $\mathbf{q}_{\mathbf{leg}}$ is the current articular position of the two legs and $\mathbf{q}^*_{\mathbf{leg}}$ is the reference position produced by the pattern generator. The Jacobian $\mathbf{J}_{\mathbf{walk}}$ is very simply:

  $$\mathbf{J}_{\mathbf{walk}} = \begin{bmatrix} \mathbf{I}_{\mathbf{nleg}} & \mathbf{0}_{\mathbf{nleg}} & \mathbf{0}_{\mathbf{n-2nleg}} \\ \mathbf{0}_{\mathbf{nleg}} & \mathbf{I}_{\mathbf{nleg}} & \mathbf{0}_{\mathbf{n-2nleg}} \end{bmatrix} \tag{5.13}$$

  where $n$ is the total number robot joints, and $n_{leg} = 6$ is the number of joints of each leg.

- In order to avoid any disturbance of the walk, in [38] a task blocking the chest has been added. It was replaced later by a CoM task taking as a reference the CoM trajectory generated by the pattern generator. In this specific work however the posture provided by the pattern generator was kept. Later on a different set-up is described.

- In order to ensure the object visibility during the servo, and to stabilize the image motion to improve the image processing, the image of the object is centered in one of the camera view.



(iter 2800) (iter 3600) (iter 3700) (iter 3800) (iter 4200)

Figure 5.1: Experiment B: Key images of the grasping sequence.

(iter 2800) (iter 3600) (iter 3700) (iter 3800) (iter 4200)

Figure 5.2: Experiment B: Key images of the grasping sequence, taken from the embedded (left) camera, and are used during the servo to track the object.

- The grasping task is mainly a 3D-positioning of the right-hand gripper at the object position. However, to ensure that the gripper will be properly oriented when grasping, we have chosen to dissociate the positioning task in two parts. The first part controls the orientation of the gripper, the second part controls the distance to the object: a gripper orientation control and a gripper position control.

The camera feedback runs at $30Hz$. A Kalman filter is used to synchronize the two process loops. A high level controller implements two rules to avoid neck joint limits avoidance and grasping only when the robot is close enough. The rules goals are keeping the object to track in sight and avoiding neck joint limits. This is realized by removing and adding the task maintaining the chest still.

A brief overview of the experiment is given in the two last figures. Fig. 5.1 gives an overview of the robot motion taken from an external camera. Fig. 5.2 gives the corresponding snapshots taken from one of the embedded stereo camera. A version where the object is put on a flexible pole has been demonstrated several times.

## 5.2 Collision detection: real-time smooth gradient

### 5.2.1 Approach

Here we address the problem of on-line collision avoidance (including auto-collision avoidance). Although several closed-loop controllers have been proposed for various reactive task purpose full body motion, see for instance [NYK+05][SK06], surprisingly

none of them guarantee, in an explicit way, non desirable collisions or auto-collisions. Adrien Escande [Esc08] proposed an efficient algorithm to compute fast proximity distances. We integrated it in a low-level reactive control. The fast proximity distance algorithm uses strict-convexity bounding volumes close to polyhedral convex hulls in order to guarantee the continuity of the proximity distance gradient and in the same time satisfying real-time control requirement. In [PSRA05] authors make use of cylinders and spheres to cover a redundant manipulator and compute in a efficient way proximity distance to be used as a secondary task in a kinematics task prioritization scheme and experimented on a 7 dof redundant manipulator. Instability that may occur from discontinuity of the witness points has been addressed in configurations where the cylinders are nearly parallel. The proposed algorithm makes use of patches of spheres and toruses to cover in a more precise way the convex hull of robot's part and get rid of this problem.

This algorithm has been integrated as an additional task in the stack of the task sequencing architecture described above. In humanoid robotics, a similar idea has been proposed in [SGJG06] where self-collision avoidance has been implemented on the simulator of ASIMO, their method makes use of an artificial force which changes the value of the desired posture in the gradient of a posture cost function projected in the null space of the main tasks; tasks are described in derivative of joints or Cartesian spaces (velocity). This virtual force penalizes a close proximity distance obtained from sphere sweeping along the humanoid skeleton lines [LGLM90]. It creates simply a convex bounding volume of the humanoid. Their approach is very elegant and simple to implement, yet they did not consider stability problems that may occur from discontinuous witness points and their formalism would not allow marking auto-collision avoidance with higher priority. This is possible with our method.

### 5.2.2 A new strictly convex bounding volume

Proximity distance and more generally collision avoidance have been widely studied, which results in numerous algorithms and schemes (the interested reader may refer to the recent, exhaustive and excellent review books [van04][Eri05]). However, little attention has been paid to the continuity properties of the proximity distance. It has been pointed out however that in singular cases, the gradient of this distance is not continuous, generating oscillations or misbehaviors in the control scheme [PSRA05]. For a complete discussion on the continuity problem as well as a method to regularize the proximity distance gradient, see [Esc08]. The main idea is to build strictly convex hulls of the robot bodies in a way that can be seen as a slight blowing up of the usual convex hull. It is realized through patches of spheres and toruses. Such a volume is called Sphere-Torus-Patches Bounding Volume (STP-BV). Its advantages are (i) to ensure continuity of the gradient, and (ii) to accurately approximate the convex hull of the object (in typical cases we experienced, the volume increase from convex hull to STP-BV is 1–2%), while maintaining the number of collision pairs low (one volume per body) as well as the computation time for each of them.

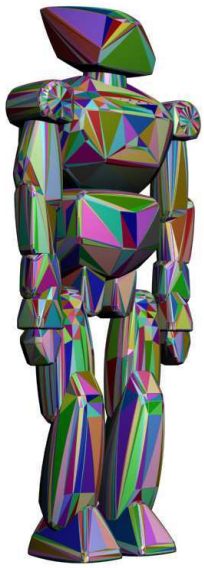


Figure 5.3: STP-BV representation of HRP-2.

One of the most interesting property of STP-BV main result presented is it is sufficient to have only one strictly convex body to have a continuous minimum distance between two witness points of two convex bodies. This means that the STP-BV construction is not mandatory for obstacles to hold the continuity. It is sufficient to build the patches only for the robot's bodies. The HRP-2's associated representation is depicted in Fig. 5.3. Details on the computation of the distance gradient and the cost function associated to the tasks used for avoiding obstacle can be found in [34].

### 5.2.3 Reactive obstacle avoidance using the SoT

Building upon the example of catching a ball while walking, the collision avoidance task is introduced in the stack of tasks just after walking. Two more modifications have been realized to implement obstacle avoidance. Firstly a set of 116 pairs are chosen to track auto-collision. Bodies linked to each other are not included because they are treated by the joint limits constraints. The second modification is the inclusion of a 25 cm diameter ball in the collision pairs. The ball position is put in such way that the robot's hand comes into collision, The value starting the inclusion of the distance inside the control law is set to $a_i = 0.1$ m.

The robot avoids the ball in two occasions. First by holding the hand when moving forward. Then the avoidance stops while the robot is moving away. In second when the robot swing back to the left for the next step the hand and the arm moves to the right to avoid collision as depicted in Fig.5.4. As in [38] the high-level controller removes the orientation and grasp tasks right after grasping, and a task extending the right arm is put inside the stack. The CoM task then uses the left arm to maintain

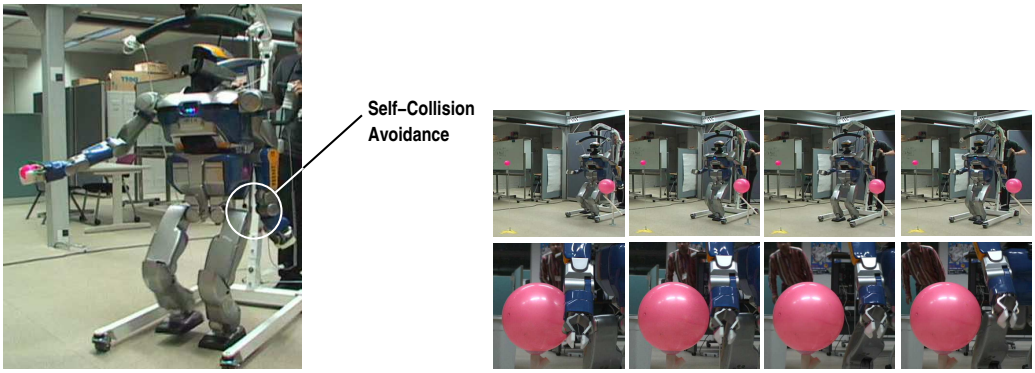



Figure 5.4: (left) Avoiding Self-Collision (right) Avoiding an obstacle, here the biggest pink ball.

the CoM given by the pattern generator. However this motion makes the robot reach its joint limits. The left arm then moves toward the left leg, and avoid it thanks to self-collision pairs in the controller, see Fig. 5.4.

## 5.3 Application to Human-Humanoid robot interaction

This work was realized in the context of the project ROBOT@CWE. The aims is to have a human and a humanoid robot collaboratively working together to perform a task. In parallel to the work described in section 4 we implemented the walking pattern generator proposed in [MHK+07]. Indeed chronogically, the work realized in the frame of ROBOT@CWE was done before the results presented in chapter.4. The overall control architecture is depicted in Fig. 5.5. The system interact with two different users. One is in direct interaction with the robot through the force sensors mounted in the robot wrists. The other user is far away and interact with the robot through an haptic interface. The robot behaves passively using an admittance controller, while stepping. The robot decides its next foot step location by trying to maintain a neutral position between its feet and its wrists. The CoM trajectory is recomputed in real-time as well as the foot steps trajectories. The following section describes in more details the experiments involving a direct human-humanoid robot interaction.

### 5.3.1 Experiments

As the pattern generator used in this experience is strictly limited to the LIPM, the stack of tasks used in this context is different from the previously described experiments. Instead of having a task following the articular values generated by the pattern generator, two tasks are used. One realizes a relative rigid trajectories between both feet. A second one realizes the CoM expressed in the frame of the foot

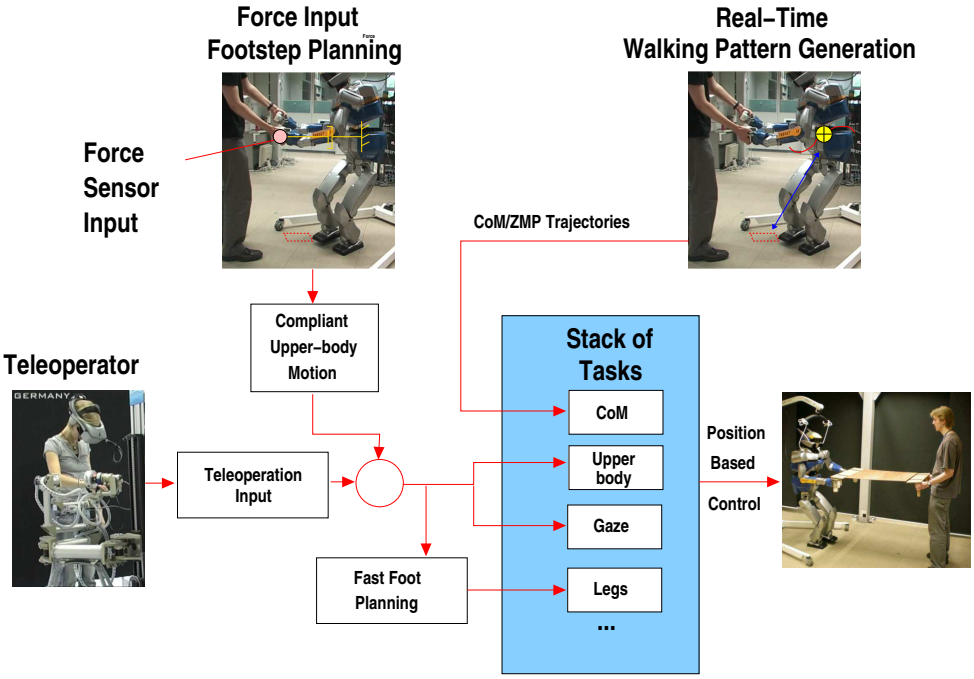



Figure 5.5: Overview of the control architecture in the frame of the final experiment of the ROBOT@CWE project

in contact with the ground.

We have tested two different experimental contexts for this direct manipulation. The first one constrained the motion in the sagital plane. In this setup there are no sideways walking, only frontal direction. Therefore as the kinematic constraints are becoming less activated, it is possible to use an interval of 20 cm in the sagital direction. The robot is quite reactive, and it was possible for several people to test the system with no particular training as depicted in the left part of Fig.5.6. A video of the corresponding experiments can be found at [88]. The second context released the constraint on the sagital plan, and allow 3d foot motion: $(x, y, \theta)$. It is the one depicted in Fig.5.6-(a-d). The sub-figures (a) and (b) depicts the robot turning with the human operator. Sub-figure (c) demonstrate a side-way walk, and (d) a backward walk. These motions can be seen in the companion video of this paper at [88]. At the end the last segment presents a preliminary result on holding an object. Compare to the work presented in [YHI+03a] and for which a video is available at [YHI+03b] it appears clearly that our work generates whole body motion, when [YHI+03a] is mostly keeping its upper body still. Unfortunately the lack of information in [YHI+03a] forbids to perform further comparisons.

Fig.5.7 and Fig.5.8 depict the ZMP reference along the X-axis computed by the pattern generator using Morisawa's method[MHK+07]. The real ZMP is also represented. One can see the deformation induced by the change of the swinging foot position when this one starts its flying phase. This occurs when the time-shift is

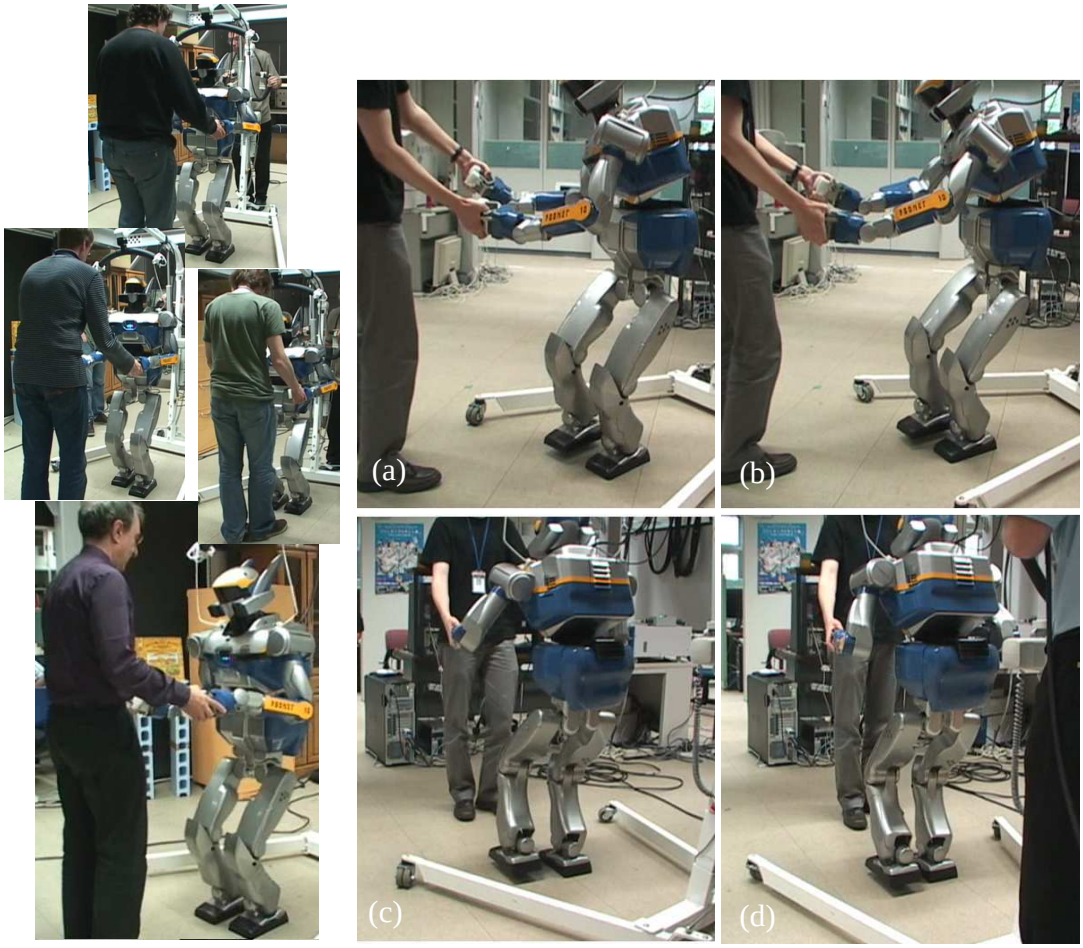



Figure 5.6: Experimental results: (left) several persons interacting in real-time with HRP-2 using the force sensors in the wrist. For safety the steps of the robot are constrained along the sagital plane. (right) Whole-body motion with steps going in all the directions. Self-collision is avoided thanks to the work described in [21]

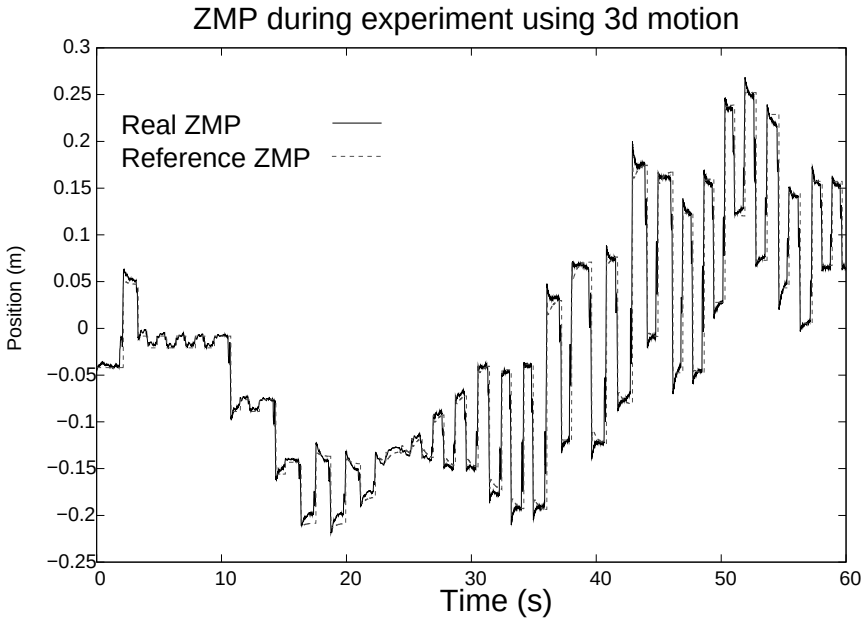



Figure 5.7: Reference and real ZMP

applied according to the Y-axis selected because it has the biggest ZMP reference perturbation. The real ZMP is deviating 2 cm away from the reference trajectory which is quite similar to the result obtain by [NK09]. Most of our current fall are due to the constraints of the robot forbidding to execute fully the tasks of the system. When going backward the robot reach more quickly its chest joint limit, and therefore in this direction the CoM task is more difficult to realize.

Although there is one SVD decomposition to be computed for each task, this control scheme fits into the 5 ms control loop of the HRP-2 as can be seen in Fig.5.9. The peaks correspond to the computation of new steps. It takes less than 1 ms to compute the new weights of the ZMP and CoM trajectories. Between these peaks the pattern generator merely computes polynomials.

A video of the experiment realized in the frame of the FP6 ROBOT@CWE European project can be found at the following address
http://www.youtube.com/watch?feature=player_embedded&v=hRf0XJaIlmY.

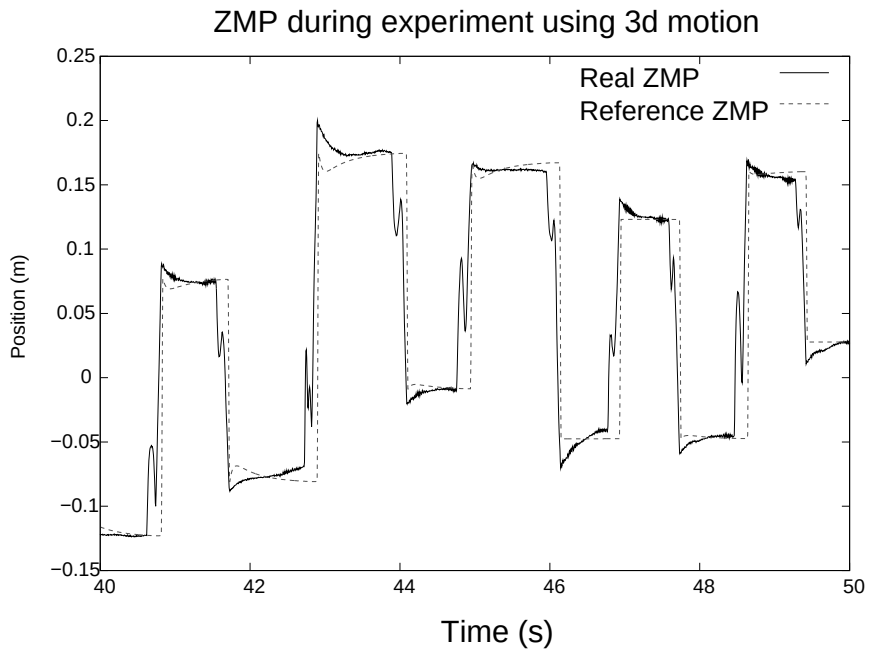



Figure 5.8: Zoom of Fig.5.7

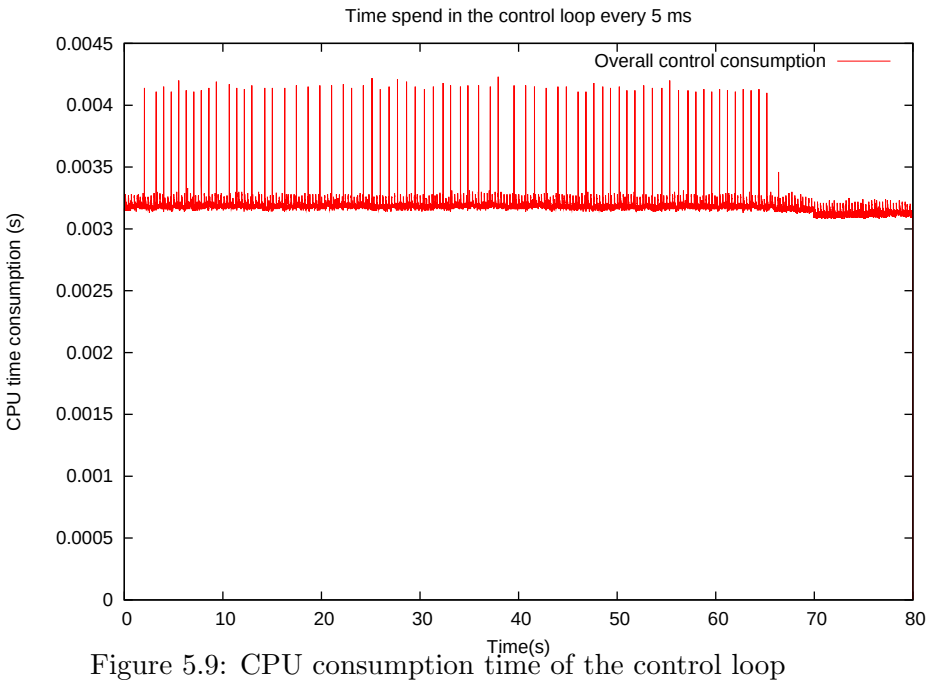



Figure 5.9: CPU consumption time of the control loop

# Chapter 6

# Angular momentum and foot step planning

When a humanoid robot is walking and performs whole body motion, the legs and the arms create an angular momentum which acts upon the robot CoM. This effect can be neglected, as done in chapter 4, with robots for which the mass distribution is centered around the CoM walking at a nominal speed. This is the case for instance for the humanoid robot HRP-2. However when one recomputes the center-of-pressure induced by the motion of the robot, there is still a 2 *cm* difference between the most stable trajectory and the one really performed by the robot. For some motions this may make the robot falling. Taking into account angular momentum is even more necessary for robots with heavy legs and arms (such as Johnny [BLB+07]).

We have introduced in chapter 3 the dynamic filter which modifies the CoM trajectory to compensate the effect of the limbs inertia. This chapter describe the use of this dynamic filter to plan limbs motion and modifies the CoM trajectory. At the end we even described a method which integrates this aspect at the planning level, and allow to realize real-time foot-step planning. As the Generalized Inverted Kinematics scheme described in the previous chapter might be quite costly, a key ingredient here is to be able to compute very quickly the inverse kinematics of the legs. In this chapter we assume a rigid relationship between the CoM and the waist of HRP-2. When the walking pattern generator provides the position orientation of the CoM, as well as the position and orientation of the feet, 12 constraints are generated. As HRP-2 has 12 degrees of freedom for the legs, it is possible to solve analytically the inverted kinematics. From this, the second preview controller is used extensively to compensate for constraint imposed upon stability: namely the passive joint and collision obstacle avoidance with a large object. In both case we build upon the preview control and the dynamic filter to enhance the capabilities of HRP-2.

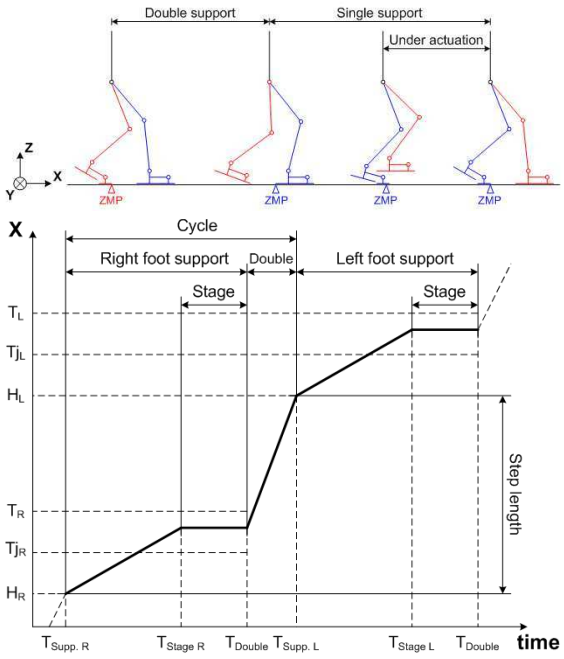



Figure 6.1: (left) Walking Gait With Toed Feet (right) ZMP trajectory for toed feet gait.

## 6.1 Simulation: toe-joint

### 6.1.1 Goal

In [43], we have simulated a modified HRP-2 humanoid robot equipped with the passive toe-joints depicted in Fig.6.2 (left). By considering an under-actuated phase during walking and an appropriate ZMP trajectory it was possible to increase by 1.5 times the step-length of HRP-2, and have smoother CoM trajectories.

### 6.1.2 ZMP trajectory during single support phase with a toe joint

During the single support phase, the ZMP moves linearly from the heel (H) to the toes (T) (Fig.6.1 (right)). At the end of this phase (from $T_{Stage}$ to $T_{Double}$), the ZMP is kept in a constant position between the toe joint ($T_j$) and the tip of the toe ($T$), which makes the whole body rotate around this joint. During the double support phase, starting with the swing foot touch down ($T_{Double}$), the ZMP is transferred from the toes of the support foot to the heel of the swing one.

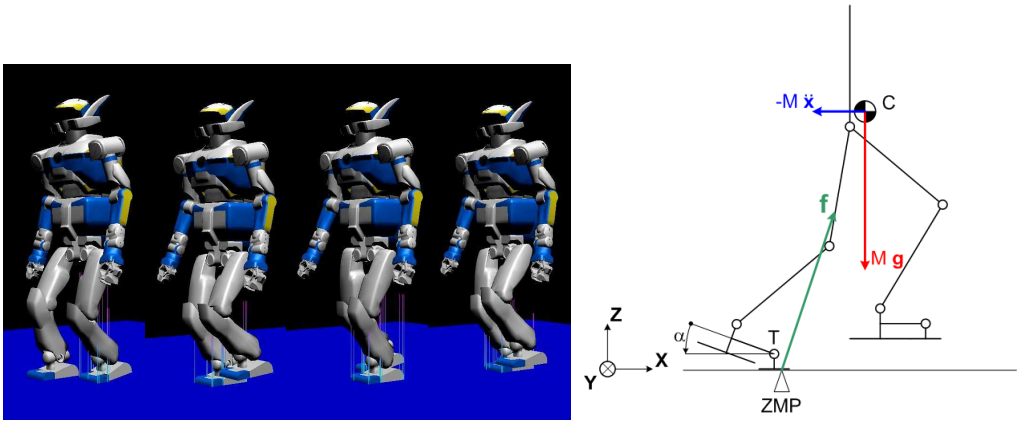



Figure 6.2: (left) Dynamical simulation of the HRP-2 model with Toe Joint ($K_p$ = 10Nm/rad $K_d$ = 5Nms/rad) (right) Force and toe joint angle.

### 6.1.3 Under-actuated phase

During the flat feet walking gait, the feet are kept parallel to the ground. However, with toed feet, a rotation around $T_j$ occurs during the single support phase. In order to know what $T_j$'s angle is, a model of the spring related to the ground reaction force is used. The toe joint is orthogonal to the sagital plan (X, Z). Its angular position depends on the dynamical effects in this plan. During single support phase, the forces acting on the CoM are gravitational and forward accelerations. A simple model was used in order to compute the toe joint angle as depicted in Fig.6.2 (right). At the end of this phase, the ZMP is located under the toe, which can be considered as stable on the ground ( slipping phenomena are not considered at this stage). This leads to the following equations:

$$\mathbf{f} = -m(\mathbf{g} - \ddot{\mathbf{c}}_x) \tag{6.1}$$

$$\tau_{toe} = -\mathbf{p}_{toe} \times \mathbf{f} \tag{6.2}$$

$$\tau_{toe} = -(K_p\alpha + K_d\dot{\alpha})\mathbf{Y} \tag{6.3}$$

with $\mathbf{f}$ the ground reaction force, $m$ the mass of the robot, $\ddot{\mathbf{c}}_x$ the CoM acceleration along the X axis, $\tau_{toe}$ the toe joint's torque, $\mathbf{p}_{toe}$ the position vector of ZMP with respect to the toe joint, $\alpha$ the toe joint angle, $\mathbf{Y}$ the position of the toe in the foot reference frame, and finally $K_p$ and $K_d$ toe joint spring and damper constants. Given the CoM acceleration along the X axis, the toe joint's torque is computed using eq.(6.1) and eq.(6.2). Then, the angular position is derived from the "spring-damper" model at each simulation step from eq.(6.3).

### 6.1.4 Contribution

The merit of this work was to show that it is possible to increase the step-length, and to decrease the variation of the knee joints using passive joints. Interestingly

Kajita [KKM+07] uses a similar technique to make his prototype hoping. The main difference in his approach is to additionally consider the energy restitution during the spring release phase. The simulation reported by Kajita shows the robot able to run at 3km/h.

## 6.2 Dynamically stepping over obstacles

### 6.2.1 Goal

This study proposes a complete solution to make the humanoid robot HRP-2 dynamically step over large obstacles. As compared to previous results using quasi-static stability [GYT05] where the robot crosses over a 15 cm obstacle in 40 s, our solution allows HRP-2 to step over the same obstacle in 4 s. This approach allows the robot to clear obstacles as high as 21% of the robot's leg length (15 cm) while walking. Simulations show the possibility to step over an obstacle that is 35% of the length(25 cm) with a margin of 3 cm.

### 6.2.2 Approach

The input of the proposed method is an evaluation of the obstacle position and size provided by vision for instance. First the algorithm acts as a dynamical planner. It uses the fact that the double support phase is quite short in dynamical walking. During this period the CoM motion is negligeable. The satisfaction problem $MCSP_{\mathbf{u}}$ during this phase is then reduce to find a key configuration similar to that of Fig.6.3(left). Thus a feasible trajectory exists, i.e. satisfying $MCSP_{\mathbf{q}}$, if a key configuration fulfilling the following criteria exists:

- the knee angle is beyond a minimum angle to avoid over-stretching,
- there is no collision between the legs and the obstacle and,
- there is a kinematic solution for a chosen height of the waist.

This key configuration is found by a greedy search along the waist height. Starting from a standard height it is lowered until the previous condition are satisfied.

Because the motion generator used in this algorithm assumes that the waist and the CoM are rigidly fixed, once the waist height is found then the CoM height is automatically fixed for the double phase of the stepping over. Let us briefly explain why it is possible to lower the height of the CoM inside the preview control method despite the initial assumption of the LIPM stating that $c_z$ is constant. If we consider

the model of the Inverted Pendulum one can write:

$$\begin{aligned} p_x &= \frac{mgc_x - m(c_z\ddot{c}_x - c_x\ddot{c}_z)}{mg + m\ddot{c}_z} \\ &= \frac{mc_x(g - \ddot{c}_z) - mc_z\ddot{c}_x}{m(g + \ddot{c}_z)} \end{aligned} \tag{6.4}$$

If $\ddot{c}_z \ll g$ then we have the Linearized Inverted Pendulum:

$$p_x = c_x - \frac{c_z}{g}\ddot{c}_x \tag{6.5}$$

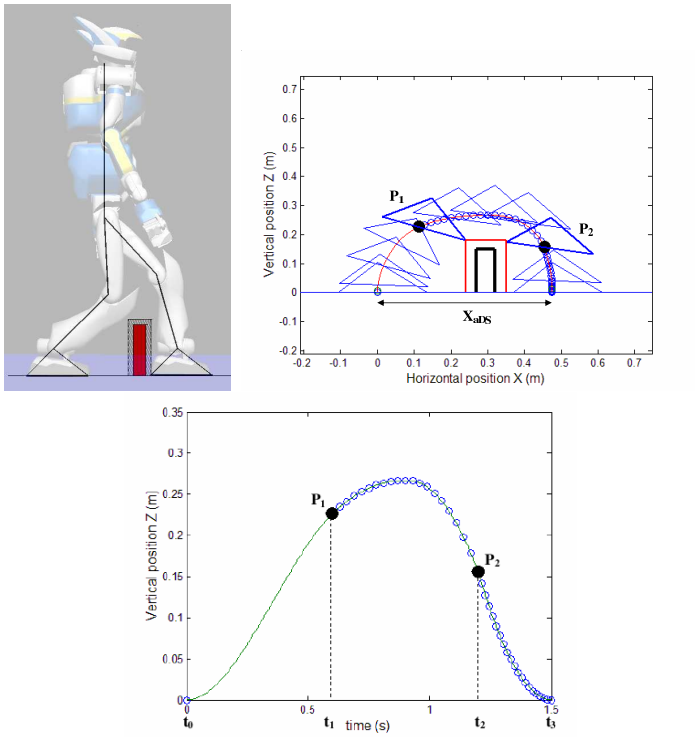



Figure 6.3: (left) Double support phase feasibility (middle) Spline Trajectory in the XZ plane (right) Spline Trajectory to time for vertical direction Z

### 6.2.3 Foot Trajectory

The foot trajectory for this application is different from more classical polynomial trajectories because the latter ones tend to oscillate in the presence of control points. Instead, Clamped Cubic Splines are used to specify intermediate velocities. This is needed to specify control points $\mathbf{P}_1$ and $\mathbf{P}_2$ to avoid the obstacle, and to minimize impacts upon touch-down. Indeed, for such moments Clamped Cubic spline functions ensure zero velocity, but they do not specify what the acceleration should be. In real experimentation, the desired trajectory cannot be perfectly tracked due to the

compliance in the foot. Clamped Cubic splines allow smoothing of the touch-down condition.

For normal walking the step length is 0.23m, while stepping over requires a step length of 0.48m. The size of the step being almost doubled, its duration is increased. The step sequence lasts 0.78s in single support and 0.02s in double support for the normal steps. Stepping over lasts 1.5s and the preceding and the succeeding steps 0.04s.

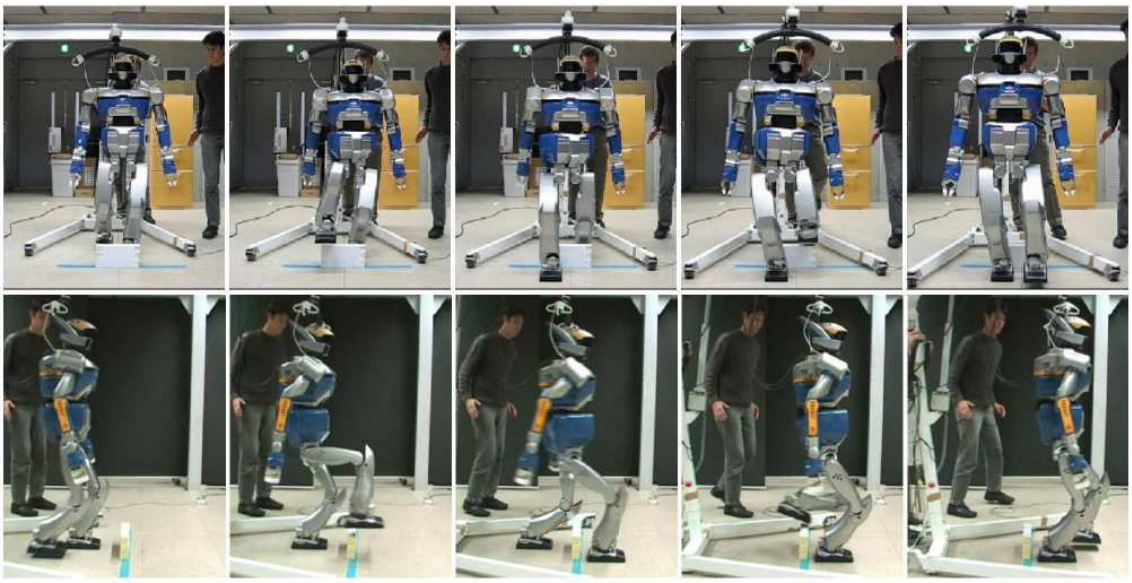


Figure 6.4: Stepping over experiments with HRP-2.

### 6.2.4 Upper body motion

To clear more space during the double support over the obstacle and consequently allow for larger obstacles to be stepped over, the waist of the robot is rotated. The HRP-2 robot includes two extra degrees of freedom (yaw and pitch) between the waist and upper-body so the upper-body and head remain oriented towards the walking direction.

### 6.2.5 Experimental results

The results of stepping over an 15-cm-high and 5-cm-wide obstacle (with a 3 cm safety boundary and 2x3 cm safety boundary, respectively) are depicted in Fig.6.5, Fig.6.6 and Fig.6.4. The figures show the desired *ZMP* and waist position for both the walking direction ($X$) and the perpendicular horizontal direction ( $Y$) (showing 7 steps). A normal step takes 0.78 s for single support and 0.02 s for double support,

while the stepping over step and both previous and subsequent steps take 1.5 s and 0.04 s respectively. The stability of the system is given by the position of the *ZMP*, which is calculated using the complete multi-body model of the robot.

The bottom graphs in Fig.6.6 show both the *ZMP* calculations after the first and second preview controllers. The first preview is clearly jerky and different from the desired *ZMP*, specifically for the stepping over. However the dynamic filter compensates completely for the use of the simplified model, the disturbances of the large swing leg motions and the waist height variation during the stepping over. It is particularly important in this direction because this is where the largest perturbations occur, while the support area is the shortest.

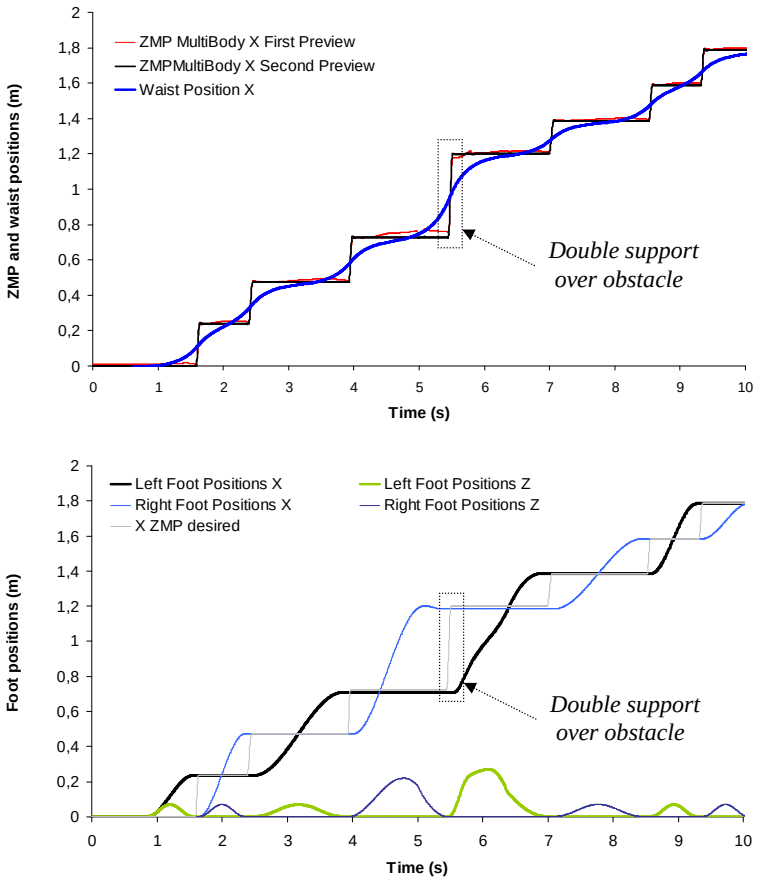



Figure 6.5: *ZMP* and waist position in the walking direction($X$), including horizontal and vertical foot positions, for stepping over a 15-cm-high and 5-cm-wide (plus 3 cm safety boundary and plus $2 \times 3$ cm safety boundary, respectively)

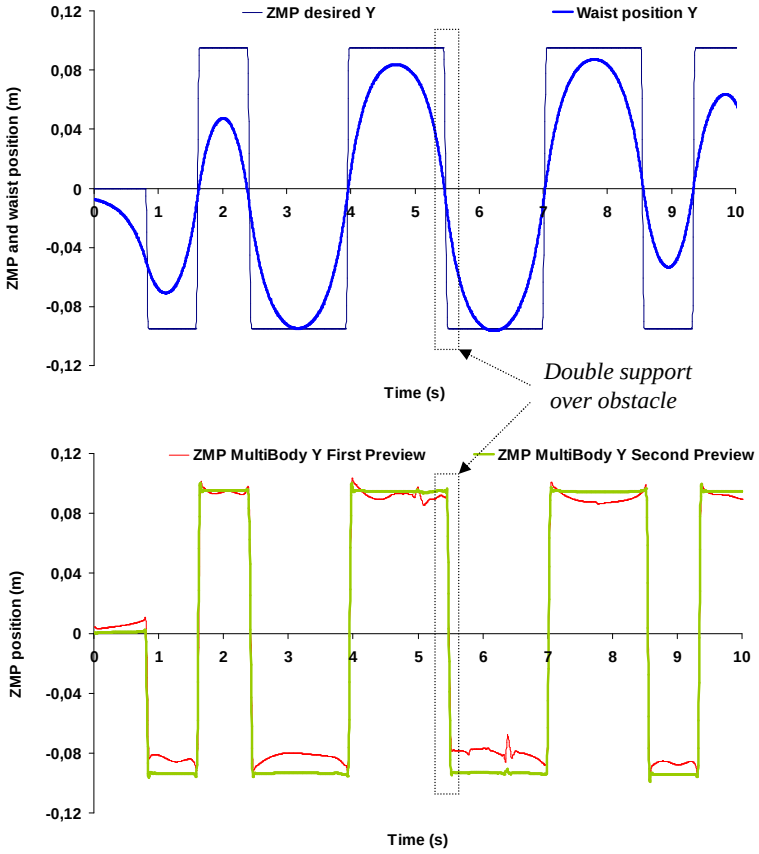



Figure 6.6: *ZMP* and waist position of the perpendicular horizontal ($Y$) direction for stepping over an obstacle of 15-cm-high and 5-cm-wide (plus 3 cm safety boundary and plus $2 \times 3$ cm safety boundary, respectively)

## 6.3 Planning foot steps and upper body motion

This section describes experiments using the new degree of freedom corresponding to the CoM height. The robot is facing an opening 2 meters ahead with an other wall (beyond the gate) at 4 meters (cf Fig.6.7(up-left)). The goal is to cross the gate while manipulating a 2 m long bar held in the right hand. The environment was reconstructed by using the embedded stereoscopic vision system using previous work [49]. In short, several dense maps were accumulated to give the map depicted in Fig.6.7 (bottom). The result is processed to simplify the geometry, and used as an entry to a general 3D motion planner called KineoWorks[Lau06]. Kineoworks searched and optimized a path by exploring the configuration space of a HRP-2 reduced model depicted in Fig.6.7 (bottom). In the reduced model the legs are regarded as their equivalent bounding box with three degrees of freedom: two in the horizontal plane, and one small translation at the top of the box for the height modification. The upper torso is kept complete. The output is a set of points $(x, y, z_{tr}, \theta, \mathbf{q}_{torso})$ also called key-points. Between each key-points it is possible to linearly interpolate the

position, the orientation and the angular values of the upper torso to realize a motion without collision from a pure geometric viewpoint. The articular values $\mathbf{q}_{torso}$ are used as direct input of the motion generator, whereas $x, y, \theta$ are used to generate steps. Indeed the upper body motion influences the ZMP, consequently the CoM trajectory has to be adapted accordingly. The bounding box corresponding to the legs ensures that these steps can be realized. More precisely, from the steps the pattern generator generates a CoM trajectory. As the waist is supposed to be here rigidly fixed with the CoM, a waist trajectory is associated but it is different from the solution by Kineoworks. However it is guaranteed to be included into the bounding box moving along the found trajectory. In order to synchronize at best the key-points on the real trajectory of the robot, the key-points are remapped on the closest points along the waist (root of the robot model) trajectory. In the present experiment, the Kineoworks optimization was **not** done taking into consideration the dynamics of the robot-bar system. It was assumed that the pattern generator is able to compensate for the variation introduced by the upper body. Of course, this hypothesis is not valid for any motion and any robot-object dynamic system. The successful experiment here was achieved with a 100 g bar, whereas a 1 kg bar happen to induces too much momentum and make the robot fall. However the range of acceptance of the system is wide, and when this hypothesis is valid it provides a faster solution. It should be noted that as this work is in its early stage, the planning phase has been done off-line. The result is a set of configuration for the upper body according to the position of the waist. The corresponding file is then loaded inside the pattern generator, and the trajectory of the foot and the waist are generated on-line as for the previous experience.

## 6.4 Fast foot-steps planning

### 6.4.1 Problem statement

Let us assume that the robot is using a specific *g*ait *g*enerator

$$gg : \mathbf{x} \rightarrow \gamma \tag{6.6}$$

with $\mathbf{x} = [\mathbf{x}_{lf}^{\top}\mathbf{x}_{rf}^{\top}]^{\top}$, and $\gamma$ is an articular trajectory $\mathbf{q}(t), \forall t \in [t_{begin}, t_{end}]$. A feasible trajectory should not make the robot self-collide, respect the articular limits and have the Center-Of-Pressure projected into the support polygon. Based on these constraints, let us consider the following function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ such that

$$f(\mathbf{x}) \begin{cases} > 0 \text{ if } \gamma \text{ is feasible,} \\ \leq 0 \text{ if } \gamma \text{ is not feasible.} \end{cases} \tag{6.7}$$

Such function can be computed from $\gamma$ by calculating the distance to the above constraints. However the time taken to generate $\gamma$ and compute the distance to the constraints can be computationally expensive. In the context of probabilistic roadmap planning, the speed taking to shoot a configuration in the foot-steps space and

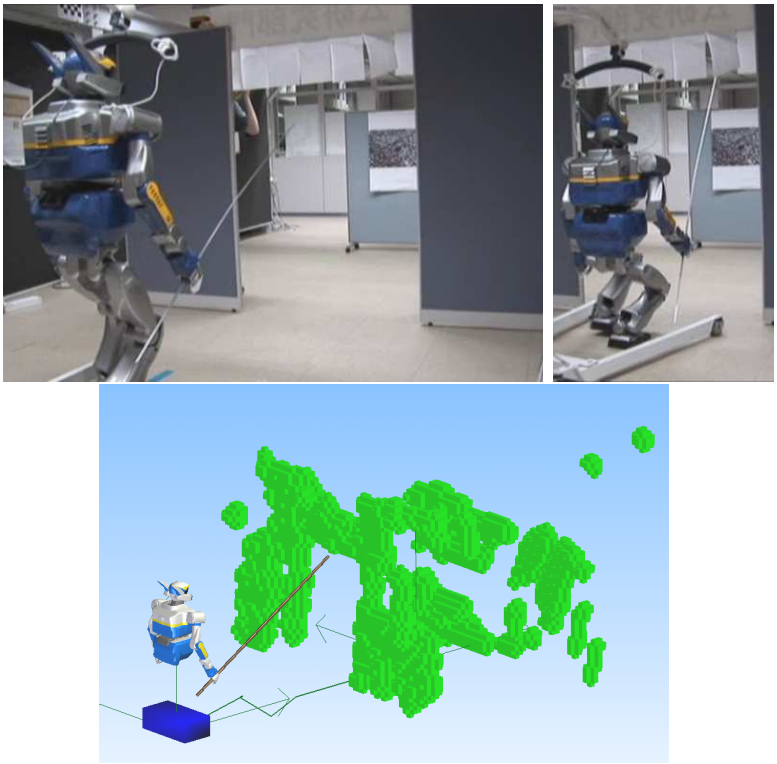


Figure 6.7: (top) HRP-2 crouching to clear a door while manipulating a 2 m bar to avoid the wall beyond the gate. (bottom) Reconstructed environment inside KineoWorks with a simplified model

checking either or not this configuration is feasible is important. It affects the resolution efficiency. Therefore our approach is to sample $f$ and build an approximation $\hat{f}$ which can be evaluated very efficiently.

### 6.4.2 Overview of related work

The current solution is to only allow a small set of steps for the robot: in that case the generation and verification phases are useless since all trajectories can be memorized and verified off-line ([KNK+03, CKNK03]). This approach is not always satisfying for it leads to a gait which has no flexibility, and combined with planning it often results in the robot making a large number of steps to perform a task for which only one or two steps would have been arguably enough.

Pre-computing robot dependent data-structure has been proposed in path planning for multi-body robots in the past [LH01, KM04, NL08]. In these papers a road-map is computed for a multi-body robot without obstacles. Once the robot is placed in an environment with obstacles, the pre-computed road-map is pruned by removing edges in collision with the obstacles. The remaining road-map is then used

to plan paths.

Closer to our application, in [TTIA07] a 2 dimensional map is built which returns the time necessary to change a HRP-2 step-length during the flying phase of the foot in order to realize an emergency stop. The key-point of this work is to build a map which verifies that the ZMP realized by the robot stays in the support polygon for a given step-length modification done at a given time while walking. Indeed walking pattern generator such as the one proposed by Kajita et al. [KKK+02a], or Morisawa [MHK+07] does not guarantee that the robot desired ZMP will stay in the support polygon. The main difference between this previous work and our approach is that we consider more constraints, and propose an adaptive partition of the input space well suited for higher dimensions. Indeed our work, taking into account free steps ([TTIA07] only considers forward walking), has to aim at dealing with higher dimensions.

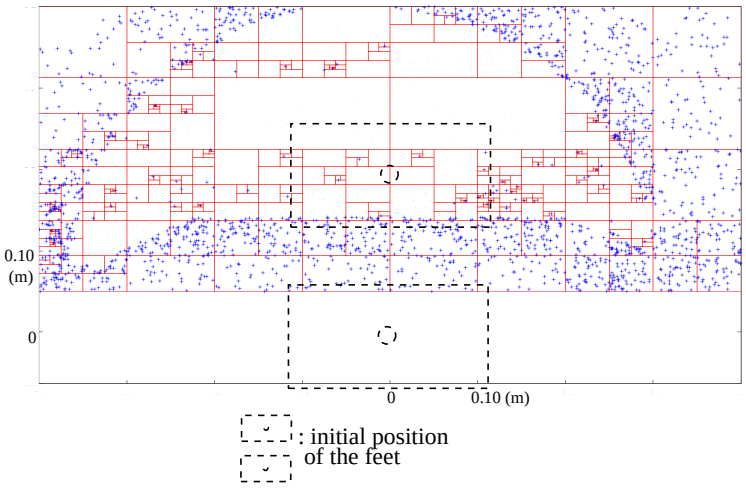



Figure 6.8: The construction of the approximation $\hat{f}$

### 6.4.3 Adaptive sampling

In order to cope with the high-dimension aspect of the function the method builds an approximation function which is not over-complete, but based on a tree representation of the input space. Inside the leaf-boxes, an optimization problem is solved to provide a local approximation similar to the Support Vector Regression method.

The leaf-boxes are separated in two categories:

- Boxes including positive and negative output values are called “frontier boxes” noted $\mathbb{B}_{Frontier}$. Empty boxes also belong to the frontier.

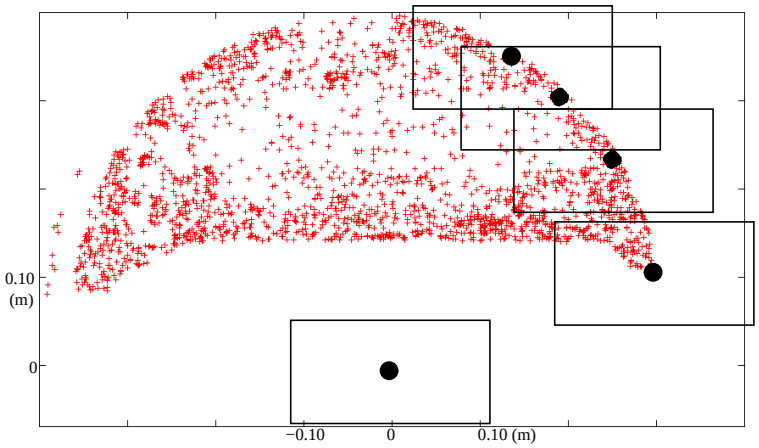



Figure 6.9: The positive samples generated and some feasible steps

- Boxes including only negative or positives values are called "regular boxes" noted $\mathbb{B}_{Regular}$.

Without further assumption on $f$ it is quite difficult to decide when the approximation at the frontier is sufficiently well sampled, or when the regular boxes do not contain areas where $f$ is of opposite sign. Therefore we assume that there is a probability $\rho_{boost}$ that the sampling should be perform on $\mathbb{B}_{frontier}$ rather than $\mathbb{B}_{regular}$. This probability can be changed by the user to modify the behavior of the algorithm. Once the set from which to choose has been decided, the box with the lowest confidence is used to generate the sample.

### 6.4.4 Computing $f$

In this specific application $f$ is computed by starting from the set of steps described in Fig.6.10. By setting $\theta = \theta' = 0$ the dimension of the input space can be reduced to 4. The pattern generator used in this specific work is the one described in [KKK+03a]. The implementation used is described in [9]. It generates a trajectory $\gamma$ in the articular space. It has the particularity to add a constraint between the waist and the CoM making sure that for one configuration of steps leads to one and only one $\gamma$ trajectory. V-Clip was used to compute self-collision between the legs, while the Center-Of-Pressure is used to compute the degree of stability of the robot. The distance to the constraints is computed at each time step of the trajectories: joint limits, self-collision and deviation of the fictuous ZMP from the desired trajectory. $f$ is finally the smallest value of the distance to the constraints over the all trajectory. A major difficulty for the approximation scheme is to reflect the non-linearities introduce by having constraints both on the task space and on the articular values.

Locally the approximation scheme is using a similar representation to the Support

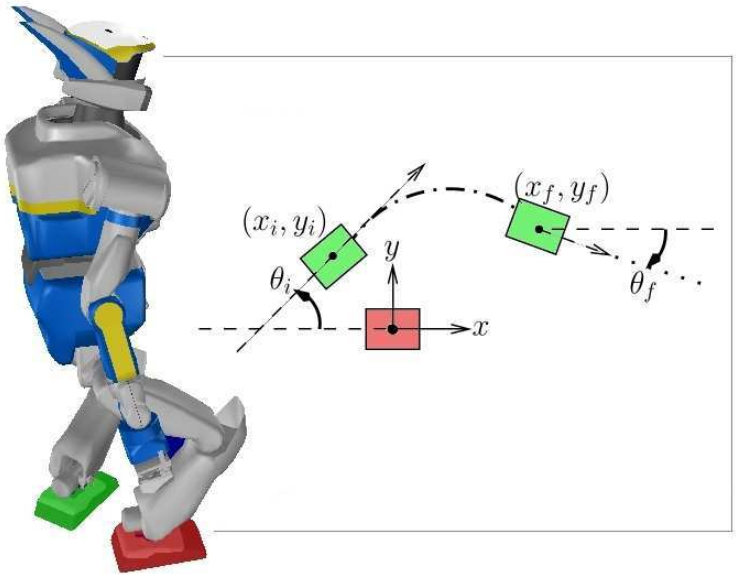



Figure 6.10: Parameterization of a step for humanoid robot HRP-2. Left foot is set at the origin of the coordinate frame, right foot moves from $(x_i, y_i, \theta_i)$ to $(x_f, y_f, \theta_f)$

Vector Regression scheme [SS04b] without feature space.

### 6.4.5 Experimental results

Fig.6.8 shows the mechanism of the decision tree which recursively divides the input space into a disjoint union of rectangular cells. It also shows the negative samples (unfeasible steps); we can see some negative samples in what one would picture as the feasible area: this is due to numerical unstability in the computation of $f$, more specifically when a possible trajectory is close by a constraint. These slow the computation as it creates a new frontier. Fig.6.9 shows the positive samples. They are concentrated near the frontier between feasible and unfeasible arrival foot-steps, which is indeed the region on which the approximation scheme should focus. Video [88] demonstrates a walking sequence where a user drive the robot with a joystick. The joystick change continuously the destination of the next flying phase until this phases begins. This solution is filtered out by the approximation function. This overall approach however do not apply to the volume swept by the robot. Therefore in [4] we have use this result to describe a dense set of discrete actions (250) between half-steps in quasi-static balance. To speed up the trajectory found between such quasi-static poses, Nicolas Perrin proposed to deform locally the trajectory. In [16] we have demonstrated how this technique can plan 3D foot-steps trajectory in a dynamic environment.

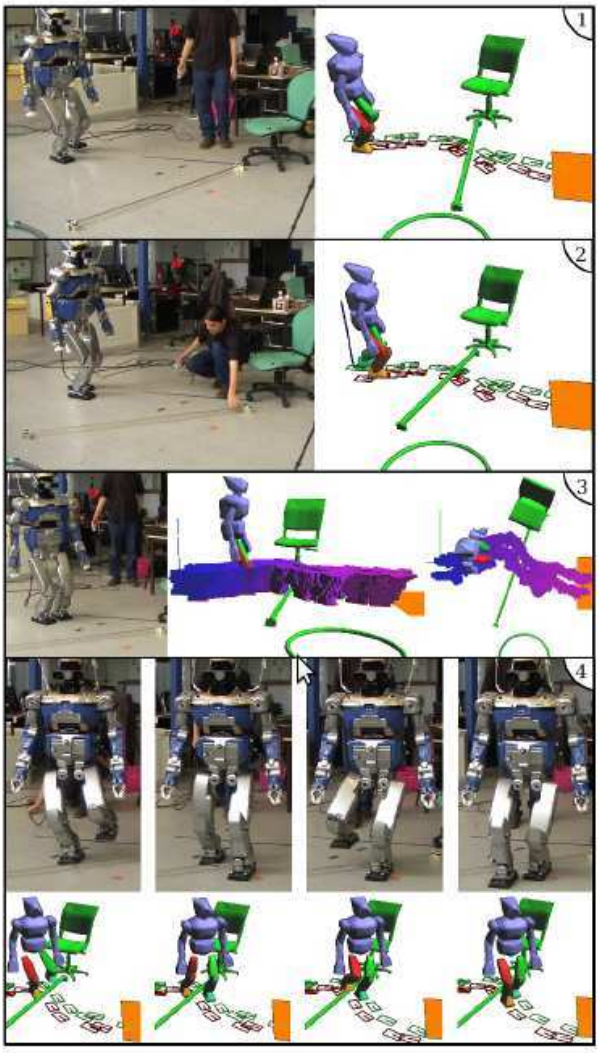



Figure 6.11: Results of the foot-step planner presented in [4]

# Part II

# Vision for humanoids

# Chapter 7

# Introduction

This chapter focuses on the acquisition of visual information on the environment to realize the treasure hunting behavior with a humanoid robot. It is split in 4 parts that we have treated separately:

- A computer vision based representation of an object dealing with recognition;
- Simultaneous Localization And Map building (SLAM);
- autonomous model building;
- autonomous visual search.

## 7.1 Object representation

The visual model used to realize this behavior consists of:

- SIFT features [Low04] enhanced with their 3D position;
- 3D surfaces based on density maps;
- Bags of features and a Markov model relating the geometrical relationship of the bag of features.

Fig.7.1 shows an example of the representation used in this work. There is a long history of work regarding the problem of object recognition and its localization. Current most advanced system in humanoid robots uses bio-inspired feature representation [WSG+07], or a combination of visual features such as edges, color and textures [OKT+07]. On this specific area my contributions lie on two topics: triangulation and object recognition for far away objects.

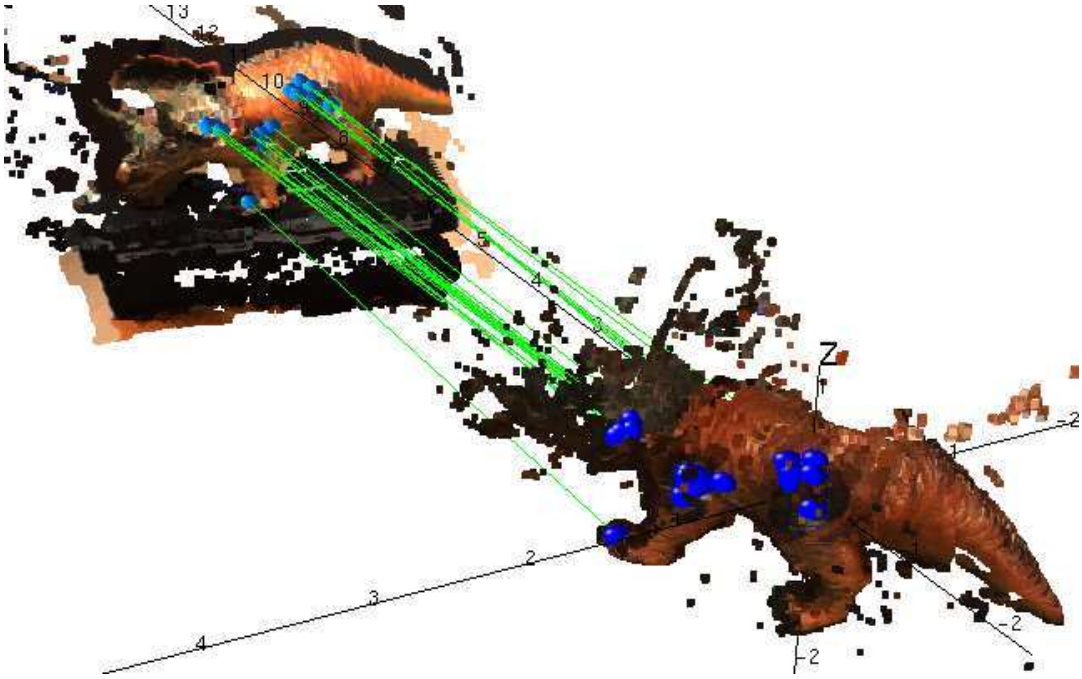


Figure 7.1: Example of the object model: SIFTs with their 3D position and associated density maps. This example shows the recognition and the localization of this object.

**Triangulation and Interval Analysis** With Benoît Telle, we have worked on the problem of error representation for triangulation with a stereoscopic system. A vast majority of the work on the subject relies on the probabilistic framework [Kan08]. The novelty of the approach presented here is to consider the pixel as an interval, and use the results of Interval Analysis to deal with the problem of triangulation [61, 60, 57]. This offers guaranteed results assuming that the model provides correct boundaries.

**Object recognition** Performing object recognition nowadays is realized with features such as SIFT[Low04] or SURF[BETG08]. However those scale invariant descriptors have some limitations when looking far away. With Diane Larlus we have explored a solution relying on generative model based on bag of features. With this method, HRP-2 is able to detect a toy 6 meters far away. This is better compare to 2 meters as the maximum detection distance with previous methods [41].

## 7.2 Simultaneous-Localization And Map-building

Having a representation of the environment, and knowing its localization in this representation is necessary for the robot to take high-level decisions such as planning the next sensing position. According to the requirements of the behavior different representations might be necessary. In this work we have mostly considered 3D rep-

resentation of the environment. Thus, because most of current state-of-the art SLAM system use particle filtering on maps, they are quite computationally intensive when dealing with 3D representations. For this reason, when dealing with low time period behaviors[1] such as visual search, we have used the classical 3D grid based representation proposed by Movarec [Mor88], and still used in more recent work in the field [AHW+11]. When higher frequency are required real-time method use sparse representation.

With Andrew Davison, we have tried to address the problem of monocular Simultaneous-Localization and Map Building on a humanoid robot. The idea was to try to build a sparse map of landmarks in 3D for the robot to self-localize. Indeed this is necessary to correct position of the robot when walking inside an environment. After integrating the gyroscope information, limitation on the height variation and information of the pattern generator, we have shown that this could run inside the HRP-2 humanoid robot in real-time[10][44].

## 7.3 Autonomous model building

With Torea Foissotte and Pierre-Brice Wieber we have developed a novel algorithm to plan the next pose of the HRP-2 for building the visual model of an object. This algorithm maximizes the new information while considering the overall constraints of the robot: self-collision, stability and articular limits. It relies on NEWUAO an algorithm developed by Powell which can build a quadratic approximation of a function through measurements [28, 25]. Using an OpenGL implementation of the camera model, it is possible to obtain those measurements very quickly.

## 7.4 Autonomous visual search

For visual search, François Saïdi and myself proposed a spatial based accumulator called the visibility map. It relies on a simple model of the recognition system behavior, and take into account some limitations of the robot. In addition we have demonstrated the implementation of a system exploring the environment and several refinements allowing a fast implementation of visual search on HRP-2 [13, 40, 39, 37].

[1] typically every minute

# Chapter 8

# Bounded error model for vision-based robotic applications

## 8.1 Introduction

This section reformulates the 3D triangulation problem into the framework of Interval Analysis. This provides a warranty on the space where the reconstructed 3D point might lie. Such result is specially useful for safety issue namely robots interacting with humans.

### 8.1.1 Problem statement

We consider a stereoscopic rig for which the two cameras $C_l$ and $C_r$ are modeled by two projective matrix $\mathbf{P}_l$ and $\mathbf{P}_r$. A 3D point noted $\mathbf{X}$ is projected by those two matrices on two pixels $\mathbf{x}_l$ and $\mathbf{x}_r$. Considering that $\mathbf{x}_r$ and $\mathbf{x}_l$ are *measured*, and thus noisy, we want to obtain a bounded estimation of $\{\mathbf{X}_s\}$ the space where lies $\mathbf{X}$. $\{\mathbf{X}_s\}$ represents the position uncertainty of $\mathbf{X}$. The matching between $\mathbf{x}_l$ and $\mathbf{x}_r$ is assumed to be given by an adequate segmentation process. It is assumed that the cameras are calibrated beforehand through a proper process as the one described in [HZ03b] for instance.

Therefore we count three sources of error for 3D triangulation: the segmentation process, the chosen parameters in the camera model and the digitalization noise due to pixel position description. Taking into account such errors is done through an additive term to the pixels position.

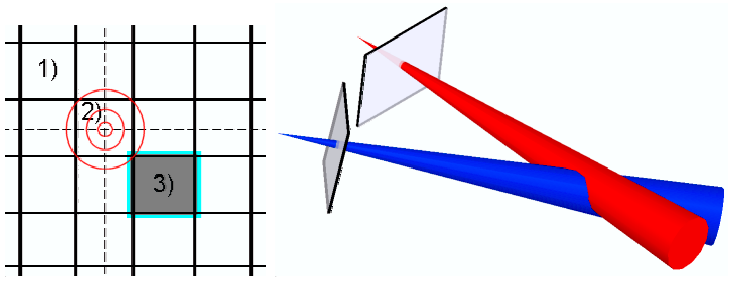



Figure 8.1: (a) 1) real pixel 2) probabilistic model of pixel 3) interval representation of pixel by an interval. (b) Uncertainty or inaccuracy of 3D reconstruction is based on computing the intersection of two cones

### 8.1.2 Contributions

The 3D triangulation problem can also be solved using Interval Analysis as proposed in [TAR03] and [Tel03]. This approach provides an upper bounded evaluation of the error and does not need a sufficient number of samples to be statistically representative. An application using this approach is to build partial boundaries representation of objects as we have proposed in [61]. Moreover we proposed a solution to the so-called *wrapping effect*[Moo66]. Indeed Interval Analysis tends to provides over-pessimistic solution. This is solved by proposing a tighter approximation of this error using a paving representation. As the latter representation is not very convenient to handle, we proposed a way to build an ellipsoid in which the paving is included.

## 8.2 Interval Based Model

We present here an original way to describe a pixel by the use of an interval (Fig. 8.1-(a)3). We distinguish the *inaccuracy* of a point position in the image which is introduced as a random variable and the *uncertainty* of this position introduced by an unknown but bounded variable. This unknown but bounded variable is described by a vector of intervals $[\epsilon]$. The projection is now written:

$$[\mathbf{x}] = Round\left(\frac{\mathbf{PX}}{\mathbf{P}_3^t\mathbf{X}}\right) + [\epsilon] \tag{8.1}$$

Where *Round* is the round operator which provides the nearest integer of a value. We take into account the data normalization in the image: $P_3$ is the third column of the camera model $P$. This allows us to fix the scale factor and to define the value of the error vector: $[\epsilon] = \begin{pmatrix} [\epsilon_l] & [\epsilon_r] & 0 \end{pmatrix}^t$. According to the model, there is no error on the scale factor, but only an uncertainty on the position of the point in the image plane. In a first and empiric approximation, we have to choose a value for the uncertainty ($[\epsilon]$), based on the digitalization noise, which permits to cover the surface of a pixel:

$$[\epsilon] = \begin{pmatrix} [-0.5; +0.5] & [-0.5; +0.5] & 0 \end{pmatrix}^t \tag{8.2}$$

If we consider the set of points described by the vector $[\mathbf{x}]$, they described a planar surface. This is equivalent to an affine model for the camera. Indeed, since we normalise the whole set of points included in $[\mathbf{x}]$ with a unique scale factor ($\mathbf{P}_3^t\mathbf{X}$), this means that the set of points $[\mathbf{x}]$ is the projection of a set of coplanar points $\{\mathbf{X}\}$. This hypothesis characterizes an affine functioning of the camera. Nevertheless, in our case, we do not use intervals in order to describe a set of points $[\mathbf{x}]$ but only one point ($\mathbf{x}$) with an unknown but bounded position: $\mathbf{x} \in [\mathbf{x}]$. Thus, the only hypothesis which are made with this description is based on the uncertainty of data according to the model: the uncertainty is bounded, especially if the image point is in the image plane.

According to these two different kinds of error modeling namely uncertainty and inaccuracy, we aim now to compute the volume of uncertainty or inaccuracy for the reconstruction of a 3D point. The geometrical approach consists in computing the intersection of two cones which are centered on the origin of the camera frame and whose basis section in the image plane are the uncertainty ellipses or intervals associated to each pixel (Fig. 8.1-(b)). The tool proposed by the probabilistic approach is the covariance propagation [Har96]. It provides an estimation of the reconstruction inaccuracy. The first approach proposed by interval analysis are contractors and provide reconstruction uncertainty [Tel03].

## 8.3 3D-set triangulation problem

### 8.3.1 The equivalent uncertain linear system

Classically the relationship between a 3D point and its projections provides the overdetermined system given by equation (8.3). They are 6 equations and 4 unknown data in homogeneous coordinates. The system represents the intersection of two lines in space.

$$\begin{pmatrix} \mathbf{q}_l \\ \mathbf{q}_r \end{pmatrix} = \begin{pmatrix} \mathbf{P}_l \\ \mathbf{P}_r \end{pmatrix} \mathbf{X}_h \tag{8.3}$$

If we avoid the degenerate case, it is easy to invert this linear system. The resolution of equation (8.3) is equivalent to the resolution of a full rank squared system in the form:

$$\mathbf{A}\mathbf{X}_{nh} - \mathbf{B} = 0 \tag{8.4}$$

Where $\mathbf{A}$ and $\mathbf{B}$ are respectively a $(3 \times 3)$ matrix and a 3 components vector. They are built with the elements of $\mathbf{P}_l, \mathbf{P}_r, \mathbf{x}_l$ and $\mathbf{x}_2$. $\mathbf{X}_{nh}$ is the non homogeneous vector of the 3D point in the scene.

Let us now describe position of two matched pixels in a stereoscopic pair of image with the intervals $([\mathbf{x}_l], [\mathbf{x}_r])$. As these intervals can have any size, the main idea is to choose them such that they contain the errors of our system: digitalization noise, and localization uncertainty due to segmentation. Uncertainty on each pixel can also

be different, for instance, it can be grown so as to ensure the epipolar constraint (in our context of calibrated system, this constraint is usually verified).

From previous remarks, it is now possible to write the system associated to 3D reconstruction with interval arithmetics rules. The matrix $P$ associated to a camera model is divided such as:

$$\mathbf{P} = (\mathbf{M} \mid \mathbf{V}) \tag{8.5}$$

Where $\mathbf{M}$ is a $(3 \times 3)$ matrix and $\mathbf{V}$ is a $(3 \times 1)$ vector. The operator $[*]_\times$ is the anti-symmetrical matrix associate to the cross product function. The system given by equation (8.4) is written with the interval matrix $[\mathbf{A}]$ and the interval vector $[\mathbf{B}]$ as follows:

$$[\mathbf{A}] = \begin{pmatrix} [[\mathbf{x}_l]]_\times \mathbf{M}_l \\ [[\mathbf{x}_r]]_\times \mathbf{M}_r \end{pmatrix}; [\mathbf{B}] = \begin{pmatrix} -[[\mathbf{x}_l]]_\times \mathbf{V}_l \\ -[[\mathbf{x}_r]]_\times \mathbf{V}_r \end{pmatrix} \tag{8.6}$$

Section 8.2 shows that linear mappings are optimal inclusion functions. Then, this description of the problem is exact. It exists several interpretations of its solution [Tel03]. We will deal with the exact set of 3D points $\{\mathbf{X}_s\}$ which is solution of the uncertain linear system

$$[\mathbf{A}]\,\mathbf{X}_{nh} - [\mathbf{B}] \doteq 0 \tag{8.7}$$

And the sign $\doteq$ expresses the solution in the following terms:

$$\{\mathbf{X}_s\} = \left\{\mathbf{X}_{nh} \in \mathbf{R}^3 | \exists \mathbf{A} \in [\mathbf{A}], \exists \mathbf{B} \in [\mathbf{B}], \mathbf{A}\mathbf{X}_{nh} - \mathbf{B} = 0\right\} \tag{8.8}$$

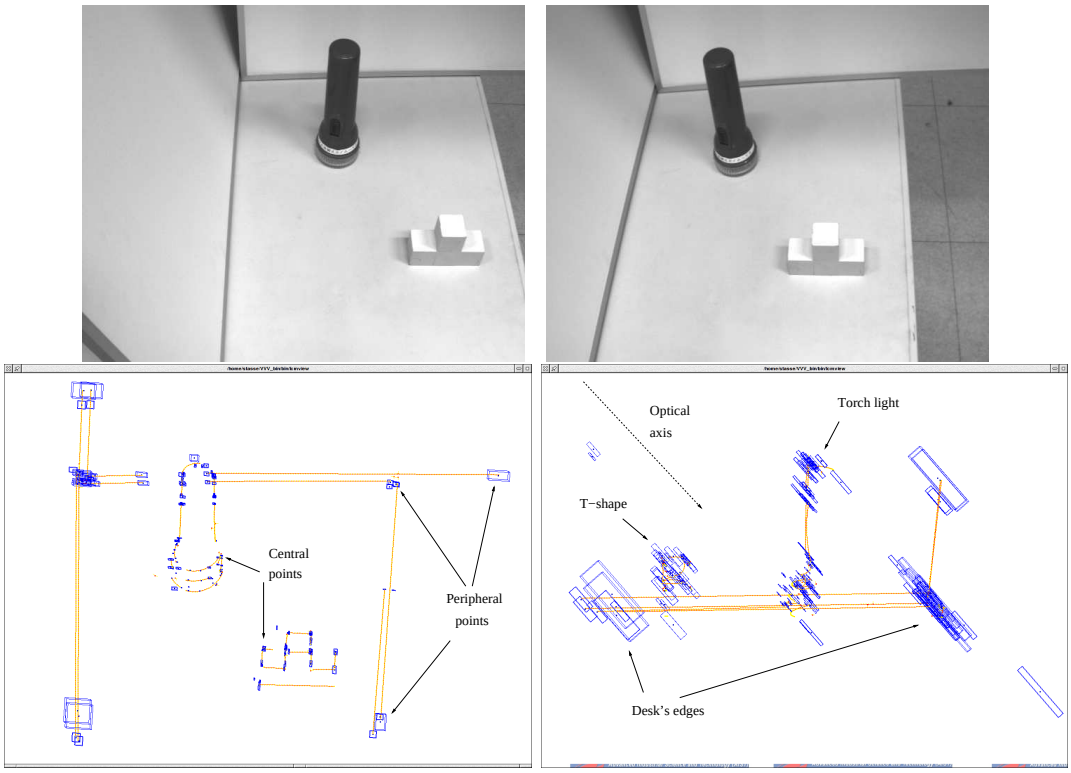
By the way of interval analysis, we aim to find the minimal external bounding box $[\{\mathbf{X}_s\}]$ of $\{\mathbf{X}_s\}$.

### 8.3.2 Over determined system

It is known that lines 3 and 6 of matrix $\mathbf{A} \in [\mathbf{A}]$ are linearly dependant with the other lines. Extracting those two lines allows to reduce the system to dimension $(4 \times 3)$. Then 4 square systems $(3 \times 3)$ can be obtained. Since we propose to produce a solution as an external bounding box for each system, the intersection of solutions provided by each of them is still an external bounding box. For this reason, the next developments are proposed with four square matrices $[\mathbf{A}_k], k = 1...4$ extracted from $[\mathbf{A}]$. The method based on contractors will be applied to each system. At last, we will keep only the intersection of all the results:

$$[\{\mathbf{X}_s\}] \subset \bigcap_{k=1...4} [\{\mathbf{X} \mid [\mathbf{A}_k]\mathbf{X} - [\mathbf{B}_k] \doteq 0\}] \tag{8.9}$$

Others methods which use contractors in the case of over-determined systems are presented in [Tel03][Rum83]. One of our contribution is to show through experimental results presented in [61] that the intersection of square systems lead to better results. Practically, using the Gauss Siedel contractor with 9 iterations gives the best precision with a reasonable time cost (0.002 second for a point).




(a) (b)

Figure 8.2: (up) Original images for 3D reconstruction. 2 kinds of objects are considered: Geometrical form and free form. (bottom) Different viewpoints of the 3D reconstruction with Boundaries representation and interval bounding boxes. Only digitalization noise is taken into account.

The 3D triangulation has been applied to the scene represented in Fig. 8.2. Matched points and camera models are provided by the Versatile Volumetric Vision System developed by the 3D Vision Group of the Intelligent System Research Institute (AIST Japan). This versatile system presented in [KKK+02b] uses a trinocular stereo camera setup which is able to reconstruct 3D information of a scene, to recognize an object [SKYT98], and to track a recognized object [SIT02]. Two kinds of objects allowing a good segmentation process have been tested on this scene: a T-shape object and a torch-light. A view of the 3 images obtained by the VVV system are displayed in Fig. 8.2. Three viewpoints of the reconstructed scene are given in Fig.8.2. The uncertainty which have been taken into account in this experimentation is only digitalization noise ($[\epsilon] = ([-0.5, 0.5][-0.5, 0.5][0^-, 0^+])^t$). We can observe the boxes representing the intervals in which the reconstructed points lie according to the contractor resolution of 3D reconstruction problem. As we could expect, deeper objects, or objects far away from the optical axis are less accurately located. It can be seen in Fig. 8.2.c and this is particularly clear for the edges of the desk.

Fig. 8.3 gives some examples of depth maps reconstructed using Gauss Siedel.

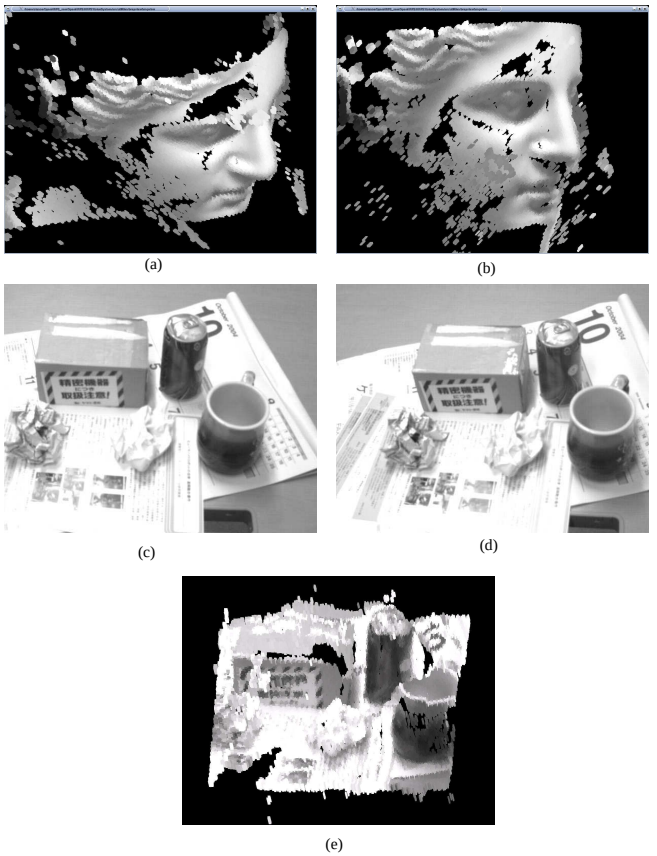



Figure 8.3: Depth Map of the Milo Venus (a-b) using Interval Analysis. The uncertainty is represented using boxes. (e) shows a scene reconstructed with the two images depicted in (c-d).

## 8.4 More accuracy with Set Inversion

The previous approach provides a bounding box of the exact set $\{\mathbf{X}_S\}$. However considering equation (8.8), and the fact that we are using a bounding box, we find a hull for the exact set $\{\mathbf{X}_s\}$ which does not fit perfectly with the real shape of $\{\mathbf{X}_S\}$. In other words, several points do not fullfill the property stated by equation (8.4). This problem is known as the wrapping effect [Moo66]. The geometry of a box does not map the true shape of $\{\mathbf{X}_s\}$. Moreover, boxes used in interval analysis are not oriented and then depend the world frame's orientation. In this section a new approximation based on paving is proposed. It builds a better approximation which still bounds the real solution $\{\mathbf{X}_s\}$.

We propose here to consider the projection in a set inversion problem. Indeed, we can say that reconstruction consists in searching the set of 3D points $\{\mathbf{X}_s\}$ the projection of which in each camera $\mathbf{P}_i$ $(i = 1...n)$ is the image $[\mathbf{x}_i]$ $(i = 1...n)$. By using the model of camera we proposed (8.1):

$$\mathbf{X} \in \{\mathbf{X}_s\} \quad \Rightarrow \quad \left\{ Round\left(\frac{\mathbf{P}_i\mathbf{X}}{\mathbf{P}_{i3}^T\mathbf{X}}\right) + [\epsilon] \right\}_{i=1...n} \in [\mathbf{x}_i] \tag{8.10}$$

Unlike the previous formulation of the 3D reconstruction problem, which solves an uncertain linear system, we consider the problem of set inversion related to this projection. It consists in retrieving the domain of definition of a given function when the image domain is known. This is done through the algorithm SIVIA developed by Jaulin [JKDW01]. In [60] an inclusion function build through a mechanism similar to voxel carving is shown to provide a more compact, but still bounded, representation of the space we are looking for. The result is depicted in Fig. 8.4. Although this representation is more precise, its memory cost is more important and it is more complex to manipulate than intervals. Therefore we proposed to build an bounding ellipsoid around this tree which guarantees the inclusion of the reconstructed space. For further details, the reader is invited to read [60].

## 8.5 Conclusion

Interval analysis applied to computer vision has already been proposed for stereoscopic vision [MN98], and object recognition [Bre03a, Bre03b]. In [MN98] the problem considers only two parallel cameras. In [FBFB04] Fusiello et al. propose to use a global convergent interval analysis technique to solve the problem of auto-calibration of a moving camera with unknown constant intrinsic parameters. However, the work presented focus only on the auto-calibration problem. The main difference highlighted in this document lies in the formulation of the problem and the study regarding the contractors. First, Fusieollo and al. in [FBB04] do not reformulate the problem explicitly

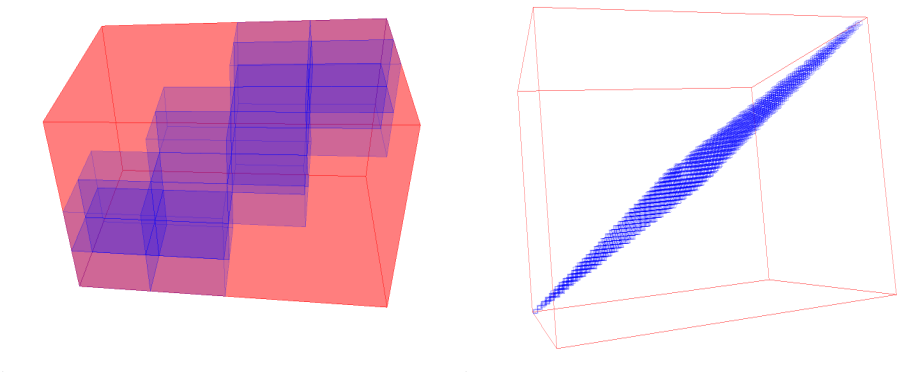


Figure 8.4: Application of SIVIA using octree on the same point but with different depth. In the left image, $depth = 2$, in the right image $depth = 5$.

in the form given by equation(8.1). Secondly in [FBB04] the over-determined case is treated the same way as proposed in [TAR03] (by considering solving $A^T Ax = A^T b$), but the second method is not considered, i.e. to intersect the three linear system which can be extracted from the over-determined one.

Being able to find guarantee on the quality of 3D triangulation is a first step towards quality assessment of robot services. We believe this is the first time that this problem has been cast into a proper set theory representation. We have presented an implementation of contractors and obtained minimal bounding box of 3D reconstruction. This computation is realized with a calibrated stereoscopic system and a pair of matched points. Experimental results show the robustness of the method. Our method is based on a new description of a pixel. Using intervals in uncertain homogeneous variable is a robust and flexible way to deal with image data. The only assumption is the bounding aspect of the uncertainty. This assumption ensures the flexibility of this description. It allows to consider any kind of uncertainties for the variables. It can handle uncertainties of camera parameters, uncertainties of positions of matched points and uncertainties produced by segmentation processes. The accuracy of interval based results runs into the drawback of wrapping effect. This problem has been solved by the use of paving algorithm which has been presented. It enhances the accuracy of the computed uncertainty but the point description is more complex. Based on this work, bounding ellipsoids were introduced.

# Chapter 9

# Visual object models: automatic construction and visual search

This section deals with the problem of object representation for visual search by a humanoid robot. A classical model representation is described in details in [41]. It is based on SIFTs for which the 3d relative positions are reconstruct using HRP-2 stereoscopic system. For an object of 20 cm, the current limitation to find the object and its pose in a scene is 2 meters. An important problem for a robot to search for an object is to have possible candidates even when the locations are over the limits. Two approaches are proposed to handle this problem. One consists in considering the limits of the recognition system and plan the sensor location accordingly. Such strategy is developed in section 12. A second approach consists in trying to develop an image based method to detect possible candidates. This work is the result of a fruitfull collaboration with Diane Larlus. We described the method in the following.

### 9.0.1 Overview of related work

**Object learning and recognition on Humanoids** Fitzpatrick et al. [FMN+03] proposed to use active perception to have a humanoid robot learn a visual representation of an object. By light tapping, the humanoid robot Cog segments the boundaries of the object, and builds an orientation based representation. This representation is invariant by scale and orientation; however it is not intended to handle 3D representation, and mostly targeted for closed-range interaction. Ude et al. [UGC04] proposed to use a classification on a Gabor filter representation of objects. The classification is realized by a SVM classifier. It is also associated with detectors such as colors to elect candidates in the environment. Ude et al. uses the wide view cameras of the DB humanoid robot to elect possible candidates, and brings them in the foveated images. Extending on this work, Welke et al. [WOU+06] uses a feed-forward neural network to learn the feature representations of an object model. There is however no autonomous mechanism to learn the visual features of an object as for Fitzpatrick.

Of particular interest is the work of Taylor and Kleeman [TK06], where visual object models are build using a range laser stripe, and fit into geometric primitives; their robot was able to build a high-level representation of the scene through a graph, and then perform grasping in a cluttered environment.

**Object detection** Object detection has receive a lot of attention during the last decades. Most of the methods use sliding windows techniques. A classifier is trained in order to decide if a region contains an object or not and is applied at all possible positions and scales. We can cite among others, the cascade of classifiers proposed by Viola and Jones [VJ01]. These methods are known to be efficient but need a huge number of training images (few thousands) and require a very long training stage.

We decided to focus on a generative model based on local representation. Models considering images as collection of small patches have been used first for image retrieval and more recently for object classification. A popular technique is to use a quantification of these local representations into so called visual words (first introduced by [CDF+04]). These techniques can also be applied to the localization of objects in the image. They are particularly adapted when dealing with strong variations of object appearances and occlusions.

## 9.1 Seeing far away: a generative model based approach

As shown in [41], the sift-based reconstruction method fails at detecting objects far away. The method presented in this section aims at providing consistent of the object position and scale in the robot field of view. It can detect object in challenging conditions, such as difficult viewpoints, small scale, extreme illumination conditions and occlusions. This hypothesis can be used as an input for the visual search when the 3D object reconstruction fails. It is an extension of the method of [LJJ09] and uses additional information coming from the robot to guide the model estimation process. In particular we will use both the left and right images of the robot cameras to compute dense disparity maps and then use the resulting depth information as an extra component of the model.

### 9.1.1 Visual Features

Images are represented by a set of $n$ overlapping patches and a gradient map (see Fig. 9.1).

**Overlapping visual patches.** Patches, denoted $\mathcal{P}_i, i \in \{1, \ldots, n\}$, are sets of pixels belonging to square image regions. Five different characteristics are computed from each patch.

First of all, a visual codebook is obtained by $k$-means clustering SIFT [Low04]

based representations of the patches. Then, each patch $\mathcal{P}_i$ is associated to the closest codeword. The assigned codeword is denoted $w_i^{sift}$; this is the first characteristic. We also produce visual words based on color information by clustering color descriptors [vdWCS06]. The patch $\mathcal{P}_i$ is also characterized by its closest color codebook word $w_i^{color}$. A RGB value is computed by averaging over pixels extracted in the center of the patch. This 3D-vector is denoted $rgb_i$. We also consider the coordinates of the patch center $X_i = (x_i, y_i)$ in the image. Finally, the dense disparity map provides an estimation of the depth $d_i$ of the patch.

**Gradient Map.** In addition to this patch based characteristics computation, we also extract a gradient map $\mathcal{G}(x, y)$, that consists of the strength of the gradient at each $(x, y)$ pixel location.

In the end, the gradient map $\mathcal{G}(x, y)$ and the characteristics of the $n$ overlapping patches $\mathcal{P}_i$: $\{w_i^{sift}, w_i^{color}, rgb_i, X_i, d_i\}$, $i \in \{1 \cdots n\}$ compose all the information we use to describe an image.

### 9.1.2 Model description

The strength of our model lies in the combination of two (different but) complementary components: (i) a blob based generative model using visual words for its good object localization properties, and (ii) a MRF (Markov Random Field) structure which provides a coherent field of labels following object boundaries.

#### A blob-based generative model

We consider that an image is made of "blobs", and that each blob generates some patches with its own model. Intuitively, if an image contains three objects, we may have three blobs, one over each object region. Each blob is thus responsible for generating a set of patches the appearance of which corresponds to the object category.

The generation of a patch requires to a) select a blob, and b) generate a patch with the patch model *specific* to that blob.

The blob generation is assumed to follow a Dirichlet process. The Dirichlet process exhibits a self-reinforcing property: the more often a given value has been sampled in the past, the more likely it is to be sampled again. This means that each newly generated patch can either belong to an existing image blob $B_k$ or start a new region.

We characterize each blob $B_k, 1 \leq k \leq K$, with a set of random variables: $\Theta_k = \{\mu_k, \Sigma_k, C_k, l_k, N_k, S_k\}$. $\mu_k, \Sigma_k$ are respectively the mean and the covariance matrix describing the geometric shape of the blob, $l_k$ is the blob label (object category), $C_k$ is a Gaussian mixture model representing the colors of the blob, $N_k$ is the number of patches generated by the blob, $S_k$ is the scale of the blob which is closely related to the distance between the object and the camera.

We characterize each patch $\mathcal{P}_i$ by its features $(w_i^{sift}, w_i^{color}, rgb_i, X_i, d_i)$.

The probability of generating a patch, given that it is generated by the blob $B_k$ of parameters $\Theta_k$: $p(\mathcal{P}|\Theta_k)$ is made of 5 distinct parts, as the model assumes that patch position and scale, color and appearance are independent for a given blob. The probability for a blob $B_k$ to have generated patch $\mathcal{P}$ thus consists of five terms:

$$\begin{aligned} p(\mathcal{P}|\Theta_k) &= p(w^{sift}, w^{color}, rgb, X, d|\Theta_k) \\ &= p(w^{sift}|\Theta_k)p(w^{color}|\Theta_k) \\ &\quad p(rgb|\Theta_k)p(X|\Theta_k)p(d|\Theta_k) \end{aligned} \tag{9.1}$$

The position $X$ of a patch is chosen according to a normal distribution of parameters $\mu_k$ and $\Sigma_k$ for object blobs. It is uniform for background blobs.

We assume that background and object blobs have a Gaussian Mixture color model. The patch depth is closely related to the blob size. Finally, the probability of the SIFT and color codewords only depend on the class label. These distributions encode object appearance information and are responsible for the recognition ability of our model. They are learned using training images in a way described later on (section 9.1.4).

### A MRF structured field of blob assignment

A MRF of blob assignment regularizes the assignment of neighboring patches and also aligns borders between the object and the background with natural image contrast and with strong depth changes. This field is defined over a grid (8-connectivity), nodes correspond to patch centers.

This component basically defines a Gibbs energy that is used to compute conditional probability of patch assignment. This energy has a model fitting term based on the blob representation previously defined as well as neighboring constraint terms for spatial regularization.

The total energy $E$ of the field is the sum of local energies $E_i$ defined for each patch $\mathcal{P}_i$

$$E_i = U_i + \gamma \sum_{j \in \mathcal{N}(i)} V_{i,j} \tag{9.2}$$

where $\mathcal{N}(i)$ represents the 8 neighbors of $\mathcal{P}_i$, $\gamma$ balances the proportion of the two terms. Let $b_i$ be the blob assignement index of patch $\mathcal{P}_i$. $U_i = -\log p(b_i|\mathcal{P}_i, N_{1:K}, \Theta_{b_i})$ is a potential that measures the consistency between the patch and the blob model, and $p(b_i|\mathcal{P}_i, N_{1:K}, \Theta_{b_i})$ is the probability of the blob assignment knowing the patch and the parameters of all the blobs. It stems from the model presented in the last section and makes the link between the two components of the model. $V_{i,j}$ is a potential that measures similarity between two patches $\mathcal{P}_i$ and $\mathcal{P}_j$. It enforces local coherence of

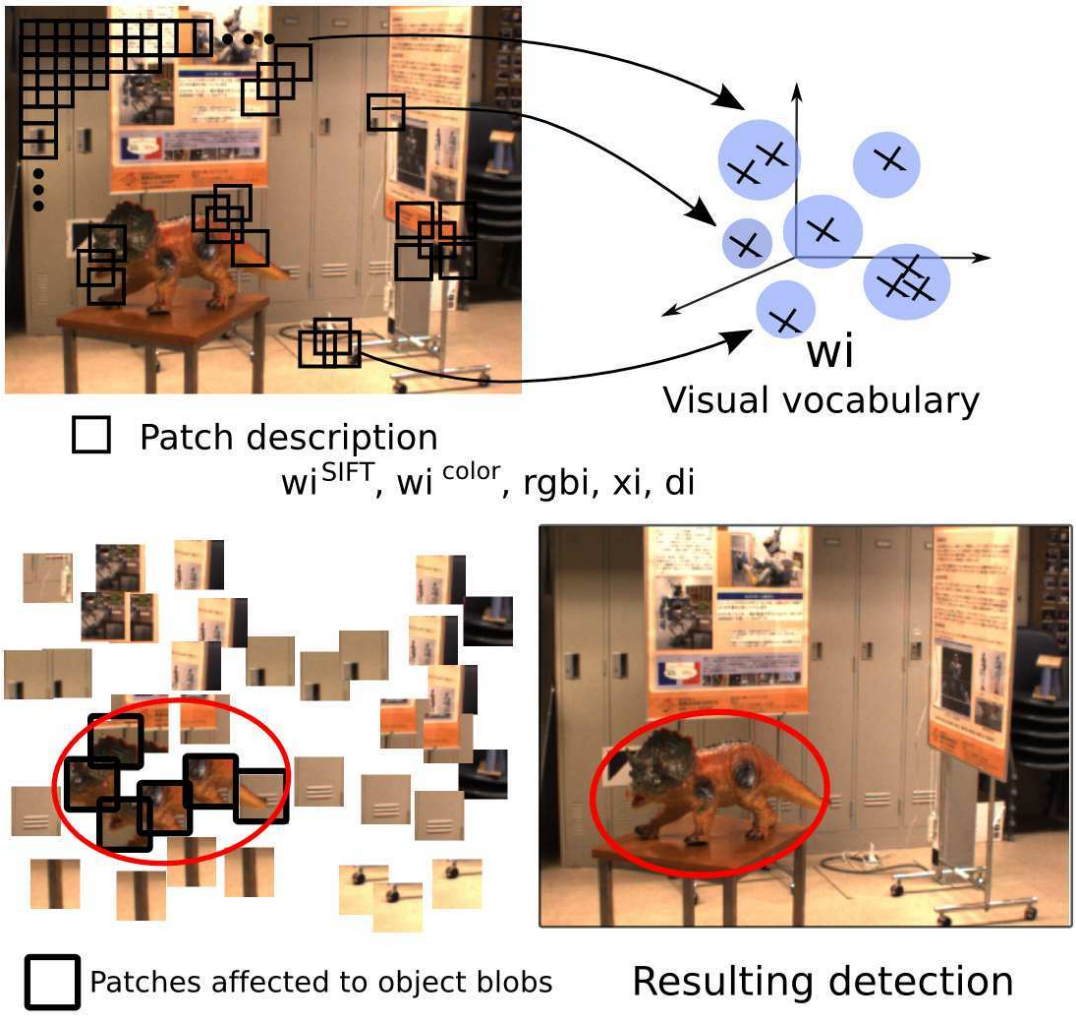



Figure 9.1: First row: patches are extracted in a very dense manner. Each patch is associated to the closest visual word for sift and color descriptors, and then represented by the words indexes $(w_i^{SIFT}, w_i^{color})$, a RGB value $(rgb_i)$, a position $(x_i)$ and a depth $(d_i)$ given by the disparity map. Second row: the model computes the best assignment of patches to object blobs or background and estimates to blobs positions.

the object/background labels, via constraints on the similarity of neighboring patch labels. These contraints are computed using the gradient map $\mathcal{G}$ and the distance between depth values of neighboring patches. It encourages cuts along high image gradients and depth discontinuity.

### 9.1.3 Model Estimation

Now that the model has been defined, its parameters have to be estimated for each image to produce object/background blob labels ($l_i$) and patch assignments to blobs ($b_i$). The model is estimated by a Gibbs sampling algorithm [Nea98] (specific case of Markov Chain Monte Carlo (MCMC) method). A Gibbs sampler generates an instance of parameter values from the distribution of each variable in turn, conditional on the current values of the other variables. More details on the model estimation could be found in [LJJ09].

### 9.1.4 Learning an object appearance

In order to learn the object appearance information, examples of images containing the object are fed to the robot. These learning images are stereoscopic views, taken from several viewpoints [41]. The resulting dense disparity map provides local information that we use to create segmentation masks on positive images. This makes the estimation of object model more accurate by knowing exactly which part of the image belongs to the object and which does not.

We also use a set of negative images (*ie* not containing the object) provided by the robot camera while moving in its environment.

Descriptors (SIFT + color) are extracted on local regions exactly as described for the test images. These descriptors are used first to create visual words by a quantification process, and then to compute the probability for each visual word to be observed as a component of an object blob or not. These probability distributions ($p(w^{sift}|\Theta_k)$ and $p(w^{color}|\Theta_k)$) are stemed from an occurrence histogram obtained by a counting process.

### 9.1.5 Experiments

To evaluate our detection method, we created a test set. It is composed of 118 images taken with the robot cameras. Most of them contains the object in very different conditions. We varied the distance to the object, the view point, the illumination, tried different backgrounds and realized occlusions. The remaining images do not contain the object.

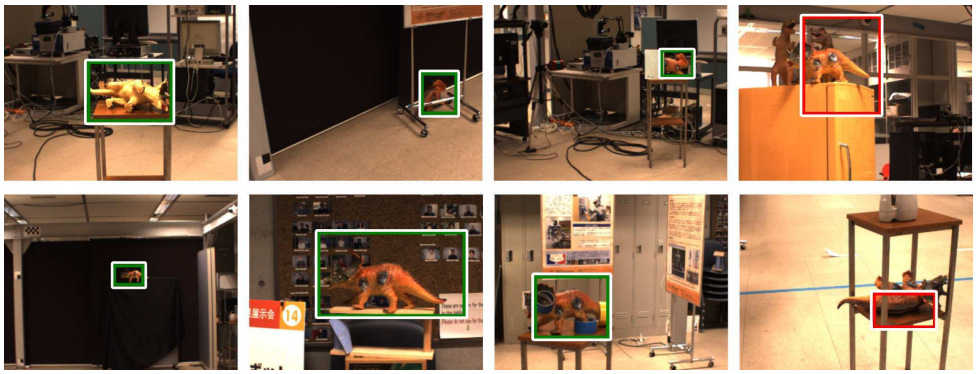


Figure 9.2: Two first columns, examples of correct detections. Last column, examples of false detections.

**Detection evaluation** Our model provides blob shape regions of the image which give approximate position of the object. In order to evaluate the detection performances, these blobs were transformed into bounding boxes describing a rectangular region of the image containing the object. These images were also hand annotated in order to produce a ground truth. A detection is considered as correct if the area of overlap between the predicted bounding box $B_p$ and the ground truth bounding box $B_{gt}$ exceed by a given threshold. We will use 50% by default. This is given by the formula

$$overlap = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} \tag{9.3}$$

The detection performance is evaluated using precision recall values. If we define $N_d$ the number of detection obtained by our method, $N_o$ the number of objects really present on the set of test images and $N_c$ the number of correct detection, we can define the precision $P$ and the recall $R$ with the following formulas: $P = \frac{N_c}{N_d}$ and $R = \frac{N_c}{N_o}$. Multiple detections of the same object are considered as false detection.

In this part we would like to evaluate both the gain of using the depth information, and the overlap between detected bounding boxes and ground truth. Table 9.1 shows the precision and recall values for different thresholds. The two columns present precision and recall values for our method (left), and for the original one (right) which is not using any depth information.

First, we can clearly see the improvement obtained using the information given by the disparity map. The depth introduced within the model helps to estimate accurate boundaries of the object, and give good clues on the object expected size.

Looking at the precision values, we can see that on most of the cases (81%) the detected objects have at least a part in common with the original object.

Finally, the recall values indicate that 64% of the object were partially detected.

| | with depth info | | without depth info | |
|---|---|---|---|---|
| tolerance area | precision | recall | precision | recall |
| 0.2 | 0.81 | 0.64 | 0.65 | 0.61 |
| 0.3 | 0.73 | 0.57 | 0.50 | 0.48 |
| 0.4 | 0.60 | 0.48 | 0.39 | 0.37 |
| 0.5 | 0.41 | 0.32 | 0.26 | 0.24 |
| 0.6 | 0.15 | 0.12 | 0.17 | 0.16 |

Table 9.1: Precision and recall values

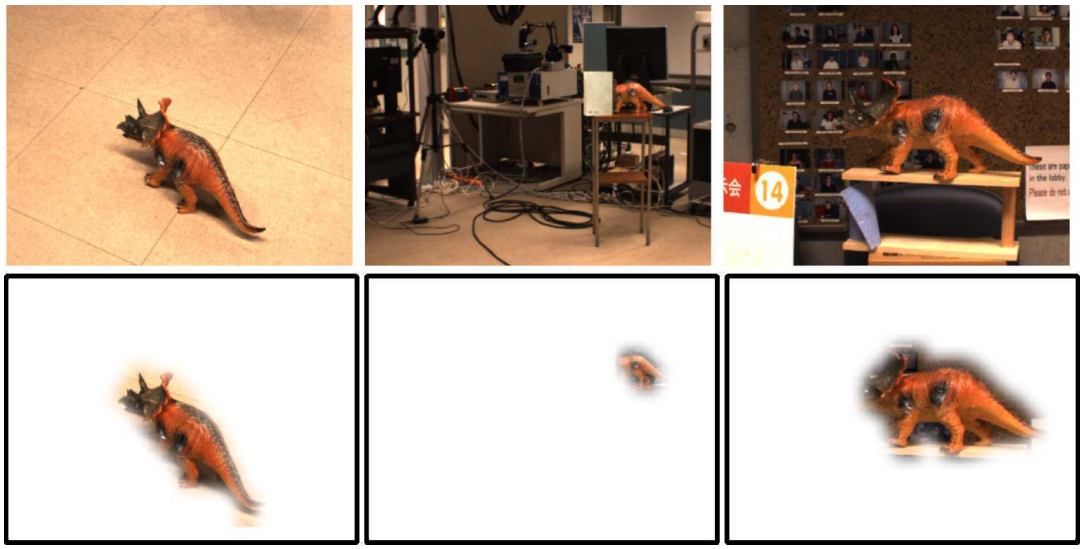


Figure 9.3: The model gives a list of patches actually being components of the model. This produces segmentation masks.

Fig. 9.2 shows examples of correct detections (overlap greater than 50%) and false detections (overlap lower than 50%).

**Qualitative segmentation evaluation** The model also provides the list of patches belonging to a particular object instance. The patches correspond to sets of pixels belonging to their support. Using the information on all patches containing a given pixel, we can create a segmentation of the object. Fig. 9.3 provides segmentation masks in terms of probability maps of the object location on images where the detection was successful.

# Chapter 10

# Simultaneous Localization and Map Building

## 10.1 The simplified model

Simultaneous Localization And Map-building is a corner stone to achieve autonomous behavior. The key point of our system is to make the robot able to self-localize in **real-time** in a new environment equivalent to a room without any a-priori knowledge. Another key point of the presented system is its capability to "close the loop", i.e. to increase the precision by seeing again the same landmarks when returning to a previous location. The core of this work has been proposed by Davison and is described in more details in [10]. We briefly recall the model used to evaluate the motion model and used for prediction. First the estimated state and covariance of the system are given by:

$$\hat{\mathbf{x}} = \begin{pmatrix} \hat{\mathbf{x}}_v \\ \hat{\mathbf{y}}_1 \\ \hat{\mathbf{y}}_2 \\ \vdots \end{pmatrix} \ , \ \mathtt{P} = \begin{bmatrix} \mathtt{P}_{xx} & \mathtt{P}_{xy_1} & \mathtt{P}_{xy_2} & \dots \\ \mathtt{P}_{y_1x} & \mathtt{P}_{y_1y_1} & \mathtt{P}_{y_1y_2} & \dots \\ \mathtt{P}_{y_2x} & \mathtt{P}_{y_2y_1} & \mathtt{P}_{y_2y_2} & \dots \\ \vdots & \vdots & \vdots & \end{bmatrix}. \tag{10.1}$$

Explicitly, the camera state vector $\mathbf{x}_v$ comprises a metric 3D position vector $\mathbf{r}^W$, orientation quaternion $\mathbf{q}^{WR}$, velocity vector $\mathbf{v}^W$ and angular velocity vector $\omega^R$ (a total of 13 parameters). Feature states $\mathbf{y}_i$ are 3D position vectors. In the Extended Kalman Filter (EKF) prediction step, a model for smooth motion anticipates Gaussian-distributed perturbations $\mathbf{V}^W$ and $\mathbf{\Omega}^R$ to the camera linear and angular velocity at each time-step — modeling motion with a generally smooth character. The explicit process model for motion in a time-step $\Delta t$ is:

$$\mathbf{f}_v = \begin{pmatrix} \mathbf{r}^W_{new} \\ \mathbf{q}^{WR}_{new} \\ \mathbf{v}^W_{new} \\ \omega^R_{new} \end{pmatrix} = \begin{pmatrix} \mathbf{r}^W + (\mathbf{v}^W + \mathbf{V}^W)\Delta t \\ \mathbf{q}^{WR} \times \mathbf{q}((\omega^R + \mathbf{\Omega}^R)\Delta t) \\ \mathbf{v}^W + \mathbf{V}^W \\ \omega^R + \mathbf{\Omega}^R \end{pmatrix} \tag{10.2}$$

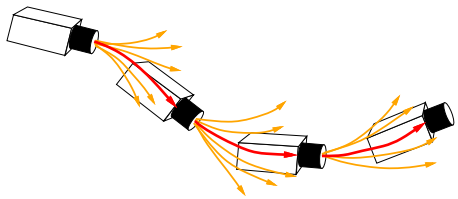


Figure 10.1: Visualization of the "constant velocity" model for smooth motion

Figure 10.1 illustrates the model potential deviations from a constant velocity trajectory. Implementation requires calculation of the Jacobians of this function with respect to both $\mathbf{x}_v$ and the perturbation vector.

The main contribution of this work has been to show that is was possible to perform initialization of landmarks in the EFK using a particle filter to estimate the depth. A more robust initialization has been proposed in [MCD06].

## 10.2 Taking walking information into account

In order to improve the estimation of the Kalman Filter, we added as a measurement the motion from the 3D-LIPM (Linear Inverted Pendulum Model). Supplying an EKF based SLAM system with robot motion model is not new, but specificity of the humanoid robot makes it interesting. Indeed the oscillation of the CoM in the horizontal plane creates a parallax effect which helps new landmarks initialization. Moreover - and this is one of this work innovative contributions - initial landmarks no longer remain necessary. The latters were indeed required in the system [10] for an initial state estimation, but they can be removed if the election of the new landmarks is triggered only when the robot moves. The idea is to pinpoint 3D features only when the robot is in motion. In [44] we described precisely the experimental context and the current limitation of the system (number of landmarks, size of the environment).

## 10.3 Performances

If a drift accumulates for too long, the search for landmarks can take long enough for image frames to be skipped, and the uncertainty grows. This effect builds up and the EKF eventually should be reset. Feeding motion information to the Kalman Filter partially compensates this effect. by locally helping to maintain good estimates of the landmarks. Regarding performances the motion feedback from the Walking Pattern Generator allows to run the SLAM process at 30 ms with some peaks up to 40 ms. A well-known problem of EKF based problem is their inconsistency due to the linearization. Moreover recent advances in local bundle adjustement has shown that they outperform EKF based methods [SMD12b]. The current open issue is to make sufficiently dense and coherent maps to be able to perform real-time path planning

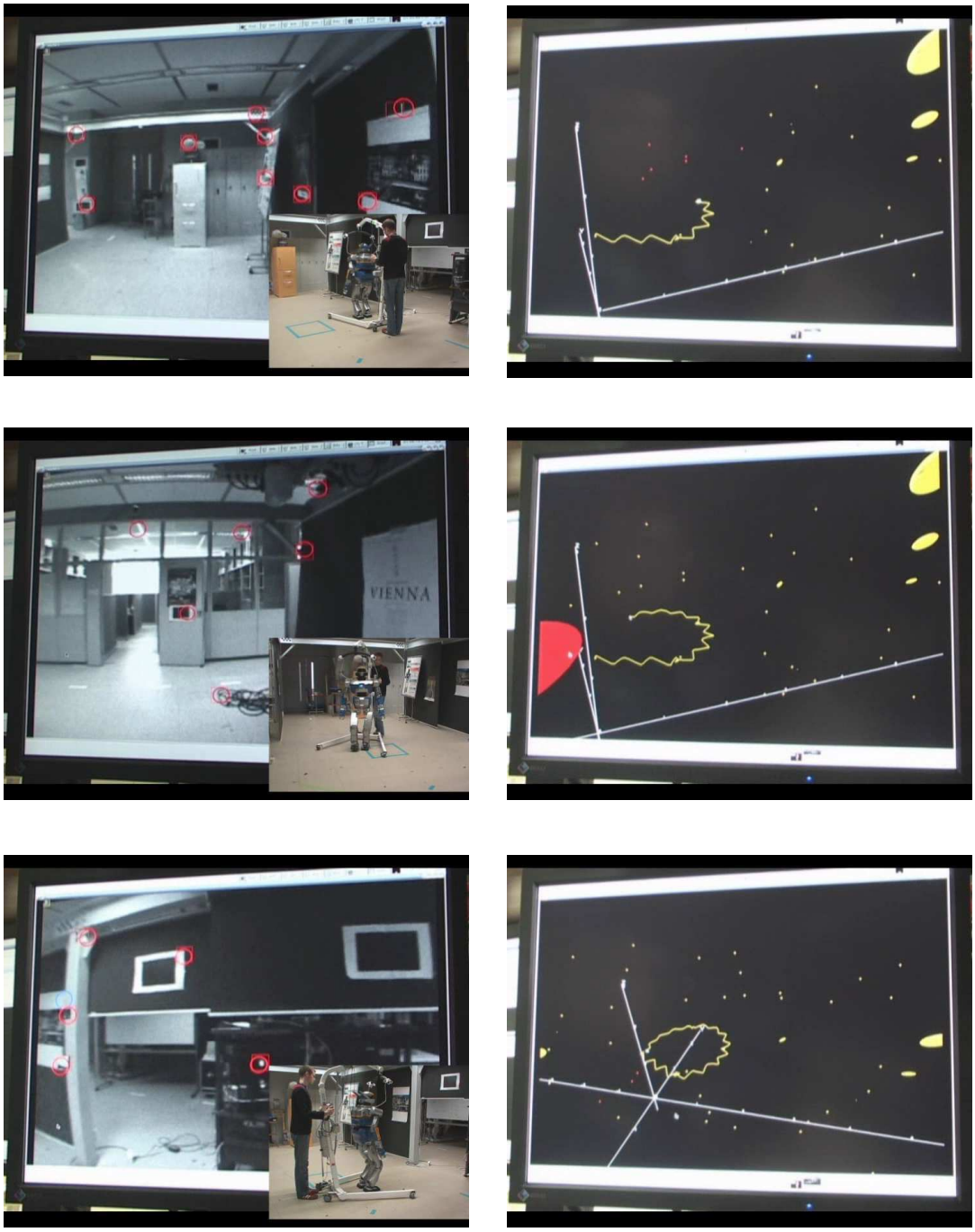



Figure 10.2: Trajectory generated by coupling vision and motion information. (left) Landmarks (right) Trajectories and sparse map.

with humanoid robot in cluttered environments. The impressive result of Newcombe on this topic using Kinect and intensive GPU computation open up new interesting paths in this direction [NIH+11].

# Chapter 11

# Autonomous 3D Object model building

## 11.1 Introduction

This work focuses on the autonomous modeling of an unknown object in a known environment by a humanoid robot. Such functionality enhances the ability of multi-purpose robots in collaboration conditions with human partners. For instance, visual models of new objects can be built and stored inside a knowledge database in an autonomous fashion. The constraints of our system are then set according to such

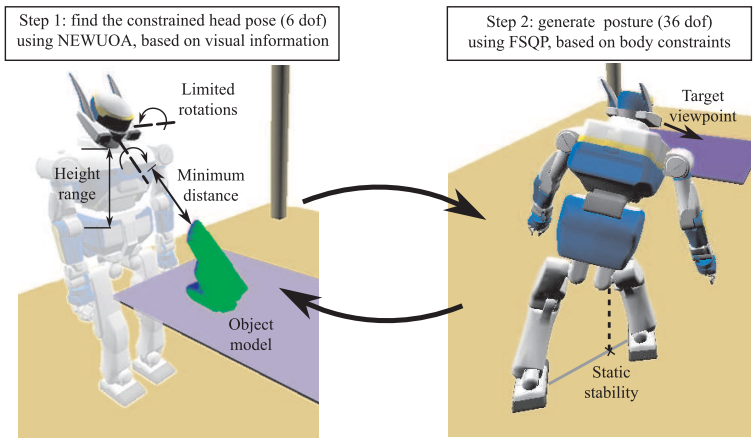



Figure 11.1: Two steps to generate the next posture to update the object model.

scenario as illustrated in Fig. 11.1. The stereo rig should be at a minimal distance, it has limited rotations and is submitted to kinematical constraints. At the same time, the pose of the robot should comply with constraints similar to the ones in *Motion*

*Constraint Satisfaction Problem*[1] i.e. constraints such as balance, self-collision and articular limits. At each step, the system should be able to find the next sensor position which maximize the expected information to be acquired on the unknown object.

In [32] we presents an attempt to include directly such evaluation of viewpoints as a $\mathcal{C}^1$ function which can be included in the PG. Such solution would solve the NBV problem in one coherent step but, although our analytical formulation results in a good evaluation, it has a relatively high computation cost and it also presents high variations in the gradient which result in convergence problems to generate a posture. Moreover it is difficult to put additional vision constraints in such formulation.

Regarding the visual model, we consider the model presented in paragraph 7.1. It aims at allowing not only detection and recognition but also manipulation of the object of interest.

The main originality of this work comes from the solution given to generate viewing postures for a humanoid robot considering its incremental knowledge of the environment and the object of interest. Two steps are necessary to achieve the result and are depicted in Fig. 11.1: (i) find a viewpoint which maximizes the amount of unknown data from the object which can be visible, and (ii) generate a whole-body pose statically stable, collision-free and with the head positioned in order to get the desired camera pose. This solution is then integrated with other recent works relative to path planning and control in order to simulate realistically the modeling task using an HRP-2 robot model and the dynamic engine of OpenHRP-3.

Sections 11.2 and 11.3 respectively detail the first and second step to generate a robot posture. Finally some results of our method on simulation are presented in section 11.4.

## 11.2 Generation of the desired viewpoint

The problem can be formulated as the minimization of a function $f$ which evaluates the unknown visible area projected into the camera. Traditional works in the NBV field solve this problem considering the camera space as the input space of $f$. Then the problem dimensionality is reduced to sample the configuration space and find a solution in an acceptable amount of time. To do so, some assumptions are usually made on the size and complexity of the object to model and the environment is considered free of obstacles. By using recent hardware and optimization algorithms though, it is possible to relax these assumptions while keeping a reasonable computation time. In order to broaden the types of object to be modeled while taking into account the constraints related to the use of a humanoid, a novel solution to the Next-Best-View problem is introduced by using two steps, illustrated in Fig. 11.1: first, find a camera

[1] $MCSP_{\mathbf{u}}$ described by Eq.(3.6)

position and orientation that maximizes the amount of unknown visible while solving specific constraints related to the robot head, then generate a whole-body posture for the robot by considering the desired viewpoint as well as additional constraints related to the humanoid body.

We propose to solve the first step by using NEWUOA [Pow04], a method that search for the minimum of a function by refining its quadratic approximation through a deterministic iterative sampling, and which can thus be -used for non-differentiable functions. The sampled vectors at each step in the NEWUOA search process are selected according to the previous sampling results and the actual state of the quadratic approximation. A trust region must be defined using two radius parameters: $\rho_{beg}$ and $\rho_{end}$, and a given starting vector, which will be the camera pose in our case. This region also influences the sampling process but it does not limit it. Indeed, depending on the quadratic approximation found, vectors outside of this region can be tested. NEWUOA has the advantages of being fast and robust to noise while allowing us to keep the 6 degrees of freedom required for the viewpoint.

### 11.2.1 Evaluation of the unknown visible

In this approach, the estimation of unknown data visible from a specific viewpoint relies on the visualization of the current occupancy grid. This can be computed quickly by taking advantage of current hardware acceleration. As oscillations of small amplitude have only a negligible influence on the convergence of NEWUOA, such approximate representation is considered to be a useful indicator to find a good viewpoint. The amount of unknown visible, noted $N_{up}$, is set as the number of *unknown* voxels visible multiplied by the logarithm of the number of *unknown* voxels' pixels, in order to influence the robot to have a closer and more detailed look at the object. In fact, due to the perspective projection, more voxels can be visible from farther position but then they appear unnecessary small.

### 11.2.2 Constraints on the camera pose

Though NEWUOA is supposed to be used for unconstrained optimization, some constraints on the camera pose need to be solved in order to be able to generate a posture with the PG from the computed viewpoint. The constraints on the camera position $\mathbf{C}$ and orientation $\psi_{\mathbf{c}}$ included in the evaluation function of the first step given to NEWUOA are:

$$\begin{cases} C_{zmin} < \mathbf{C}_z < C_{zmax} & (11.1) \\ \forall Vi, d_{min} < d(\mathbf{C}, \mathbf{Vi_{center}}) & (11.2) \\ \psi_{\mathbf{c}xmin} < \psi_{\mathbf{c}x} < \psi_{\mathbf{c}xmax} & (11.3) \\ \psi_{\mathbf{c}ymin} < \psi_{\mathbf{c}y} < \psi_{\mathbf{c}ymax} & (11.4) \\ N_l > N_{lmin} & (11.5) \\ \forall \mathbf{C} \neq \mathbf{F}_i, \psi_{\mathbf{c}} \neq \mathbf{Fr}_i & (11.6) \end{cases}$$

The range of the camera height is limited by (11.1) to what is accessible by the humanoid size and joint possible configurations. A minimum limit distance $d_{min}$ is needed between the robot head and $\mathbf{Vi_{center}}$ the center of each voxel of the object in order to use efficiently the stereo vision. This constraint is expressed in (11.2). The rotations on X and Y axes are limited by (11.3) and (11.4) to ranges set according to the robot particularities. The constraint (11.5) keeps a minimum number of landmarks, i.e. features that were detected in previous views, visible from the resulting viewpoint. Finally, the particular constraint (11.6) ensures that the resulting pose will not be near previously found poses, with position $F_i$ and orientation $Fr_i$, which could not be reached by following steps in the modeling process. It can also be used to avoid positions in the environment where known obstacles are located. This constraint is necessary to ensure the algorithm can converge toward a valid posture even though some constraints are not expressed in the viewpoint or posture generation search. For example, some obstacles in the environment may limit the possible motion trajectories while not visually occluding the view of the camera and thus, the motion planner may fail to find a way between the current posture and the target one. A detailed formulation of $f_e$ can be found in [5].

### 11.2.3 NEWUOA configuration

NEWUOA is used to seek the minimum of $f_e$ by approximating it with a quadratic model, as illustrated in the left graph of Fig. 11.2. Three parameters are used as input to this optimization algorithm: an initial vector from where the search is started, a value which delimits the trust region around the initial vector in order to build the initial quadratic approximation, and a desired accuracy value used as a stopping criterion. Due to the nature of the NBV problem and due to the constraints

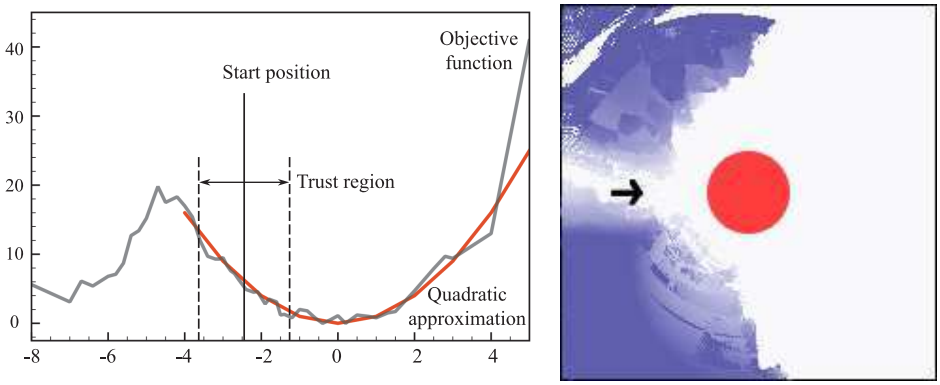



Figure 11.2: NEWUOA evaluation. (left) NEWUOA's method to find the minimum of a non-differentiable function. (right) Objective function results depending on the sensor position XY. The object location is displayed as the disk in the center and the previous perception pose is represented as an arrow.

used, our objective function can present many local minima that are quite disjoint

as is presented in the right part of Fig. 11.2. This figure shows the best results for $f_e$ obtained for the knight object carved once from Fig. 11.1, by constraining the camera at a fixed height and moving the sensor on the XY plane. For each sampled position, the orientation with the best result for the objective function is selected. Darker points corresponds to better evaluations. The landmark visibility constraint formulation as well as the presence of possible self-occlusions can result in abrupt local variations in the objective function. The minimum distance constraint also produces an important maximum in the middle of the search space. Thus the trust region used needs to be limited to some local space around the object in order to be considered pertinent enough.

### 11.2.4 Viewpoint search process

During our tests, we found that the quality of the results depends greatly on the

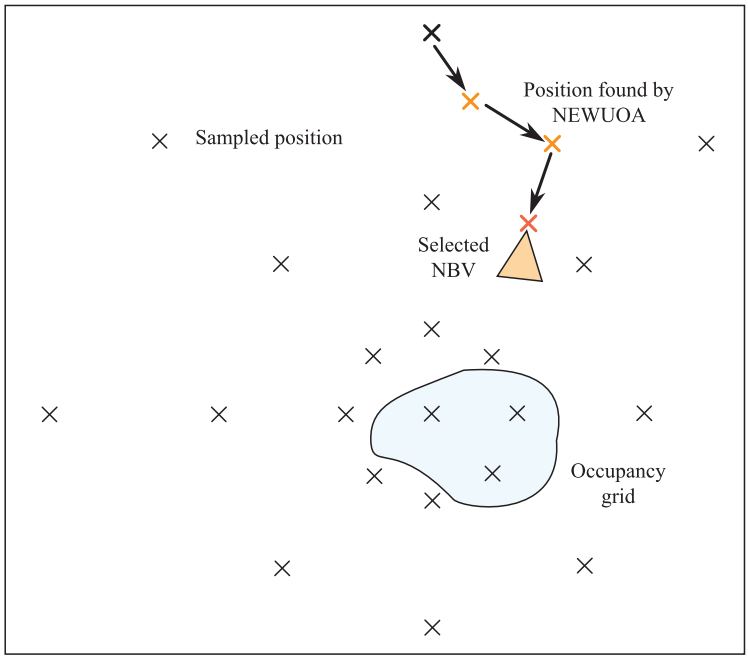



Figure 11.3: Illustration of the NBV selection for the robot head.

starting poses given, thus two additional techniques are implemented which are illustrated in Fig. 11.3. First, we run NEWUOA in an iterative way, i.e. it is run once using a defined starting pose and run again by using its result configuration as a new starting pose. This is done until a chosen maximum number of iterations has been reached, or until the result pose is not better than the last starting one. A step of this iterative process is formulated as:

$$pose_k = Newuoa_k\left(pose_{k-1}\right) \tag{11.7}$$

with $k$ the iteration number of the NEWUOA algorithm from 1 to $n$, and $pose_{k-1}$ and $pose_k$ respectively the starting and found camera poses.

Second, we also reached a significant improvement of the result by precomputing a set of 3D starting positions around the object and launching the iterative process for each of them. Results of all optimizations are then compared to select the best camera pose. The sampled positions can be generated inside the object to handle cases where it has large empty spaces. For example, the algorithm can be applied to model both the inside and outside of a house. The positions are distributed in the space relatively to their distance to the object: the density gets smaller when getting far away from the object as greater motions are required to get significant visual changes.

## 11.3 The Posture Generator

The Posture Generator (PG), presented as part of a work from Escande *et al.*[AEM06], provides a whole-body posture in the second step of our NBV algorithm. The PG relies on FSQP, a gradient-based optimization method, to give a posture that minimizes an objective function while solving given constraints. Once an optimal camera pose has been found in the first step, the result is used as a constraint on the humanoid robot head in order to generate a whole-body posture that also takes into account other constraints: static stability, self-collisions avoidance, collisions avoidance with the environment, keeping the feet flat on the ground, and joints limitations. In cases where the PG cannot converge, the goal camera pose is put inside the list of forbidden poses which is used in the constraint 11.6 and the first step is launched again to find another viewpoint.

## 11.4 Simulation and experimental results

### 11.4.1 NEWUOA tests for camera pose evaluation

Using our objective function $f_e$ the initial conditions for a NEWUOA search influences the viewpoint found. We thus tested the influence of the starting position and of the trust region parameters. Some results are presented in Fig. 11.4. The left graph of the figure shows the average results for the viewpoint obtained depending on the $\rho$ parameters. $\rho_{beg}$ sets the maximum variation that can be taken by the camera pose parameters for the initial quadratic approximation, and the parameter $\rho_{end}$ sets the desired accuracy of the optimum search. The tests were conducted by selecting a camera pose around an object model and by launching the optimization with different values for $\rho_{beg}$ and $\rho_{end}$. This was repeated for 14 different objects with 3 different starting poses for each. Overall, better evaluated poses are obtained when $\rho_{beg}$ is equal or superior to the object maximum size, and when $\rho_{end}$ is smaller than one

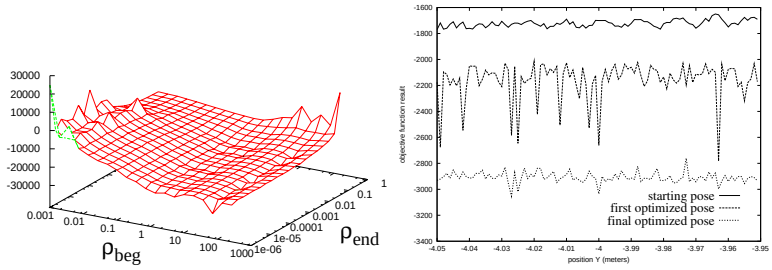



Figure 11.4: Influence of NEWUOA parameters on the search results. (left) Influence of the trust region parameters. $\rho$ parameters are multiplied by the object maximum size. (right) Influence of the starting position on $f_e(pose_s)$ and the viewpoints found by our iterative optimization process $Newuoa_1\,(pose_s)$ and $Newuoa_n\,(pose_{n-1})$.

hundredth of this size.

The right graph shows the influence of the starting pose on the viewpoint found. This was tested by launching a NEWUOA search with different initial configurations, i.e the camera is translated on the Y axis in front of an object model. These tests were done with $\rho_{beg} = 0.4$ and $\rho_{end} = 10^{-5}$. First, we can note that the evaluation of the unknown function, i.e the 'starting pose' curve, can change abruptly even with small variations of the pose. This highlights the complexity of our evaluation function, already discussed in 11.2.3, which has a lot of local minima. Depending on the starting position, NEWUOA can thus generate relatively different quadratic approximations which will lead to different samples selected. This graph also highlights that, though a single iteration of NEWUOA results in an improved pose, it is often stuck inside a local minimum. Nevertheless, by using successive iterations, much better viewpoints are reached. In fact, the camera can get moved up to 0.7 meter and rotated up to about 50 degrees in many final optimized poses around a small object, e.g. 0.4 meter long. In order to find a good pose, a large number of iterations is not necessary. In this test, the average number of iterations was 5 and the maximum number allowed, which was set to 10, was reached for only 2 percent of the tested initial poses.

### 11.4.2 NEWUOA VS fixed sampling

We compared the results obtained with a simple NEWUOA search against a pre-computed fixed sampling of the 6D viewpoint configuration space. This sampling is done around the last position where a space carving operation has been done. The number of samples as well as the limits of the area to test are defined manually for each of the 6 dimensions.

Not surprisingly, the fixed sampling can result in viewpoints with similar or better results using roughly the same number of sampled vectors. As noted earlier, depending on various parameters such as the object complexity or the landmarks distribution, the

NEWUOA search may fall into local minima close to the starting pose. Nevertheless, such local minima can be reached by NEWUOA using less samples than a fixed sampling of the local space. Thus our search for a viewpoint presented in 11.2.4 includes the two methods: first, have a rough sampling of positions in the areas of interest, then use NEWUOA to refine the search for the closest local minima.

### 11.4.3 Modeling process simulation

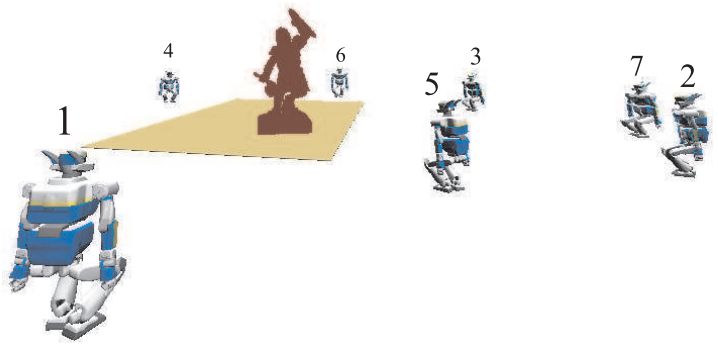



Figure 11.5: Postures generated for the reconstruction of a 3 meters high object.

The experimental setting is simulated by having a 3D object perceived by a virtual camera. One example of generated postures to complete the modeling is presented in Fig. 11.5. The first posture is set manually and the six following are generated using our NBV algorithm. The trust region parameters, $\rho_{beg}$ and $\rho_{end}$, were set respectively to 0.4 and 10e-5. Other parameters settings are: $p = 6$, $\gamma = 20$, $d_{min} = 0.6$, $Nlm_{min} = 5$, $\eta = 1$, $\delta = 1$, $\lambda_z = 200$, $\lambda_x = 80$, $\lambda_y = 80$, $\lambda_d = 100$, $\lambda_l = 1$ and $\lambda_f = 1000$.

### 11.4.4 Computation time

Each evaluation of a viewpoint relies on the OpenGL visualization of the occupancy grid which is loaded inside the graphic card memory. The evaluation time is thus relatively small and stays in the order of $10^{-2}$ seconds though, of course, it can vary depending on the number of voxels in the model.

The search for the best viewpoint in the first step of the algorithm typically requires few thousands of evaluation. This depends on the number of sampled positions for the preliminary search, and the input parameters for NEWUOA. During our tests, the first step could give a solution between 10 seconds and one minute.

The second step can generate a posture in the order of $10^{-1}$ seconds if the starting conditions are relatively close to the solution and if there are no obstacles in the final location. In others cases, it can take up to few seconds to get a solution or to abort the search.

### 11.4.5 Pose generation

Figure 11.6: Postures generated on an HRP-2 humanoid.

The second step of our Next-Best-View algorithm was tested by verifying that camera poses obtained in the first step do not result in a constraint, on the robot head, impossible to satisfy when set in the PG with other constraints. Several camera poses were computed using different virtual objects with different states of space carving and the landmarks were randomly generated amongst the known voxels on the surface of the object. Some of these poses, presented in Fig. 11.6 were tested on a real HRP-2 robot to ensure their stability and the avoidance of self-occlusions.

The tests confirmed that the constraints set in the first step reduce the possible poses to what is achievable by the PG with our current settings. It should be remarked that the posture used to initialize the PG has some influence on the convergence. Highly constrained postures such as squatting poses are often difficult to generate from the default standing posture but the opposite is much easier. Thus we are initializing the PG with a squatting posture.

## 11.5 Conclusion

We introduced a new method to generate automatically postures for a humanoid robot depending on visual cues. The algorithm presented differs significantly from previous Next-Best-View solutions due to the hypotheses and constraints involved.

The postures are selected amongst the possible configurations allowed by stability, collisions, joint limitations and visual constraints, so as to complete the modeling of an unknown object using a minimum number of postures. Two complementary optimization methods, NEWUOA and FSQP, are used in our two step algorithm to generate each next-best-posture. The NEWUOA search, coupled with a fixed

sampling of the robot head configuration space, can deal efficiently with the noise and discontinuities of the viewpoint evaluation function to minimize. The Posture Generator can then quickly find a posture addressing all necessary constraints on the humanoid body. This approach was validated in simulation by building successfully models of various objects with complex shape.

This work is being integrated with other components focused on vision, motion planning and motion control tasks, in order to test experimentally the autonomous modeling of the object with an HRP-2 robot.

# Chapter 12

# Sensor Planning for Active Visual Search with a Humanoid Robot

This work adresses the sensor planning problem in the case of active object search. The search is formalized as an optimization problem in which the goal is to maximize the target detection probability while minimizing the energy/distance and time to achieve the task. Natural constraints on the camera parameter space based on the characteristics of the recognition system are used to reduce the dimension of the problem and to speed up the optimization process to achieve a real time behavior. We present simulation and real experimental results using an HRP-2 robot.

## 12.1 Introduction

### 12.1.1 Problem statement and contributions

We search for the position of the camera that maximize the function of probability to observe the hidden object.

Object search is a sensor planning problem which is proven to be NP-complete [YT96] thus a heuristic strategy is needed to overcome that task. Because of the limited field of view, the limited depth, the lighting condition, the recognition algorithm limitation, and possible occlusion, many images from different points of view are necessary to detect and locate a given object.

The initial knowledge of the target position is encoded in a discrete presence probability [YT99] which is updated after each detection attempt. By combining the target distribution knowledge and a model of the recognition system accuracy, we are able to calculate the likelihood of detecting the target for a given sensor parame-

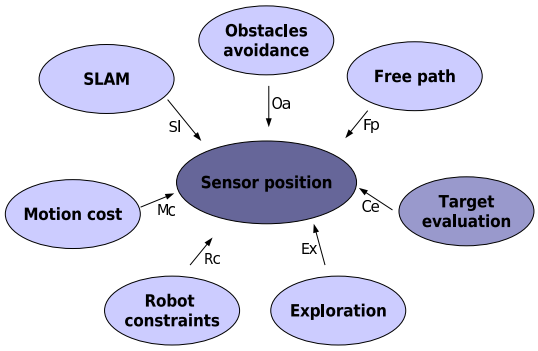



Figure 12.1: Oa: Local obstacle avoidance with visual servoing, Fp: Each motion must be within the persived free path, Ce: Determining whether the target is the object we are looking for or not, Ex: Exploration for mapping and electing new targets for further evaluation, Rc: Each motion must comply with humanoid robot constraint, Mc: Criterion for promoting shorter motion, Sl: Imaging known feature to help the SLAM process

ter. The proposed planning strategy consists in optimizing a rating function at each sensing step. The rating function analyzes the expected field of view (according to already mapped environment) for a given configuration according to various criteria defined further below. In [51], we introduced the concept of *Visibility Map* a statistical accumulator in the sensor configuration space which takes into account the characteristics of the recognition system to constrain the sensor configuration space and speed up the optimization. Here we present the full search planning strategy along with experiments on the HRP-2 humanoid robot.

### 12.1.2 Related and similar works

Whereas few works on active 3D object search are available, the sensor planning research field has received a lot of attention specially for model building, environment mapping and localization.

Works done by Ye and Tsotsos [YT99] tackle the field of sensor planning for 3D object search. The search agent's knowledge of object location is encoded as a discrete probability density which is updated after each sensing action. The detection function uses a simple recognition algorithm, and all factors which influence the detection ability such as imaging parameters, lighting condition, complexity of the background, occlusions etc. are included in the detection function value by averaging experimental results done under various conditions. The vision system uses one pan tilt zoom camera and a laser range finder to build a model of the environment. The search is not really 3D as, the object is recognized using a 2D technique, and the height of the camera is fixed.

Wixon [Wix94] uses the idea of indirect search (in which one first finds an object

that commonly has a spatial relationship with the target, and then restricts the search in the spatial area defined by that relationship) he proposes a mathematical model of search efficiency, which shows that indirect search can improve the processing speed.

Works by Sujan [SD05] are not focused on object search but on accurate mapping of unknown environment by the mean of sensor planning. The author proposes a model based on iterative planning, driven by an evaluation function based on Shannon's information theory. The camera parameter space is explored and each configuration is evaluated according to the evaluation function. No computational timing tests are provided, but the algorithm focuses on configurations which are close to obstacles or to unknown areas to improve the algorithm efficiency. This latter constraint is formalized with the notion of visibility map introduced in 12.2.3.

Other interesting results are those of [SS04a] which deals with gaze control of a humanoid robot for authonomous naviguation. The sensor direction planning is done to fulfill two objectives, obstacle avoidance and self-localization while performing a given trajectory. A decision making strategy combines both objectives to decide the next sensor orientation. The constraint on the sensor position which must follow a predefined trajectory reduces the search space dimension and removes one of the challenging aspect of the problem.

In [LGG92] the authors present a method for selecting viewpoints and sensing tasks to confirm by a multisensory perception machine, an identification hypothesis previously generated. The determination relies on the use of a compiled knowledge base that links object and sensor models, and defines a priori the best sensing tasks to be performed. The object model seems heavy to build and no automated method is proposed.

The operational research community [Koo80] has extensively studied the problem of optimal search, they came up with interesting theoretical results on search effort allocation. These work are mostly intended for search and rescue procedures and are conducted from airplanes and thus do not take into account occlusions and are done in 2D.

The Next Best View (NBV) research field [Con85b] studied the sensor planning problem mainly for C.A.D. model building. These works, although sharing some common aspects with the present topic, rely on the fundamental assumption that the object is always in the sensor field.

## 12.2 Constraints on the sensor - Adaptative subsampling

### 12.2.1 Model of the recognition system

All recognition algorithms have some restrictions regarding the imaging condition (lighting, occlusion, scale...). One of the main assumption which can be easily con-

trolled by active vision is the scale limitation: the smallest scale at which the object can still be recognized constitute a maximum distance limit for the recognition algorithm ($R_{max}$). It is also suitable to have a sensor configuration in which the whole object is projected inside the image in order to maximize the number of imaged features. This imposes a lower limit for the sensor distance to the object ($R_{min}$).

Without any loss of generality regarding the recognition algorithm, we can assume that these bounding values ($R_{min}$ and $R_{max}$) are determined theoretically or experimentally during the model building and are stored with the object model. These limit values depend on the recognition algorithm and on the characteristics of the searched object and are used to further constrain the sensor parameters in order to improve optimization time. We assume that a model of the recognition system, which gives the accuracy of the recognition depending on the position of the target relative to the sensor, is available. As a first approximation, we assume that the accuracy $\rho$ only depends on the distance to the sensor that is $\rho(v_i) = \rho(z)$. For instance, we use here a gaussian formulation of the recognition accuracy $\rho$ (equation 12.1), in which $z$ is the distance of a given voxel to the camera optical center.

$$\rho(c, v_i) = \rho(v_i|_{object}) = \rho(z) = \exp \frac{1}{2}\Big(\frac{z-m}{\sigma}\Big)^2 \tag{12.1}$$

with

$$m = \frac{R_{max} + R_{min}}{2} \quad and \quad \sigma = \frac{R_{max} - R_{min}}{2}$$

The accuracy model will be used in two different parts of the system:

- in the prediction part, to modulate the probability of detecting the target according the accuracy at which it will be perceived.
- in the evaluation part, after a recognition attempt to modulate the confidence we give to the detection system the result of the target distribution update process.

### 12.2.2 Robot kinematics constraints

The configuration space of the stereoscopic head has initially 6 DOF, but because the roll parameter (rotation around the line of sight) has a small influence on the visible area (the stereoscopic field of view is square), the problem is reduced to 5 dimensions.

The sensor configuration space is discretized with the same resolution as the occupancy grid for the x, y and z parameters (5 cm). Whereas for pan and tilt, a resolution of half the stereoscopic field of view value, ($18^\circ$) is used.

Unlike [SD05], the visual sensor, which is located in the head of the robot, is subjected to balance constraint. In this work we don't consider robot postures in

which the head of the robot goes over obstacles, thus the sensors configuration space is restricted by the 2D projection of obstacles on the visual plane. Moreover, we introduce a safety margin around obstacles in which sensor placement will not be evaluated.

In previous works limitations of the robot pattern generator prevented the robot's head height to be changed. As this limitation has been removed (a new pattern generator is available [48] which accepts large perturbations on the waist height) a new degree of freedom has been introduced casting the sensor planning problem in 5D.

Here to simplify the robot posture needed to reach a given sensor configuration, we restrict the sensor motion along the z axis between $Z_{max}$ and $Z_{min}$. These sensor position are reachable by changing only the waist height: $Z_{max}$ is the natural height of the robot sensors while walking which is 1m40 and $Z_{min}$ is set to 1m20. Lower $Z$ configurations could be reached by the sensor, but these configurations require additional contact point with the ground in order to stabilize the posture.

With a typical environment size of 6x12x2 meters, the configuration space $\Omega$ of the sensor has around 24 millions configurations. A greedy optimization approach is impossible to achieve in a reasonable time. To overcome that problem, we propose an *adaptative subsampling of the sensor configuration space* which takes into account the limitations of the recognition system.

### 12.2.3 The visibility map

The basic idea of the sensor configuration space reduction is to keep sensor pose which meets certain requirements:

- For each configuration, a certain amount of points of interest must be visible.
- Points of interest must be seen under imaging conditions which allow a reliable recognition.
- Configuration must have a low coupling (their view field must weakly intercept).
- The set of all configurations must partition the visible space.

In this context, a point of interest refers to a 3D environment point that might be part of the target object. Moreover these points must lie on the frontier between explored and unexplored environment (ie. unknown or solid voxels with an empty neighbor).

In order to achieve these criteria, we use the concept of visibility map introduced in [51]. We describe here the steps leading to the construction of this map.

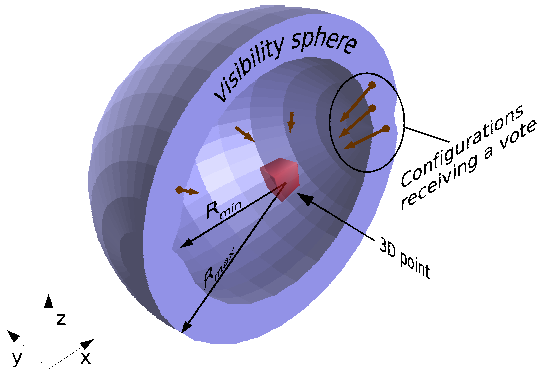



Figure 12.2: The visibility sphere represents the 5D configuration set of the stereoscopic sensor in which a particular 3D point can be well recognized by a given recognition algorithm.

A given 3D point in the environment votes in the sensor configuration space for all the configurations from which it can be imaged under good conditions (conditions allowing a reliable recognition given a recognition method), this is what we call the visibility sphere of a point. This hollow sphere has an inner radius of $R_{min}$ and an outer radius of $R_{max}$ as defined in 12.2.1. Fig. 12.2 shows a 3D representation of a visibility sphere.

All the points of the visible 3D surface of the environment create their own visibility sphere. The contribution of all the visibility spheres are summed up in an accumulation map we call the visibility map.

Fig. 12.3 shows a cut normal to the $z$ axis of a 3D projection of the visibility map in order to allow a 2D representation. The visibility map is similar to the Hough space in image processing in the sense that its a statistical accumulator in the configuration space. Just as the Hough space, the visibility map cannot be used out of the box and need some filtering/clustering described in 12.2.4.

The visibility map can be seen as a 5 dimension, gray values map:

- The value of each configuration in the visibility map is called the visibility of the configuration and noted $V(c)$. A candidate is a configuration for which $V(c) > 0$.
- The set of candidates which have the same x and y parameter is called a cluster (the cluster visibility is the sum of all its candidates visibility). Fig. 12.3 shows in fact the clusters of the visibility map.

Three key points allow this map to be computed efficiently:

1. The visibility sphere of a given point is precomputed once and stored in a lookup-table. Adding a point contribution to the visibility map is then simply achieved by adding an offset to the LUT.

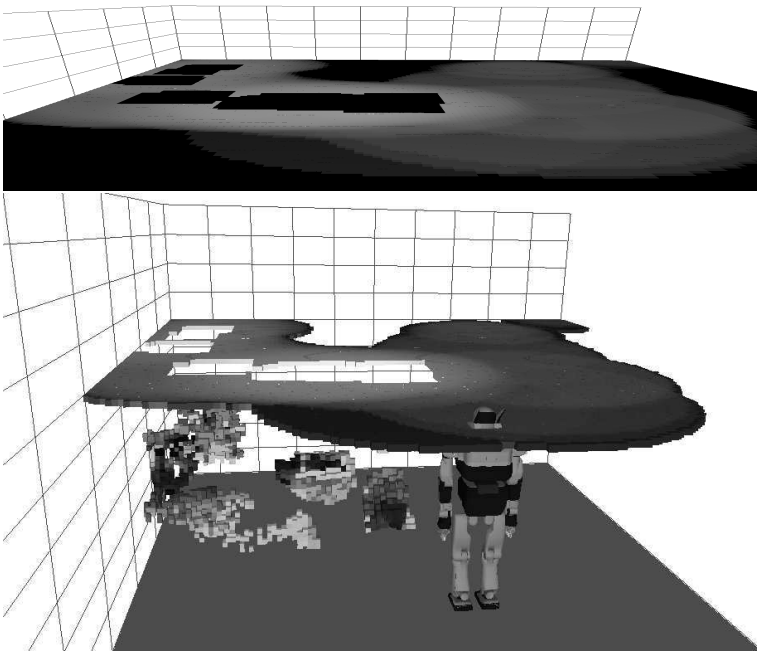


Figure 12.3: This visibility map is only computed for reconstructed solid points (gray points under the plane). Each point is creating a visibility sphere around it. Lighter area on the plane represent configurations in which the solid points can be well imaged

2. The visibility map update is done incrementally, which means that only points which have a change in their state will be considered: new boundary points add their votes to the visibility map and votes of removed points are subtracted. Because of its incremental nature, the visibility map computation gets faster (in average).

3. Each layer (a different $z$ value) is independent and is computed by a dedicated thread.

### 12.2.4 Local maxima extraction

In order to achieve the criteria listed in the previous section the visibility map is filtered. The coupling (Fig. 12.4) inside the same cluster is low because a change in the pan, tilt parameters will bring a lot of new information in the field of view. On the other hand, a change in the x, y, z parameters will most likely produce a small change in the field of view. A local maxima extraction of the visibility map based on a window with different size for the rotation and translation parameters will output the 'locally best' configurations for which a reasonable amount of points is visible. A small size is used for the pan and tilt parameter, reflecting the fact that configurations with close orientation values are weakly coupled. A larger window size is used on the translation parameters.

The extraction window, called the decoupling mask is created as follow:

We denote $S(c|_{object})$ the amount of object surface imaged when the sensor configuration is $c$ in the object reference frame. We call the coupling between two views the amount of common object information (surface) imaged by these two views. For a small change in the configuration $\delta c$ the coupling is defined as :

$$coupling = \frac{S(c_{ref}|_{object} + \delta c) \bigcap S(c_{ref}|_{object})}{S(c_{ref}|_{object})}$$

In the general case, the coupling depends on the object configuration in the reference view. To simplify the coupling formulation, the object is represented by its bounding sphere put at a fixed position (at a distance of $\frac{R_{max}-R_{min}}{2}$ on the optical axis of the reference view). With these hypothesis the coupling only depends on $\delta c$ (parameter shift) and the coupling formulation becomes: $S_{\cap}(\delta c)$. With this definition of the view coupling, we define the couplinf mask $M$ as:

$$M = \{\delta c \ / \ S_{\cap}(\delta c) > C\} \tag{12.2}$$

the set of parameter changes for which the coupling is above a given threshold $C$. We use here a threshold of 50%, the mask is computed offline and is valid for a given object size.

The visibility map filtering is then performed by sliding the mask over the configuration space and selecting configurations which exhibit a maximum over the neighborhood defined by the mask. This leads to the creation of the reduced configuration space (equation 12.3).

$$\tilde{\Omega} = \{c \in \Omega \ / \ \forall \delta c \in M, \ V(c) \geq V(c + \delta c)\} \tag{12.3}$$

The greedy exploration of sensor's parameter space is constrained to the local maxima of the visibility map denoted by $\tilde{\Omega}$ which is a subset of $\Omega$. An interesting feature of the visibility map comes from the fact that solid and unknown points are treated the same way, and generate their visibility sphere, thus suitable configurations for exploring unknown areas are also created. The constraint achieved by the visibility map and the local maxima extraction drastically reduces the configurations to consider at each step. Typical values are between 200 and 1000 candidates (to compare with the 24 millions possible sensor placement). Next section will present the overall algorithm.

## 12.3 Algorithm

The flowchart of the next best view selection process is depicted in Fig. 12.5. When a new world model is available, the corresponding visibility map is computed and

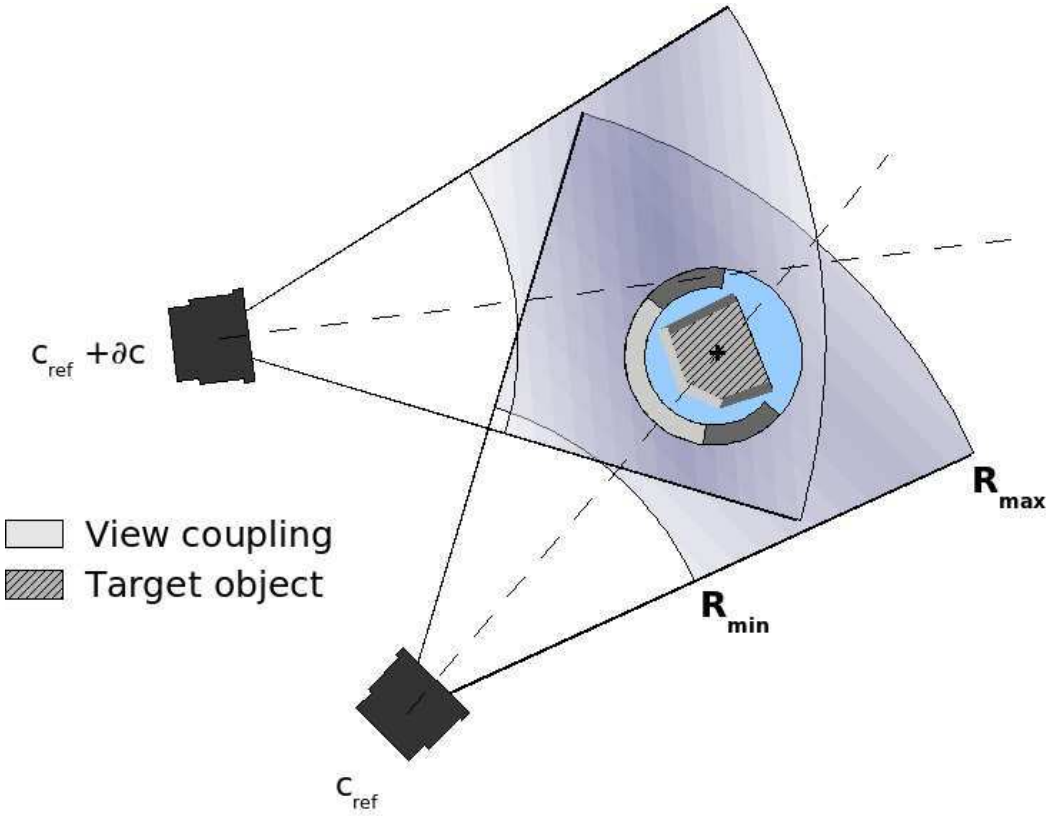



Figure 12.4: Illustration of the coupling between views

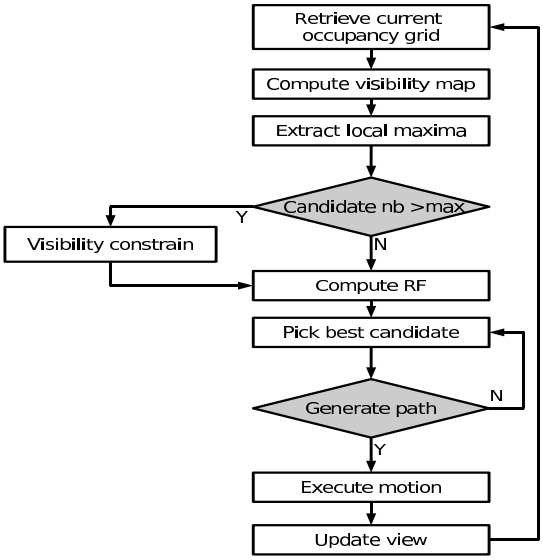



Figure 12.5: Flowchart of the next view selection

the local maxima extraction is performed providing the reduced configuration space $\tilde{\Omega}$. The formulation of the rating function, and the different steps of the next view selection are detailled in [51, 40].

## 12.4 Experiments

### 12.4.1 Simulation results

A full search behavior has been tested in simulation (Fig. 12.6). The environment is a 6x4x1.5 meter room with two obstacles, the target is hidden behind the large obstacle. A simulation of the recognition system has been implemented, although simple, it has the main characteristics of a real recognition system with false target detection that adds some noise to the probability map. In the simulation, the robot finds the object after 15 views. Depending on the settings (the $\lambda_{NI}/\lambda_{DP}$ ratio) the robot will lock the target after the first view or will do some remaining exploration before focusing its attention on the target.

### 12.4.2 Experimental setup and results

Real experiment using HRP-2 humanoid robot has been successfully achieved for an exploration experiment. The experimental room ( Picture in Fig. 12.7) has a size of 6 by 4 meters and is divided in two parts by a 2 meters wide panel. Fig. 12.8 shows an image sequence captured form the control interface during the experiment. The environment reconstruction is only based on disparity information and needs to

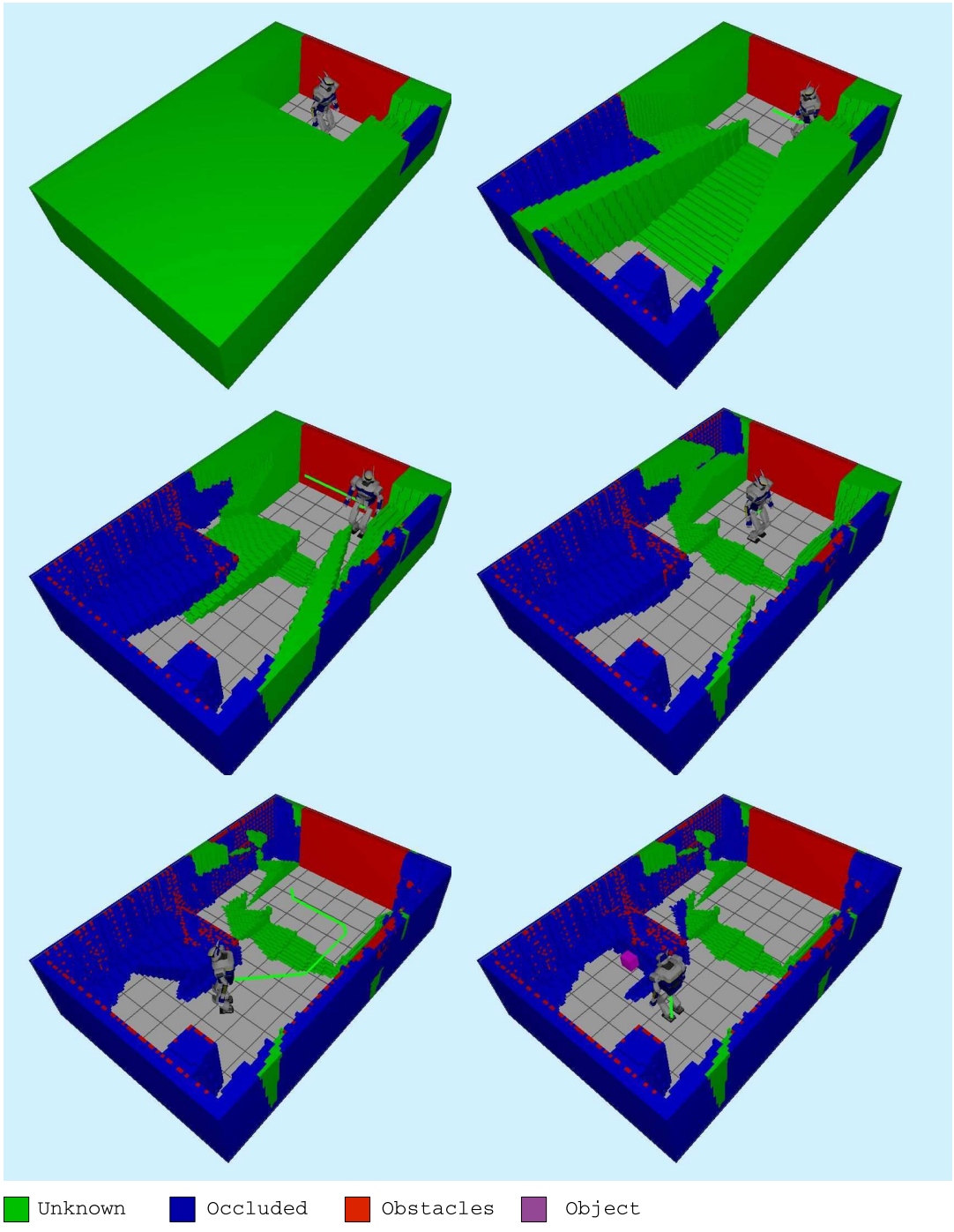



Figure 12.6: Image sequence of the search behavior, the object is hidden behind the large obstacle

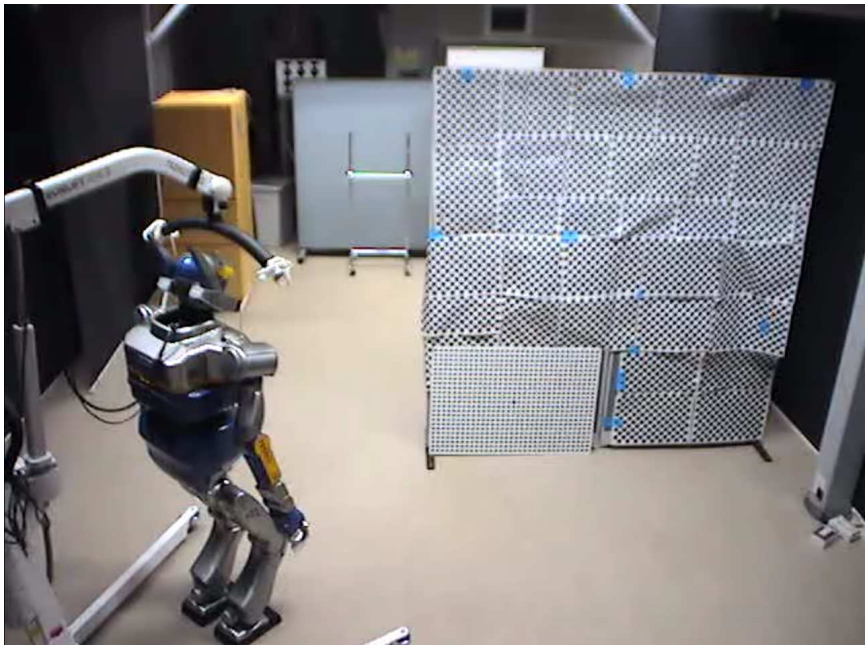


Figure 12.7: A picture of the experimental room during HRP-2 exploration

be well textured. The aim of the experiment we present here, was to have a full exploration and mapping of an unknown environment, in order to validate the model in a real world experiment in presence of heavy reconstruction noise and localization error. There was no target object hidden in the room, thus the planning was mainly driven by the new information retrieval even though the detection probability was taken into account in the optimization process. The whole exploration is done in 29 views, the robot finishes exploring the first part of the room in 23 views after mapping enough environment to allow a planning to explore behind the wall for the 6 remaining views.

## 12.5 Conclusion

We exposed a framework for a search behavior developed for the humanoid robot HRP-2. The problem, which falls in the sensor planning field, is formulated as an optimization problem. The concept of visibility map introduced in [51] to constrain the sensor parameter space according to the detection characteristics of the recognition algorithm is used to reduce the dimension of the sensor parameter space. Simulation and real experiments using the HRP-2 robot have been achieved to validate the proposed search model.

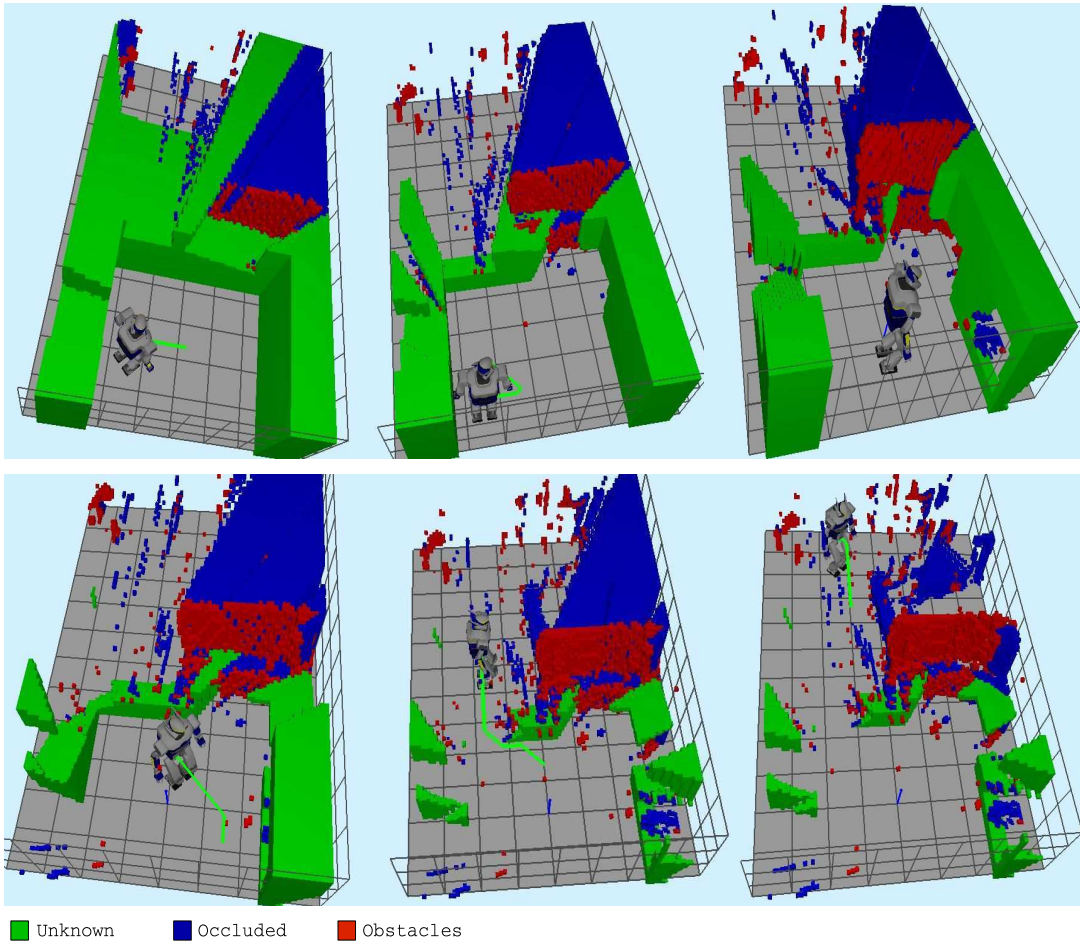



Figure 12.8: Images of the real environment exploration as seen through the control interface during the experiment. We notice the heavy reconstruction noise mainly due to false point matching

# Chapter 13

# Conclusion and future work

Figure 13.1: Human-Humanoid robot co-worker from the final demonstrator of ROBOT@CWE lead by A. Kheddar[Khe06]

This document stresses the contributions I have realized in the framework of visual-based behaviors for humanoid robots, and more specifically for the problem of treasure hunting. In this context the related cost function which evaluates the current level of success in realizing the behavior are formulated in chapter 12 and 11. Using the kinematic and dynamic model of the robot, as well as the mechanical criteria defining balance, the sub-problems related to real-time motion generation are described in chapter 4 and 5. Chapter 6 described techniques allowing extreme motions planning. Several of these techniques are real-time. Reaction time is important for humanoid robots evolving in partially structured and dynamic environment such as the ones where humans live. To adapt to such modification, the robot has to be able to perceive the information. Chapter 8 describes an original algebra formulation of

the triangulation problem to build a world representation. Chapter 9 describes a visual object model to detect far away objects. Both techniques however are too time consuming to be integrated in a continuous stream of information leading to an efficient execution of the behavior. On the contrary the method described in chapter 10 described a method to build a sparse map of the environment on the fly and in real-time. It loses however consistency over time. Most of the problem presented here are formulated as optimization problems. To solve them integrating physical constraints and properties help to decrease the size of the search space. This effect is generally called the *embodiment*, and plays a crucial role in chapters 4, 5 and 6. Still the search space is far too large for timely action-perception loop such as the one displayed in Fig. 13.1. Therefore it is necessary to approximate the feasible transitions between actions while coping with the complexity in time and in memory. In this manuscript, several examples using simplified models such as the inverted pendulum are presented. One other approach to build approximation of the feasible transitions is to use machine learning. We have use Support Vector Regression to learn the feasible foot-steps transitions [17]. However using machine learning can be extended to several aspects of the problems presented in the manuscript mostly to test quickly what kinds of actions the robot can do to realize a behavior (the *purpose*), and predict the evolution of the environment or the others actors involved in the behavior (the *situation*). In the light of this situated purposeful awareness, some keypoints of the contributions described in the manuscript are recalled as well as the perspectives of the specific related topic. From those key points an integrated project is given at the end of this chapter.

## 13.1 Motion generation

### 13.1.1 Whole-body motion generation

#### Synthesis

With Nicolas Mansard, we investigated the problem of generating articular values such that the humanoid robot HRP-2 would be dynamically balanced [9, 6], avoid (self-)collisions [34], while taking into account sensor-based input[38, 8], has been presented. Solving in real-time the related optimization problems has been the main goal of the work presented. This has been achieved by starting from the task function approach and consider their characteristics to remove and insert them in a stack [MC07] while avoiding discontinuities. The main interest is to be able to assemble basic controllers to build optimization problems allowing to generate whole body motion. This has been used successfully to generate motion between poses found by the algorithm presented in section 11 to build autonomously a model of an object.

#### Perspectives

The current state of the art in the most efficient scheme for the Generalized Inverse Kinematics running at 1 Khz, that can be found for instance in Kanehiro et al. [KMS+10] is mostly based on a formulation of the problem as a quadratic problem with inequalities at the higher level, having all the tasks at the second level, and a weight matrix to modulate the tasks between each other.

The main problem with this approach is two folds:

- When the tasks are incompatible between each other the solver will simply fail and the control output will not be computed.
- The weight matrix has to be set manually, and this calls for a profound knowledge of the system, as the linear system varies upon the robot state $\mathbf{J}(\mathbf{q})$.

Several problems are still open regarding real-time whole body motion:

- Removing and inserting new controllers is computationnally expensive when one wants to ensure continuity of the control law [KMWK11].
- Taking into account dynamic constraints at the level of forces with the complementary constraints [Man12] at a sufficient control rate.

### 13.1.2 Walking

#### Synthesis

As presented in chapter I, we focused mostly on the problem of generating the CoM trajectory. With the work done with Pierre-Brice Wieber, we have demonstrated that it is possible to guarantee that the trajectory is dynamically balanced. Indeed in the original proposition of Kajita [KKK+03a], such guarantee is not given. My personal contribution has been to show the validity of the concept on the humanoid robot HRP-2. To ease the specification of motion by the user, and increase the functional level of the pattern generator, Pierre-Brice Wieber with Andrei Herdt proposed to introduced the foot positions directly inside the related optimization problem [HPW10]. This has been made possible thanks to the new line of work developped with Nicolas Perrin consisting in finding in a systematic way the feasibility boundaries of a walking pattern generator. Using this, in [HPW10] a convex set of constraints on the footsteps positions has been proposed. With Claire Dune we have proposed [19] a new control scheme based on this pattern generator to generate real-time walking gaits using only the desired image of an object (using edges).

In [KKK+03a], a very important scheme called the dynamical filter is introduced to compensate for the inertia introduced by the remaining part of the robot body. We

have used this scheme to propose a real-time stepping over motion generator pushing the boundaries of the current robot capabilities [6].

We have also used real-time pattern generators coupled with the Stack of Tasks to realized several experiments of direct interactions between a humanoid robot and human [24, 87] as depicted in Fig.5.6.

### Perspectives

Being able to generate in real-time whole-body motion is still an open issue. It is particulary difficult with humans, or based on sensors, because the future is not known. The local scheme presented in [HPW10, 19] is very efficient in open space, but will fail in complex environments. It is an alternative to the approach developed with Nicolas Perrin on reactive motion planning. More specifically with walking pattern generator, the open problems lies on the following topics:

- **Non-linear optimization problem and solvers**. The continuation of the work introduced with Claire Dune criteria with higher semantic than speed. We could for instance try to minimize reconstruction function as the one introduced in section 11 directly at the control level. The main technical challenge consists in introducing non-linearity in the cost function. As the solvers for non-linear optimization problem is using Quadratic program, the work done on the quadratic problem is a necessary step towards more generic function to minimize.

- **Whole body motion and preview window**. Most of the techniques dealing with balance control use a preview window on the near future. Instead of using the dynamical filter technique described in paragraph 3.3, one could use directly the dynamical model of the robot. The main problem is again the time needed to find a solution.

- **Multiple-contacts**. Even if we can already formulate balance criteria more advanced than the Center of Pressure [ORH11] the main difficulty lies on the way we can generate a trajectory for the Center of Mass [1] and then for the whole-body. The main technical lock is to find solver fast enough to solve the related problem at the frequency necessary to cope with dynamical motion (typically 1 $KHz$).

[1] The dynamic of the robot can be projected on any point, but the Center of Mass make much more sense according to chapter I

## 13.2 Vision for humanoids

### 13.2.1 Visual object model

#### Synthesis

In this manuscript, we have presented various problems and their solutions related to visual object model:

- **The problem of triangulation and its error model**. In general this problem is ill-posed, because two semi-infinite line rarely intersects. Thus one has to build a distance between two such semi-infinite lines to find the most probable position of a point projected on two images. With Benoit Telle, in section 8 we have proposed a model which considers two cones intersecting, and gives the space in which the two cones intersect.
- **Apparence model based**. In order to build a model of an object, the classical approach is to use a set of features, typically SIFT, representing local information and to build the geometrical relationship between each of these features. This model however is not enough when we want to elect candidates for further examination, as for an object of 20 to 30 $cm$ the object can not be recognize clearly beyond 2 $m$. With Diane Larlus, in section 9, we have shown that such limitation can be overcome when using generative probabilistic framework based on a model using bag-of-words, disparity and local apparence relationships. We were then able to detect objects up to 6 $m$.

#### Perspectives

Since the past decade, the computer vision community has shifted massively towards the use of machine learning techniques to represent visual model of object categories [SZ03]. Such models allows a more robust and high level perception of the world, but at a very high computation cost. Recently the introduction of the Kinect, coupled with GPU, is allowing to build in real-time the environment 3D model[NIH+11]. Such information related with apparence based description for detecting, or/and to perform real-time tracking could lead to major breakthrough in humanoid robotic more specifically helping humanoid robot to directly interact with its surroundings.

### 13.2.2 Self Localization Map Building

The Self Localization and Map Building community is currently very active, and it would be difficult to give an accurate overview of the field. For this reason we will focus on the aspect directly related to humanoid robotics.

#### Synthesis

With Andrew Davision, we were the first team to propose a full visual based real-time 3D slam framework using only one camera and an Inertial Measurement Unit [10]. The system is based on an Extended Kalman Filter which takes into account the position, the orientation and the position of the landmarks (the map). The whole framework has been instanciated on the humanoid robot HRP-2 and has shown the interest of loop-closing to decrease the uncertainities related to the landmarks [44].

#### Perspectives

Since this seminal work, one of the major breakthrough in SLAM has been to introduce Structure-From-Motion techniques into the process of map-building [KM07]. This technique imposes the projective geometric constraints upon the collected data. It is realized by minimizing the error of the landmarks projection in the stereoscopic images according to the camera position and the position measured by data association. This improves drastically the quality of the map built. It avoids all the problems due to the linearization related to EKF [SMD12a]. Together with Pablo Alcantarilla we have tested a localization system based on a dense map using such techniques [1]. It also includes a visibility prediction which suppresses landmarks which are not present anymore in the map (for instance when an object has been moved). There is still some open issues with the quality of the map which may vary greatly depending on the nature of the environment and the landmarks which can be extracted. For instance, when facing a corner with a uniform color it might be very difficult for the robot to localize itself.

### 13.2.3 Autonomous 3D object model building

#### Synthesis

With Torea Foissotte, we have proposed a method to automatically found feasible poses for a humanoid robot to reconstruct the visual model of an object [5]. The criteria takes into account the visibility of the previously seen landmarks, the capabilities of the vision system, and the unknown part of the object. Building a representation of the unknown which has a continuous derivative is unfornatunately quite difficult. To overcome this problem we have proposed a method based on BobbyQ a method introduced by Powell to build convex local approximation of a function and to find the related local minima. On small models (around 20 to 30 $cm$), the number of points found to cover the overall object is coherent with the work of Claire Dune [Dun09]. In [Dun09], if an object is represented by its bounding ellipsoid, the covering number of points is four which correspond to each quadrant.

#### Perspectives

The main problem with this experiment is the execution time. The sequence of planning and blind execution could be definitely improved by using a visual-servoing approach as the one proposed in [Dun09], inside the stack of tasks. However there is no guarantee that the solution found would not be a local minimum. In order to deal with this problem, the next line of work is to couple such minimization with the planning algorithm developed with Nicolas Perrin [4]

### 13.2.4 Sensor planning for Visual Search with a Humanoid Robot

#### Synthesis

With François Saïdi, we have proposed a method [13, 40, 39] to search for objects based on a computer vision model. The method is integrating a model of object detection, a camera model with its contraints, a cost related to the motion, the robot kinematic constraints and obstacle avoidance. The integration of all such constraint allows to drastically decrease the search space, and provides solutions working in real-time. This has been tested on the humanoid robot HRP-2 together with a stereo reconstruction of the environment using a classical occupancy grid.

#### Perspectives

This problem falls in the frame of sensory planning in the robotic community and the first-passage time for target search in physics [CBT+07]. Since our proposal a new framework based upon a Bayesian formulation has been proposed by Andreopoulos et al. [AHW+11]. This work is interesting in its unifying formalization of the sensor model, recognition, and the internal representation of the world. Still it considers a limited set of actions of the robot, where as we know that the robot could make contact with the environment [BELK09], or even interact to explore for instance drawers in furniture. Integrating the large set of motions that a humanoid robot is able to perform will increase the interest of such behaviour. We hope to continue in the line of the work started with Nicolas Perrin [4] to make humanoid robots search in complex changing environments.

## 13.3 A global perspective

### 13.3.1 Key challenges

One of the key challenge for humanoid robots in the next decade is to realize natural collaborative tasks between a human person and a humanoid robot. For this, it

will be necessary to bind perception, control and planning through machine learning techniques while considering the context and specific behaviour. This can be done by incorporating the purpose of a task into methods developed for situational awareness and resulting in novel strategies to deal with the high number of degrees of freedom existing in humanoids. This imply to develop techniques to make real time decisions for generating whole body motions and push the envelope of the current limits in motion control for humanoid robotics. Recent developments in humanoid robotics such as Asimo [Hon11] or Justin [OEF+06] show impressive dynamic and dexterous capabilities. They can pour water in glasses, open pots but they can also run, or jump on their feet. Such token performances slowly expand the range of motion of humanoid robots, and seemingly take them closer to a servant being able to help human beings in daily life. However increasing the possibilities achievable by humanoid robots has a serious cost. The set of feasible actions by a humanoid robot increases dramatically with the complexity of the task, and makes the problem of planning motion to perform more involved behaviours increasingly difficult. This becomes even more complicated when interacting with a person because the robot has to generate complex whole-body motions in real-time. In order to reduce drastically this complexity, we aim at integrating as much as possible meaningful constraints on the overall behaviour. Therefore it is necessary to try to develop methodologies to extract such constraints from a realistic scenario given by industrial applications. As an example demonstrator, we can consider the task where a human and a humanoid robot are mounting together a window frame on a wall, a typical construction task where a range of timely decisions, elaborated perception and complex motions are required. The high-level constraints to extract are then the following: the purpose, the situation, and the embodiment. By taking into account such constraints together, we can decrease the complexity of motion generation in interaction, and break current humanoid robot technology limitations. In the previous chapters of the manuscript, preliminary work in this direction show that this approach is both realistic and promising. The main inspiration of this approch is the fact that humans can take a decision in a blink of an eye while being in a complex situation. For instance firemen during an intervention on a building on fire can decide suddenly to evacuate a room due to the perception of a dangerous situation. Soccer players during a game can synthetize the overall situation and decide to perform a bicycle kick. Important to note is that such complex motion decisions are not scripted but taken in real-time and are the result of experience and intense training. Fig. 13.3.1 synthesizes the reduction of the space of trajectories that can be expected with this approach.

### 13.3.2 Embodiment

The embodiment is defined here by the physical characteristics of the robot, and the feasible control architecture. Let us take for instance the humanoid robot HRP-2 available at LAAS. When considering only the dynamic model of the robot, one could expect to see this robot running. However due to the mechanical limitation of its actuators, this robot is known to be unable to run. On the other hand recent work

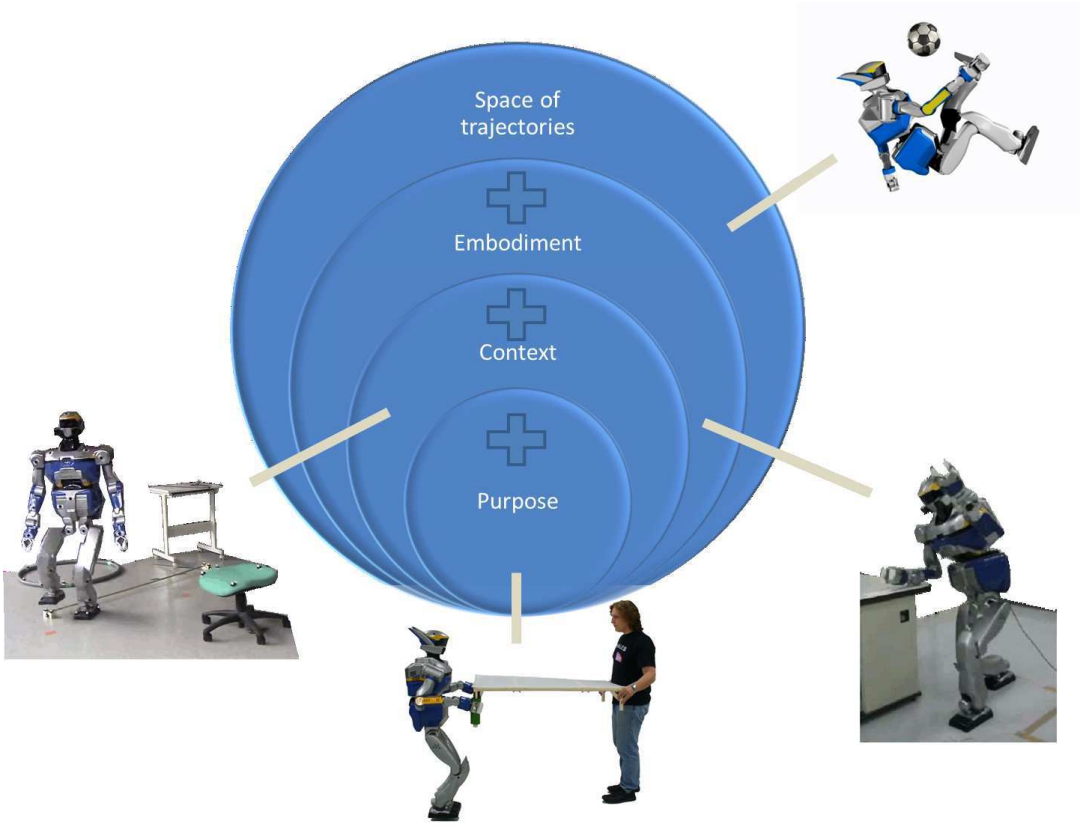



Figure 13.2: Reducing the possible space of trajectories to explore using meaningful constraints. (a) Integrating the characteristics of the robot mechanics and control, here the robot is using its capabilities to be in contact with a furniture (b) the context, here the robot evolves on a planar surface with known moving objects (c) the purpose, here a human and a robot hold together a table . The last result was obtained in the frame of the FP6 ROBOT@CWE project. Because the robot does not known the context and its purpose it is not using his full capabilites (walking speed at 0.1 m/s instead of 0.4 m/s)

on multiple-contacts planning validated on this robot has demonstrated a wide range of possible motions [EK10] [HBHL08]. Still such motions are currently unfeasible at the control level in general. Indeed an algorithm able to solve the related optimization problem in real-time has yet to be found. Therefore even if such motion is feasible on the real robot, it can not be used in a control loop. For the reasons mentioned above, to stay in the space of feasible motions, the approch I am developing is to plan motion directly in the tasks space of the robot. Current state-of-the-art in control architecture ( [SK06, MC07, GJG06]) used in the context of humanoid robotics, are based on the resolution of optimization problems including several tasks at the same time (also called here the Stack of Tasks). For instance, to reach a tool on a table, such problem will involve a task related to balance, a task related to the end effector, and one task related to perception (for visual-servoing). Moving for a purpose will therefore consists in realizing a sequence of such a stack of tasks. Tasks are intrinsically robust and allow coping with sensory feedback. However they can fail in two cases:

- when optimization gets stuck in a local minimum,
- when the tasks become incompatibles.

When such problem occurs due to the intrinsic limitation of the robot, (joint limits, self-collision, and balance) it appears natural to explore in advance such limitations in order to avoid them during execution. In [4] this approach has been used to plan foot-steps in real-time in constrained environments. Extending this to whole body motion will call for efficient representations in high dimensional space using stochastic models. In the case of the robot handling a window frame with a human, this should help us to avoid self-collision, and kinematic singularities (like for instance when the arms are extended) when taking into account small modifications of the dynamic environment. When only the robot is involved, this approach can also works with more drastic change of the environment, but for this will not be enough. Prediction is a crucial issue. Indeed one very important constraint with humanoid robot is balance. Real-time state-of-the art methods use a preview window, such as the ones presented in chapter.4, because dynamic motion implies that the future has to be taken into account. Therefore knowledge on the immediate future is of paramount importance in decision making related to motion generation for humanoid robots. Traditionally this preview window includes the foot-steps, and is used to define the Center-Of-Mass (CoM) trajectory of the humanoid robot to keep its balance. By including the feasible limitations described previously, it has been possible to expand this traditional scheme and to automatically find foot-steps at the control level [HPW10]. This simplifies drastically motion planning when no obstacle is present. Indeed the reference given to the system is not foot-steps anymore but a velocity of the CoM. The following step is to develop locomotion controllers with higher semantic inputs. Currently the classical input of walking system for a humanoid robot is foot-steps. It is possible to go beyond this by including, for instance, directly visual perception in order to decrease the complexity of motion planning [19]. In the case of the robot handling a window frame with the human, the robot will automatically generate its foot-steps

using the information provided by computer vision to perceive the spatial context and the force information from its wrists.

### 13.3.3 Situation

The situation is the environment surrounding the robot. It includes the buildings, the objects and the other agents such as the collaborator person. In order to realize a behaviour the robot will have to perceive the state of the environment to take the appropriate action. Such information can be acquired through vision, force sensors and a 3D capture system such as a ViCon. In addition to learning the appearance of the objects, it will be interesting to develop techniques to learn the spatial relation between the objects and the human agent. For instance, in order to locate the robot in a building site, Self Localization and Map Building techniques able to deal with environments where strong modifications occur are necessary. Building sites includes a high variability, offering a very interesting setup. Starting form the current well-established body of work in Simultaneous Localization And Mapping (SLAM) it is necessary to propose new algorithms to deal with dynamic environments. The second technique to relate objects with space can consist in using gaze tracking of the collaborator and of experts performing the task from where we can learn eye-object interactions. This will make possible to relate the spatial position of the human with the objects as well as the sequence of inspections that an expert user performs before proceeding to manipulation.

### 13.3.4 Purpose

For the purpose, it is possible to focus on relatively basic behaviours which do not imply complex abstract inferences from the robot, but which involves solving complex motion generation problems under hard-real time constraints. A target is to be able to handle the inherent variability of reality. More specifically assembling windows in collaboration between a human and a humanoid robot on a construction site offers such a variability. The extraction of the constraints at this stage will consist in two parts: building a representation of the human state along the behaviour realization and the sequence of controllers to execute by the humanoid robot. The latter is currently almost intractable when no a priori information is given. For this reason we can use information given on the task to segment the overall behaviour in sub-problems. In order to handle the variability around such sub-problems, three approaches can be developed to build a stochastic model. The first one is based on sample problems taken from measurements on sites. Using models of buildings, frames used in the real sites, together with the objects present on the spot, it is possible to try to solve this problem using motion planning techniques with no-prior knowledge provided by a human-human sample solution. Then there is a need to develop stochastic models relating the sample problems with their solution. The stochastic model can be used to instantiate the sequence of controllers needed. Typically in the construction site

scenario, the robot is constantly maintainning the most likely future, and plan, according to this prediction, the most relevant tasks to execute. Apart from the core motion planning strategy mainly used for illustration above, two further profitable approaches can be considered: the second approach consist in extracting the solution not from motion planning but from the interaction between two expert humans, and then to map the solutions on the robot, the third related approach is to use kinaesthetic demonstrations. In all cases, the main point is to extract the set of controllers to be used, the reference trajectories and the conditions upon which the controllers or the reference trajectories are changed. This aspect is fundamental. Indeed describing the set of trajectories feasible by the robot purely from its model and its control architecture is currently impossible due to the combinatorial explosion. Finding such trajectories using randomized techniques while following the sub-manifolds described by the tasks is still a complex problem which takes times to be solved. The recent breakthroughs in the field of programming by demonstration bear a high potential. By approximating the flow-fields, generated otherwise by the tasks, they provide the desired flexibility which copes with the variability of reality. Current techniques learn at the same time the tasks, the constraints and the reference trajectories. The latter is usually specific to the behaviour (purpose) while the others are related to the robot model and limits (embodiment). Recent breakthroughs in deep learning [SLY11] is a promising way to deal with the inherent combinatorial explosion when trying to learn complex behaviours. The trajectories and their characteristics will also be the corner stone of the information to decide when and which the transition between two stacks of tasks should occur.

# Chapter 14

# Publications

## 14.1 Peer reviewed Journals


[1] P.F. Alcantarilla, O. Stasse, S. Druon, L. M. Bergasa, and F. Dellaert. How to localize humanoids with a single camera ? *Autonomous Robots*, 34:47–71, 2013.

[2] S. Hak, N. Mansard, O. Stasse, and J.-P. Laumond. Reverse control for humanoid robot task recognition. *IEEE Transactions on System, Man and Cybernetics - part B: Cybernetics*, page accepted, 2012.

[3] J.-B. Hayet, C. Esteves, G. Arechavaleta, O. Stasse, and E. Yoshida. Humanoid locomotion planning for visually-guided tasks. *International Journal of Humanoid Robotics*, 9(2):accepted, 2012.

[4] N. Perrin, O. Stasse, L. Baudouin, F. Lamiraux, and E. Yoshida. Fast humanoid robot collision-free footstep planning using swept volume approximations. *IEEE Transactions on Robotics*, 28(2):427–439, 2012.

[5] T. Foissotte, O. Stasse, P.-B. Wieber, A. Escande, and A Kheddar. Autonomous 3d object modeling by a humanoid using an optimization-driven next-best view formulation. *International Journal on Humanoid Robotics, Special issue on Cognitive Humanoid Vision*, 7(3):407–428, 2010.

[6] O . Stasse, B. Verrelst, B. Vanderborght, and K. Yokoi. Strategies for humanoid robots to dynamically walk over large obstacles. *IEEE Transactions on Robotics*, 25:960–967, 2009.

[7] O. Stasse, R.Ruland, F. Lamiraux, A. Kheddar, K. Yokoi, and W. Prinz. Integration of humanoid robots in collaborative working environment: A case study on motion generation. *Intelligent Service Robotics*, 2(3):153–, 2009.

[8] O. Stasse, B. Verrelst, A. Davison, N. Mansard, F. Saïdi, B. Vanderborght, C. Esteves, and K. Yokoi. Integrating walking and vision to increase humanoid

autonomy. *International Journal of Humanoid Robotics,special issue on Cognitive Humanoid Robots*, 5(2):287–310, 2008.

[9] O. Stasse, B. Verrelst, P.-B. Wieber, B. Vanderborght, P. Evrard, A. Kheddar, and K. Yokoi. Modular architecture for humanoid walking pattern prototyping and experiments. *Advanced Robotics, Special Issue on Middleware for Robotics –Software and Hardware Module in Robotics System*, 22(6):589–611, 2008.

[10] A. J. Davison, I. Reid, N. Molton, and O. Stasse. Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, June 2007.

## 14.2 Book chapters

[11] D. Dimitrov, P-B.Wieber, O. Stasse, H.J. Ferreau, and H. Diedam. *Recent Trends in Optimization and its Applications in Engineering, Springer Verlag*, chapter An Optimized Linear Model Predictive Control Solver, pages 309–319. 2010.

[12] V. Nunez, N. Nadjar-Gauthier, K. Yokoi, P. Blazevic, and O. Stasse. *Humanoid Robots Human-like Machines, ISBN978-3-902613-07-3*, chapter Inertial Forces Posture Control for Humanoid Robots Locomotion, pages 175–189. IN-Tech, 2007.

[13] F. Saïdi, O. Stasse, and K. Yokoi. *Recent Progress in Robotics Viable Robotic Service to Human*, chapter Active Visual Search by a Humanoid Robot, pages 171–184. Springer-Verlag, 2007.

## 14.3 International Conference with peer review

[14] T. Moulard, F. Lamiraux, and O. Stasse. Trajectory following for legged robots. In *IEEE Int. Conf. on Biomedical Robotics and Biomechatronics BioRob)*, pages 4243–4248, 2012.

[15] N. Perrin, O. Stasse, F. Lamiraux, Y. J. Kim, and D. Manocha. Real-time footstep planning for humanoid robots among 3d obstacles using a hybrid bounding box. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, pages 977–982, 2012.

[16] L. Baudouin, T. Moulard, N. Perrin, O. Stasse, F. Lamiraux, and E. Yoshida. Real-time replanning using 3d environment for humanoid robot. In *IEEE/RAS Int. Conf. on Humanoid Robotics (ICHR)*, pages 584–589, 2011.

[17] F. Lamiraux N. Perrin, O. Stasse and E. Yoshida. A biped walking pattern generator based on half-steps for dimensionality reduction. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, pages 1270–1275, 2011.

[18] F. Lamiraux N. Perrin, O. Stasse and E. Yoshida. Weakly collision-free paths for continuous humanoid footstep planning. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 4408–4413, 2011.

[19] C. Dune, A. Herdt, O. Stasse, and K. Yokoi P.-B. Wieber. Visual servoing of dynamic walking motion by ignoring the sway motion. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 3175–3180, 2010.

[20] S. Hak, N. Mansard, and O. Stasse. Humanoid robot task recognition from movement analysis. In *IEEE/RAS Int. Conf. on Humanoid Robotics (ICHR)*, pages 314–321, 2010.

[21] N. Perrin, O. Stasse, and F. Lamiraux. Approximation of feasibility tests for reactive walk on hrp-2. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, pages 4243–4248, 2010.

[22] N. Kwak, K. Yokoi, O. Stasse, and T. Foissotte. Humanoid feet trajectory generation for the reduction of the dynamical effects. In *IEEE/RAS Int. Conf. on Humanoid Robotics (ICHR)*, pages 62–67, December 2009.

[23] P. Pierro, O. Stasse, K. Yokoi, C. Balaguer, and A. Kheddar. Humanoid feet trajectory generation for the reduction of the dynamical effects. In *IEEE/RAS Int. Conf. on Humanoid Robotics (ICHR)*, pages 454–458, December 2009.

[24] O. Stasse, P. Evrard, N. Perrin, N. Mansard, and A. Kheddar. Fast foot prints replanning and generation during walking in physical human-humanoid interaction. In *IEEE/RAS Int. Conf. on Humanoid Robotics (ICHR)*, pages 284–289, December 2009.

[25] T. Foissotte, O. Stasse, P.B. Wieber, and A. Kheddar. Using newuoa to drive the autonomous visual modeling of an object by a humanoid robot. In *IEEE Conference on Information and Automation (ICIA)*, pages 78–83, June 2009.

[26] 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 *International Conference on Advanced Robotics (ICAR)*, page 119, June 2009.

[27] D. Dimitrov, P.-B. Wieber, H. Diedam, and O. Stasse. On the application of linear model predictive control for walking pattern generation in the presence of strong disturbances. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, pages 1171–1176, May 2009.

[28] T. Foissotte, O. Stasse, A. Escande, P.-B. Wieber, and A. Kheddar. A two-steps next-best-view algorithm for autonomous 3d object modeling by a humanoid robot. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, pages 1159–1164, May 2009.

[29] P. Evrard, N. Mansard, O. Stasse, A. Kheddar, T. Schauss, C. Weber, A. Peer, and M. Buss. Intercontinental, multimodal, wide-range tele-cooperation using a humanoid robot. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2009.

[30] T. Foissotte O. Stasse, D. Larlus, A. Kheddar, and K. Yokoi. Treasure hunting for humanoid robots. In *Workshop on Cognitive Humanoids Vision,8th IEEE-RAS International Conference on Humanoid Robots*, 2009.

[31] O. Stasse, N. Perrin, P.-B. Wieber, N. Mansard, and F. Lamiraux. Ieee ro-man workshop on robot-human synergies. 2009.

[32] T. Foissotte, O. Stasse, A. Escande, and A. Kheddar. A next-best-view algorithm for autonomous 3d object modeling by a humanoid robot. In *8th IEEE-RAS International Conference on Humanoid Robots*, pages 333–338, 2008.

[33] A. Peer, S. Hirche, C. Weber, I. Krause, M. Buss, S. Miossec, P. Evrard, O. Stasse, Ee S. N., A. Kheddar, and K. Yokoi. Intercontinental multimodal tele-cooperation using a humanoid robot. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 405–411, 2008.

[34] O. Stasse, A. Escande, N. Mansard, S. Miossec, P. Evrard, and A. Kheddar. Real-time (self)-collision avoidance task on a hrp-2 humanoid robot. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, pages 3200–3205, 2008.

[35] O. Stasse, F. Lamiraux, A. Kheddar, K. Yokoi, R. Ruland, and W. Prinz. Integration of humanoid robots in collaborative working environment: A case study on motion generation. In *International Conference on Ubiquitious Robots and Ambient Intelligence*, pages 211–216, 2008.

[36] O. Stasse, Ee S. N., F. Lamiraux, T. Sakaguchi, A. Kheddar, and K. Yoikoi. Architectures and models for humanoid robots in collaborative working environments. In *International Symposium on Robotics*, pages 354–358, 2008.

[37] F. Saïdi, O. Stasse, and K. Yokoi. Active visual search by a humanoid robot. In *International Conference on Advanced Robotics*, pages 360–365, August 2007. Finalist for the best paper award.

[38] N. Mansard, O. Stasse, F. Chaumette, and K. Yokoi. Visually-guided grasping while walking on a humanoid robot. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, pages 3041–3047, 2007.

[39] F. Saïdi, O. Stasse, and K. Yokoi. The visibility map, a constraint for an active visual search by a humanoid robot. In *IFAC,Symposium on Intelligent Autonomous Vehicles*, pages Reference: p99, CD–ROM proceedings, 2007.

[40] F. Saïdi, O. Stasse, K. Yokoi, and F. Kanehiro. Online object search with a humanoid robot. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1677–1682, 2007.

[41] O. Stasse, D. Larlus, B. Lagarde, A. Escande, F. Saïdi, A. Kheddar, K. Yokoi, and F. Jurie. Towards autonomous object reconstruction for visual search by the humanoid robot hrp-2. In *IEEE RAS/RSJ Conference on Humanoids Robots, Pittsburg, USA, 30 Nov. - 2 Dec.*, pages 151–158, 2007.

[42] E. Yoshida, A. Mallet, F. Lamiraux, O. Kanoun, O. Stasse, M. Poirier, P.-F. Dominey, J.-P. Laumond, and K. Yokoi. Give me the purple ball – he said to hrp-2 n.14. In *IEEE RAS/RSJ Conference on Humanoids Robots*, pages 89–95, 2007.

[43] R. Sellaouti, O. Stasse, S. Kajita, K. Yokoi, and A. Kheddar. Faster and smoother walking of humanoid hrp-2 with passive toe joints. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 2955–2960, October 2006.

[44] O. Stasse, A. Davison, R. Sellaouti, and K. Yokoi. Real-time 3d slam for humanoid robot considering pattern generator information. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 348–355, October 2006.

[45] O. Stasse, Sylvain Dupitier, and K. Yokoi. 3d object recognition using spin-images for a humanoid stereoscopic vision system. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 2955–2960, October 2006.

[46] B. Verrelst, B. Vanderborght, O. Stasse, and K. Yokoi. Stepping over large obstacles by the humanoid robot hrp-2. In *CLAWAR 2006, 9th International Conference on Climbing and Walking Robots, Brussels, BELGIUM*, September 2006.

[47] K. Yokoi, Neo Ee Sian, T. Sakaguchi, O. Stasse, Y. Kawai, and K. Maruyama. Humanoid robot hrp-2 with human supervision. In *International Symposium on Experimental Robotics, ISER, Rio de Janeiro, Brazil*, July 2006.

[48] B. Verrelst, K. Yokoi, O. Stasse, H. Arisumi, and B. Vanderborght. Mobility of humanoids robots: Stepping over large obstacles dynamically. In *International Conference on Mechatronics and Automation, ICMA, LuoYang - Henan, China*, pages 1072–1079, June 2006. Best Conference Paper Award.

[49] O. Stasse, J. Semere, Neo E. S., T. Yoshimi, and K. Yokoi. Vision-based virtual information and semi-autonomous behaviours for a humanoid robot. In *International Conference on Intelligent Autonomous Systems*, Tokyo, Japan, March 7-9 2006. IAS.

[50] V. Nunez, N. Nadjar-Gauthier, K. Yokoi, P. Blazevic, and O. Stasse. Whole body posture controller based on inertial forces. In *IEEE/RAS Int. Conf. on Humanoid Robotics (ICHR)*, pages 188–193, 2006.

[51] F. Saïdi, O. Stasse, and K. Yokoi. A visual attention framework for search behavior by a humanoid robot. In *IEEE/RAS Int. Conf. on Humanoid Robotics (ICHR)*, pages 346–351. IEEE, 2006.

[52] Neo Ee Sian, O. Stasse, Y. Kawai, and T. Sakaguchi. A unifed on-line operation interface for humanoid robots in a partially-unknown environment. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, 2006.

[53] B. Verrelst, O. Stasse, K. Yokoi, and B. Vanderborght. Dynamically stepping over obstacles by the humanoid robot hrp-2. In *IEEE/RAS Int. Conf. on Humanoid Robotics (ICHR)*, pages 117–123. IEEE, 2006.

[54] A. Kheddar, O. Stasse, Neo Ee Sian, and K. Yokoi. A visio-haptic percept using guarded motion and augmented reality occlusion checking for humanoid self-body extraction and primary touch function. In *International Symposium on Robotics, ISR*, Tokyo, Japan, November 29 - December 1 2005. IEEE.

[55] K. Yokoi, Neo Ee Sian, T. Sakaguchi, H. Arisumi, E. Yoshida, O. Stasse, Y. Kawai, K. Maruyama, T. Yoshimi, and S. Kajita. Humanoid robot hrp-2 no.10 with human supervision. In *International Symposium on Robotics, ISR*, Tokyo, Japan, November 29 - December 1 2005. IEEE.

[56] Y. Guan, K. Yokoi, O. Stasse, and A. Kheddar. On robotic trajectory using polynomial interpolations. In *International Conference on Robotics and Biomimetics, ROBIO*, pages 284–289, Hong Kong, China, June 29- July 03 2005. IEEE.

[57] O. Stasse, B. Telle, and K. Yokoi. 3d segmentation using interval analysis and pre-attentive behaviour for a humanoid robot. In *International Conference on Robotics and Biomimetics, ROBIO 2005*, pages 284–289, Hong Kong, China, June 29 - July 03 2005. IEEE.

[58] O. Stasse, Neo Ee Sian, K. Yokoi, Gabriel Dauphin, and P. Bonnin. Fast quality measurement of a h263+ video stream for teleoperating a hrp-2 humanoid robot. In *Machine Vision and its Applications, MVA*, pages 522–525, Tsukuba, Japan, May 16–18 2005.

[59] A. J. Davison, O. Stasse, and K. Yokoi. Vision based slam for a humanoid robot. In *International Conference on Robotics and Automation, SLAM Workshop*, pages 18–22, Barcelona, Spain, April 18-22 2005. IEEE.

[60] B. Telle, O. Stasse, T. Ueshiba, K. Yokoi, and F. Tomita. Three characterisations of 3d reconstruction uncertainty with bounded error. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, pages 3905–3910, Barcelona, Spain, April 18-22 2005. IEEE.

[61] B. Telle, O. Stasse, T. Ueshiba, K. Yokoi, and F. Tomita. 3d boundaries partial representation of objects using interval analysis. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 4013–4018, Sendai, Japan, November 2004. IEEE, RSJ.

## 14.4 International Conference with peer review before JRL

[62] O. Stasse and L. Laval. Robot trajectories generation: using a chaotic oscillator as computational resource. In *7th International IFAC Symposium on Robot Control - SYROCO 2003.*, Wroclaw, Poland, September 1-3 2003.

[63] V. Hugel, A. Abourachid, H. Gioanni, M. Maurice, O. Stasse, P. Bonnin, and P. Blazevic. The robocoq project : Modelling and design of a bird-like robot equipped with stabilized vision. In *AMAM2003, 2nd International Symposium on Adaptive Motion of Animals and Machines*, Kyoto, Japan, March 2003.

[64] V. Hugel, P. Blazevic, O. Stasse, and P. Bonnin. Trot gait design details for quadrupeds. In *RoboCup 2003: Robot Soccer World Cup VII 2003,*, volume LNAI 3020 of *Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence)*, pages 495–502. Springer-Verlag, 2003.

[65] P. Bonnin, O. Stasse, V. Hugel, P. Blazevic, N.M'Sirdi, and P. Coiffet. How to extract and to exploit vision data for autonomous and mobile robots to operate in known environments. In *10th International Workshop in Robot and Human Communication, ROMAN 01*, pages 231–236, Bordeaux Paris, September 18-21 2001. IEEE.

[66] O. Stasse, Y. Ishiwata, and Y. Kuniyoshi. Frame rate distributed computing for log-polar images with a novel real-time operating system on a general purpose platform. In *6th Int. Symp.on Signal Processing and its Application ISSPA*, pages 206–209, Kuala Lumpur Malaysia, August 13-16 2001. IEEE.

[67] P. Bonnin, O. Stasse, V. Hugel, and P. Blazevic. How to introduce a priori visual and behavioral knowledge for autonomous robots to operate in known environments. In *8th Int. Conf on Emerging Technologies and Factory Automation*, Antibes Juan les Pins, France, 15 -18 October 2001. IEEE.

[68] V. Hugel, O. Stasse, and P. Bonnin et P. Blazevic. French lrp team's description. In *RoboCup 2001: Robot Soccer World Cup V 2001 in Seattle*, Springer Verlag, Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence) LNAI 2377., pages 701–704. Springer-Verlag, 2001.

[69] Y. Kuniyoshi, S. Rougeaux, O. Stasse, G. Cheng, and A. Nagakubo. A humanoid vision system for versatile interaction. In *Biologically Motivated Computer Vision, LNCS 1811*, pages 512–526, Seoul, Korea, 2000.

[70] Y. Kuniyoshi, O. Stasse, Y. Ishiwata, G. Cheng, and A. Nagakubo (Workshop). Complex real world information processing for a humanoid robot. In *Sixth International Conference on Engineering of Complex Computer Systems*. IEEE, 2000.

[71] O. Stasse and Y. Kuniyoshi. Predn : Achieving efficiency and code reusability in a programming system for complex robotic applications. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, volume 1, pages 81–87. IEEE, 2000.

[72] O. Stasse, Y. Kuniyoshi, and G. Cheng. Development of a biologically inspired real-time visual attention system. In *Biologically Motivated Computer Vision, LNCS 1811*, pages 150–159. Seoul, Korea, 2000.

## 14.5 Workshops

[73] C. Dune, A. Herdt, E. Marchand, O. Stasse, P.-B. Wieber, and E. Yoshida. Vision based control for humanoid robots. In *IEEE/RAS International Conference on Intelligent Robot and Systems (IROS), Workshop on Visual Control of Mobile Robots (ViCoMor)*, 2011.

[74] N. Perrin, O. Stasse, L. Baudouin, F. Lamiraux, and E. Yoshida. Real-time replanning using 3d environment for humanoid robot. In *IEEE/RAS International Conference on Humanoid Robot (ICHR)*, pages 584–589, 2011.

[75] O. Stasse, T. Foissotte, D. Larlus, A. Kheddar, and K. Yokoi. Treasure hunting for humanoid robots. In *Workshop on Cognitive Humanoids Vision, 8th IEEE-RAS International Conference on Humanoid Robots*, 2009.

[76] Andrew J. Davison, Olivier Stasse, and Kazuhito Yokoi. Vision based slam for a humanoid robot. In *International Conference on Robotics and Automation, ICRA Barcelona, Spain, Workshop*, April 18-22 2005.

## 14.6 National Conference with peer review

[77] N. Mansard, O. Stasse, F. Chaumette, and K. Yokoi. Saisie guidée par la vision pendant la marche d' un robot humanoide. In *2de Journée Nationnale de la Robotique Humanoide, JNRH 07, Montpellier, France*, 2007.

[78] F. Saïdi, O. Stasse, and K. Yokoi. A visual attention framework for search behavior by a humanoid robot. In *National Conference of the Robotic Society of Japan*, September 2006.

[79] O. Stasse, G. Dauphin, and P. Bonnin. New image quality measurement of a h263+ video stream for teleoperation. In *National Conference of the Robotic Society of Japan*. RSJ, 2004.

## 14.7 Reports

[80] L. Blasi and O. Stasse. Deliverable 3.3@m22: Design support software for robot@cwe. Technical report, ROBOT@CWE - Strep project (FP6 framework), 2008.

[81] O. Stasse, P. Pierro, and A. Kheddar. Deliverable 2.2/3@m22: Architectures and models of ist-robotics: Different integration models of collaborative work in different real environments. Technical report, ROBOT@CWE - Strep project (FP6 framework), 2008.

[82] M. Arbulu, C. Balaguer, A. Bauer, P. Evrard, A. Kheddar, C. Perez, P. Pierro, N. Mansard, S. Miossec, O. Stasse, K. Yokoi, and E. Yoshida. Ist-robots concepts and preliminary integration. Technical report, ROBOT@CWE - Strep project (FP6 framework), 2007.

[83] O. Stasse. Predn user's manual. Technical report, RNTL Projet Cleopatre, 2004.

[84] O. Stasse and G. Dauphin. Rapport sur l'implantation d'une structure logicielle permettant la mesure de qualité sur un flux vidéo comprimé par le protocole h263+. Technical report, RNTL Projet Cleopatre, 2004.

[85] Olivier Stasse. *"Distributed real time development architecture for complex robotic applications with Visual Attention as a study case"*. PhD thesis, Universit Pierre et Marie Curie (PARIS VI), 2000.

## 14.8 Videos

[86] S. Hak, N. Mansard, O. Ramos, L. Saab, and O. Stasse. Capture, recognition and imitation of antropomorphic motion. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, pages 3539–3540, 2012. Video Session.

[87] O. Stasse, P. Evrard, N. Mansard, P. Gergondet, A. Kheddar, K. Yokoi, E. Yoshida, T. Schauss, C. Passenberg, A. Peer, M. Buss, A. Weiss, M. Tscheligi, E. Gribovskaya, A. Billard, L. Blasi, J. Gancet, and M. Segara, 2009.

[88] O. Stasse, P. Evrard, N. Perrin, N. Mansard, and A. Kheddar. Fast foot prints replanning and generation during walking in physical human-humanoid interaction. In *IEEE/RAS Int. Conf. on Humanoid Robotics (ICHR)*, pages 284–289, 2009.

[89] A. Peer, S. Hirche, C. Weber, I. Krause, M. Buss, S. Miossec, P. Evrard, O. Stasse, Ee S. N., A. Kheddar, and K. Yokoi. Intercontinental cooperative telemanipulation between germany and japan. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 2715–2716 Video Session, 2008.

[90] F. Saïdi, O. Stasse, K. Yokoi, and F. Kanehiro. Online object search with a humanoid robot, 2007. http://homepages.laas.fr/ostasse/iros07.mpg.

[91] O. Stasse, B. Verrelst, A. J Davison, N. Mansard, B. Vanderborght, C. Esteves, F. Saïdi, and K. Yokoi. Integrating vision and walking to increase humanoid autonomy. In *IEEE/RAS Int. Conf. on Robotics and Automation (ICRA)*, pages 2272–2273, 2007. Finalist for the best video award.

# Chapter 15

# References

## 15.1 Bibliography


[AAV+08] T. Asfour, P. Azad, N. Vahrenkamp, K. Regenstein, A. Bierbaum, K. Welke, J. Schroeder, and R. Dillmann, *Toward humanoid manipulation in human-centred environments*, Robotics and Autonomous Systems **56** (2008), no. 1, 54–65.

[AEK07] S. Miossec A. Escande and A. Kheddar, *Continuous gradient proximity distance for humanoids collision-free optimized postures*, IEEE-RAS International Conference on Humanoid Robots, 2007, pp. 188–195.

[AEM06] A. Kheddar A. Escande and S. Miossec, *Planning support contact-points for humanoid robots and experiments on hrp-2*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2006, pp. 2974–2979.

[AHW+11] A. Andreopoulos, S. Hasler, H. Wersing, H. Janssen, J.K. Tsotsos, and E. Körner, *Active 3d object localization using a humanoid robot*, IEEE Transactions on Robotics **27** (2011), no. 1, 47–64.

[AKOI06] R. Adachi, S. Kanzaki, K. Okada, and M. Inaba, *Load distributed whole-body motion generation method for humanoids by minimizing average joint torque ratio*, Proc. of International Conference on Intelligent Autonomous Systems (IAS-9), 2006, pp. 804–811.

[ARA+06] T. Asfour, K. Regenstein, P. Azad, J. Schroder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann, *Armar-iii: An integrated humanoid platform for sensory-motor control*, 2006, pp. 169–175.

[ASOH07] A. Albu-Schäffer, C. Ott, and G. Hirzinger, *A unified passivity-based control framework for position, torque and impedance control of flexible joint robots*, The International Journal of Robotics Research **26** (2007), no. 1, 23–29.

[BA03] G.A. Borges and M.J. Aldon, *Robust estimation algorithm for mobile robot localization on geometricalenvironment maps*, Robotics and Autonomous Systems (2003), 131–159.

[BCE02] J.-M. Bourgeot, N. Cislo, and B. Espiau, *Path-planning and tracking in a 3d complex environment for an anthropomorphic biped robot.*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), vol. 3, 2002, pp. 2509–2514.

[BELK09] K. Bouyarmane, A. Escande, F. Lamiraux, and A. Kheddar, *Potential field guide for humanoid multicontacts acyclic motion planning*, IEEE/RAS ICRA, 2009, pp. 1165–1170.

[BETG08] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, *Surf: Speeded up robust features*, Computer Vision and Image Understanding (CVIU) **110** (2008), no. 3, 346–359.

[BLB+07] T. Buschmann, S. Lohmeier, M. Bachmayer, H. Ulbrich, and F. Pfeiffer, *A collocation method for real-time walking pattern generation*, Proc. 7th IEEE-RAS International Conference on Humanoid Robots, November 2007, pp. 1–6.

[BLMV06] O. Benichou, C. Loverdo, M. Moreau, and R. Voituriez, *Two-dimensional intermittent search processes: An alternative to lvy flight strategies*, Physical Review E **74** (2006), 020102–1 – 0201020–4.

[Bre03a] T. M. Breuel, *Implementation techniques for geometric branch-and-bound matching methods*, Computer Vision and Image Understanding **90** (2003), 258–294.

[Bre03b] ———, *On the use of interval arithmetic in geometric branch-and-bound algorithms*, Pattern Recognition Letters **24** (2003), 1375–1384.

[Bru01] H. Bruyninckx, *Open robot control software: The orocos project.*, IEEE/RAS Intl. Conf. on Robotics and Automation, vol. 3, 2001, pp. 2523–2528.

[BSF+12] T. Buschmann, M. Schwienbacher, V. Favot, A. Ewald, and H. Ulbrich, *The biped walking robot lola - hardware design and walking control -*, Journal of the Robotics Society of Japan **30** (2012), no. 4, 363–366.

[Bus10] T. Buschmann, *Simulation and control of biped walking robots*, Ph.D. thesis, Technical University of Munich, 2010.

[CBT+07] S. Condamin, O. Bénichou, V. Tejedor, R. Voituriez, and J. Klafter, *First-passage times in complex scale-invariant media*, Nature letters **450** (2007), 77–80.

[CDF+04] G. Csurka, C. Dances, L. Fan, J. Willamowski, and C. Bray, *Visual categorization with bags of keypoints*, ECCV International Workshop on Statistical Learning in Computer Vision, 2004.

[CKNK03] J. Chestnutt, J. Kuffner, K. Nishiwaki, and S. Kagami, *Planning biped navigation strategies in complex environments*, IEEE Intl Conf. on Humanoid Robotics, 2003.

[CKO11] B.-K. Cho, J.-H. Kim, and J.-H. Oh, *Online balance controllers for a hopping and running humanoid robot*, Advanced Robotics **25** (2011), no. 9-10, 1209–1225.

[CLC+05] J. Chesnutt, M. Lau, G. Cheung, J. Kuffner, J. Hodgins, and T. Kanade, *Footstep planning for the honda asimo humanoid*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), April 2005, pp. 631–636.

[Con85a] C. Connolly, *The determination of next best views*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), 1985, pp. 432–435.

[Con85b] C. J. Connolly, *The determination of next best views*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), 1985, pp. 432–435.

[DDW+08] H. Diedam, D. Dimitrov, P.-B. Wieber, K. Mombaur, and M. Diehl, *Online walking gait generation with adaptive foot positioning through linear model predictive control*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), September 22–26, 2008, pp. 1121–1126.

[Dou02] L. Douadi, *Recalage de nuage de points 3d fournis par un capteur embarqué*, Tech. report, LIRMM, 2002.

[Dun09] C. Dune, *Localisation et caractrisation d'objets inconnus partir d'informations visuelles : vers une saisie intuitive pour les personnes en situation de handicap*, Ph.D. thesis, Thse de l'Universit de Rennes 1, April 2009.

[DWFD08] D. Dimitrov, P.-B. Wieber, H. Ferreau, and M. Diehl, *On the implementation of model predictive control for on-line walking pattern generation*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), May 19–23, 2008, pp. 2685–2690.

[EK10] A. Escande and A. Kheddar, *Motion planning for humanoid robots*, ch. Multi-contact Acyclic Motion Planning and Experiments on HRP-2 Humanoid, pp. 161–179, Springer-Verlag, 2010.

[EKMG08] A. Escande, A. Kheddar, S. Miossec, and S. Garsault, *Planning support contact-points for acyclic motions and experiments on hrp-2*, ISER, 2008, pp. 293–302.

[ER01] F.-X. Espiau and P. Rives, *Extracting robust features and 3d reconstruction in water images*, OCEAN (Hawaii), MTS/IEEE, 2001, pp. 131–159.

[Eri05] C. Ericson, *Real-time collision detection*, The Morgan Kaufmann Series in Interactive 3D Technology, Morgan Kaufmann Publishers, 2005.

[Esc08] A. Escande, *Planification de points d'appui pour la génération de mouvements acycliques: application aux humanoïdes*, Ph.D. thesis, University of Evry-Val d'Essonne, 2008.

[FB81] M. A. Fischler and R. C. Bolles, *Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography*, Comm. of the AC **24** (1981), no. 6, 381–395.

[FBB04] M. Farenzena, A. Busti, and A. Benedetti, *Rigorous accuracy bounds for calibrated stereo reconstruction*, Proceedings of the 17 th International Conference on Pattern Recognition, vol. 4, 2004, pp. 288–292.

[FBFB04] A. Fusiello, A. Benedetti, M. Farenzena, and A. Busti, *Globally convergent autocalibration using interval analysis*, IEEE Transactions on Pattern Analysis and Machine Intelligence **26(12)** (2004), 1633–1638.

[FIR+12] M. Fumagalli, S. Ivaldi, M. Randazzo, L. Natale, G. Metta, G. Sandini, and F. Nori, *Force feedback exploiting tactile and proximal force/torque sensing - theory and implementation on the humanoid robot icub*, Autonomous Robots **33** (2012), no. 4, 381–398.

[Fle81] R. Fletcher, *Practical methods of optimization*, John Wiley & Sons, 1981.

[FMN+03] P. Fitzpatrick, G. Metta, L. Natale, S. Rao, and G. Sandini, *Learning about objects through action - initial steps towards artificial cognition*, IEEE/RAS Int. Conf. on Robotics and Automation, 2003, pp. 3140–3145.

[GJG05] M. Gienger, H. Janßen, and C. Goerick, *Task-oriented whole body motion for humanoid robots*, IEEE/RAS Intl. Conf. on Humanoids Robot, 2005, pp. 238–244.

[GJG06] M. Gienger, H. Janssen, and C. Goerick, *Exploiting task intervals for whole body robot control*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2006, pp. 2484–2490.

[GSV98] E. Grossmann and J. Santos-Victor, *The precision of 3D reconstruction from uncalibrated views*, proc. BMVC, vol. 1, 1998, pp. 115–124.

[GYT05] Y. Guan, K. Yokoi, and K. Tanie, *Feasibility: Can humanoid robots overcome given obstacles ?*, IEEE ICRA, 2005, pp. 1066–1071.

[Har96] R.M. Haralick, *Propagating covariance in computer vision*, Proceedings of Workshop on Performance Characterisics of Vision Algorithms, 1996, pp. 1–12.

[Hau08] K. Hauser, *Motion planning for legged and humanoid robots*, Ph.D. thesis, Stanford University, 2008.

[HBHL08] K. Hauser, T. Bretl, K. Harada, and J. Latombe, *Algorithmic foundations of robotics vii*, Springer Tracts in Advanced Robotics (STAR), vol. 47, ch. Using motion primitives in probabilistic sample-based planning for humanoid robots, pp. 507–522, Springer Verlag, 2008.

[Her99] A. Hertzmann, *Introduction to 3d non-photorealistic rendering: Silhouettes and outlines, course notes*, SIGGRAPH 99, 1999.

[HHW+10] A. Herdt, D. Holger, P.B. Wieber, D. Dimitrov, K. Mombaur, and D. Moritz, *Online walking motion generation with automatic foot step placement*, Advanced Robotics **24** (2010), 719–737.

[HKKH06] K. Harada, S. Kajita, K. Kaneko, and H. Hirukawa, *An analytical method for real-time gait planning for humanoid robots*, Internation Journal of Humanoid Robotics **3** (2006), no. 1, 1–19.

[HKM09] S. Holmes, G. Klein, and D. W. Murray, *An $o(n^2)$ square root unscented kalman filter for visual simultaneous localization and mapping*, IEEE Transactions on Pattern Analysis and Machine Intelligence **31** (2009), no. 7, 1251–1263.

[HMN+09] K. Harada, M. Morisawa, S. Nakaoka, K. Kaneko, and S. Kajita, *Kinodynamic planning for humanoid robots walking on uneven terrain*, Journal of Robotics and Mechatronics **21** (2009), no. 3, 311–316.

[Hon11] Honda, *Honda unveils all-new asimo with significant advancements*, November 2011.

[HPW10] A. Herdt, N. Perrin, and P.-B. Wieber, *Walking without thinking about it*, IEEE International Conference on Robot Systems (IROS), 2010, pp. 190–195.

[HZ03a] R. Hartley and A. Zisserman, *Multiple view geometry - 2nd edition*, ch. 5, pp. 132–150, Cambridge University Press, 2003.

[HZ03b] Richard Hartley and Andrew Zisserman, *Multiple view geometry*, Cambridge University Press, 2003.

[IKN05] M. Inaba, S. Kagami, and K. Nishiwaki, *Robot anatomy*, ch. Hardware system, Iwanami, 2005.

[JEBA00] C. Dumont J. E. Banta, L. R. Wong and M. A. Abidi, *A next-best-view system for autonomous 3-d object reconstruction*, IEEE Transactions on Systems, Man, and Cybernetics, Part A **30** (2000), no. 5, 589–598.

[JKDW01] Luc Jaulin, Michel Kieffer, Olivier Didrit, and Eric Walter, *Applied interval analysis*, Springer Verlag, London, 2001.

[Kaj05] Shuuji Kajita, *Humanoid robot*, Omsha, 2005, (In Japanese) ISBN4-274-20058-2.

[Kan08] K. Kanatani, *Statistical optimization for geometric fitting: Theoritical accuracy bound and high order error analysis*, International Journal of Computer Vision **80** (2008), 167–188.

[KATT95] P. K. Allen K. A. Tarabanis and R. Y. Tsai, *A survey of sensor planning in computer vision*, IEEE Transactions on Robotics and Automation **11** (1995), no. 1, 86–104.

[Kha86] O. Khatib, *Real-time obstacle avoidance for manipulators and mobile robots*, Int. Journal of Robotics Research **5** (1986), no. 1, 90–98.

[Kha87] ———, *A unified approach for motion and force control of robot manipulators: The operational space formulation*, International Journal of Robotics Research **3** (1987), no. 1, 43–53.

[Khe06] A. Kheddar, *Robot@cwe*, 2006.

[KKK+01] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, *The 3d linear inverted pendulum mode : A simple modeling of a biped walking pattern generation*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), November 2001, pp. 239–246.

[KKK+02a] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Yokoi, and H. Hirukawa, *A realtime pattern generator for biped walking*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA) (Washington, USA), 2002, pp. 31–37.

[KKK+02b] K. Kaneko, F. Kanehiro, S. Kajita, K. Yokoyama, K. Akachi, T. Kawasaki, S. Ota, and T. Isozumi, *Design of prototype humanoid robotics platform for hrp*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), October 2002, pp. 1083–1090.

[KKK+03a] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, *Biped walking pattern generation by using preview control of zero-moment point*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), September 2003, pp. 1620–1626.

[KKK+03b] ———, *Resolved momentum control: Humanoid motion planning based on the linear and angular momentum*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), IEEE, 2003, pp. 1644–1650.

[KKM+07] S. Kajita, K. Kaneko, M. Morisawa, S. Nakaoka, and H. Hirukawa, *Zmp-based biped running enhanced by toe springs*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), 2007, pp. 3963–3969.

[KKM+11] K. Kaneko, F. Kanehiro, M. Morisawa, K. Akachi, G. Miyamori, A. Hayashi, and N. Kanehira, *Humanoid robot hrp-4 - humanoid*

*robotics platform with lightweight and slim body* -, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2011, pp. 4400–4407.

[KKN+02] J.J. Kuffner, S. Kagami, K. Nishiwaki, M. Inaba, and H. Inoue, *Dynamically-stable motion planning for humanoid robots.*, Autonomous Robots (Special Issue on Humanoid Robotics) **12** (2002), 105–118.

[KLK+09] O. Kanoun, F. Lamiraux, F. Kanehiro, E. Yoshida, J.-P. Laumond, and P.-B. Wieber, *Prioritizing linear equality and inequality systems: Application to local motion planning for redundant robots*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), 2009, pp. 2939–2944.

[KM01] K. Kanatani and D. D. Morris, *Gauges and gauge transformations for uncertainty description of geometric structure with indeterminacy*, IEEE Transactions on Information Theory **47** (2001), no. 5, 2017–2028.

[KM04] M. Kallmann and M. Mataric, *Motion planning using dynamic roadmaps*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), 2004.

[KM07] Georg Klein and David W. Murray, *Parallel tracking and mapping for small ar workspaces*, ISMAR, 2007, pp. 225–234.

[KMM+10] S. Kajita, M. Morisawa, K. Miura, S. Nakaoka, K. Harada, K. Kaneko, F. Kanehiro, and K. Yokoi, *Biped walking stabilization based on linear inverted pendulum tracking*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2010, pp. 4489–4496.

[KMS+10] F. Kanehiro, M. Morisawa, W. Suleiman, Kenji Kaneko, and E. Yoshida, *Integrating physical constraints into reactive leg motion generation*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), October 2010, pp. 4069 – 4076.

[KMWK11] F. Keith, N. Mansard, P.-B. Wieber, and A. Kheddar, *Optimization of tasks warping and scheduling for smooth sequencing of robotic actions*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2011, pp. 3887–3892.

[KNK+03] J. Kuffner, K. Nishiwaki, S. Kagami, M. Inaba, and H. Inoue, *Motion planning for humanoid robots*, 11th Intl Symp. of Robotics Research, 2003.

[KNKH07] S. Kajita, T. Nagasaki, K. Kaneko, and H. Hirukawa, *Zmp-based biped running control*, Robotics & Automation Magazine, IEEE **14** (2007), no. 2, 63–72.

[Koo80] B. O. Koopman, *Search and screening*, Pergamon Press, 1980.

[KTYP08] O. Khatib, P. Thaulad, T. Yoshikawa, and J. Park, *Torque-position transformer for task control of position controlled robots*, IEEE/RAS ICRA, 2008, pp. 1729–1734.

[KYK92] S. Kajita, T. Yamaura, and A. Kobayashi, *Dynamic walking control of a biped robot along a potential energy conserving orbit*, IEEE Transactions on Robotics and Automation **8** (1992), no. 4, 431–438.

[KYST01] S. Kajita, K. Yokoi, M. Saigo, and K. Tanie, *Balancing a humanoid robot using backdrive concerned torque control and direct angular momentum feedback*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), 2001, pp. 3376–3382.

[KYY04] T. Tsubouchi K. Yamazaki, M. Tomono and S. Yuta, *3-d object modeling by a camera equipped on a mobile robot*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), 2004, pp. 1399–1405.

[Lau06] J-P. Laumond, *Kineo cam: a success story of motion planning algorithms*, IEEE Robotics and Automation Magazine **13** (2006), no. 2, 90–93.

[LGG92] S. Lacroix, P. Grandjean, and M. Ghallab, *Perception planning for a multi-sensory interpretation machine*, Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on, 12-14 May 1992, pp. 1818–1824vol.2.

[LGLM90] Eric Larsen, Stefan Gottschalk, Ming C. Lin, and Dinesh Manocha, *Fast proximity queries with swept sphere volumes*, Tech. report, Department of Computer Science, University of North Carolina, 1990.

[LH01] P. Leven and S. Hutchinson, *Toward Real-Time Path Planning in Changing Environments*, Algorithmic and Computational Robotics: New Directions: the Fourth Workshop on the Algorithmic Foundations of Robotics (2001).

[Lie77] A. Liegeois, *Automatic supervisory control of the configuration and behavior of multibody mechanisms*, IEEE Trans. on Systems, Man and Cybernetics **7** (1977), no. 12, 868–871.

[LJJ09] D. Larlus, Verbeek J, and F. Jurie, *Category level object segmentation by combining bag-of-words models with dirichlet processes and random fields*, International Journal of Computer Vision (2009), in press.

[Low01] D. G. Lowe, *Local feature view clustering for 3d object recognition*, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2001, pp. 682–688.

[Low04] D.G. Lowe, *Distinctive image features from scale-invariant keypoints*, International Journal of Computer Vision **60** (2004), no. 4, 91–110.

[MAA+06] A. Morales, T. Asfour, P. Azad, S. Knoop, and R. Dillmann, *Integrated grasp planning and visual object localization for a humanoid robot with five-fingered hands*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2006, pp. 5663–5668.

[Man12] N. Mansard, *A dedicated solver for fast operational-space inverse dynamics.*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), 2012, pp. 4043–4049.

[MB93] J. Maver and R. Bajcsy, *Occlusions as a guide for planning the next view*, IEEE Transactions on Pattern Analysis and Machine Intelligence **17** (1993), no. 5, 417–433.

[MC04] N. Mansard and F. Chaumette, *Tasks sequencing for visual servoing*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS) (Sendai, Japan), 2004, pp. 992–997.

[MC07] ———, *Task sequencing for sensor-based control*, IEEE Transactions on Robotics **23** (2007), 60–72.

[MCD06] J.M.M. Montiel, J. Civera, and A.J. Davison, *Unified inverse depth parametrization for monocular slam*, Robotics: Science and Systems, 2006.

[MCdFB+09] R. Martinez-Cantin, N. de Freitas, E. Brochu, J. Castellanos, and A. Doucet, *A bayesian exploration-exploitation approach for optimal online sensing and planning with a visually guided mobile robot*, Robot and Autonomous Systems **27** (2009), no. 3, 93–103.

[MCK+08] P. Michel, J. Chestnutt, S. Kagami, K. Nishiwaki, J. J. Kuffner, and T. Kanade, *Humanoid navigation planning using future perceptive capability*, IEEE/RAS International Conference on Humanoid Robots, 2008, pp. 507–514.

[MFL+08] D. Meger, P-E. Forssen, K. Lai, S. Helmer, S. McCann, T. Southey, M. Baumann, J. J. Little, and D. G. Lowe, *Curious george: An attentive semantic robot*, Robotics and Autonomous Systems **56** (2008), no. 6, 503–511.

[MHK+07] M. Morisawa, K. Harada, S. Kajita, S. Nakaoka, K. Fujiwara, F. Kanehiro, K. Kaneko, and H. Hirukawa, *Experimentation of humanoid walking allowing immediate modification of foot place based on analytical solution*, IEEE Int. Conf. on Robotics and Automation, 2007, pp. 3989–3994.

[MMK+11] K. Miura, M. Morisawa, F. Kanehiro, S. Kajita, K. Kaneko, and K. Yokoi, *Human-like walking with toe supporting for humanoids*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2011, pp. 4428–4435.

[MN98] Steven Mills and Kevin Novins, *Interval computations in computer vision*, Proceedings of Image and Vision Computing New Zealand, 1998, pp. 142–145.

[Moo66] R.E. Moore, *Interval analysis*, Prentice-Hall, Englewood Cliffs, 1966.

[Mor88] H. Moravec, *Sensor fusion in certainty grids for mobile robots*, AI Magazine **9** (1988), no. 2.

[MSC05] E. Marchand, F. Spindler, and F. Chaumette, *Visp for visual servoing: a generic software platform with a wide class of robot control skills*, IEEE Robotics and Automation Magazine **12** (2005), no. 4, 40–52.

[MSV+08] G. Metta, G. Sandini, D. Vernon, L. Natale, and F. Nori, *The icub humanoid robot: an open platform for research in embodied cognition*, Proc. of the 8th Workshop on Performance Metrics for Intelligent Systems, 2008, pp. 50–56.

[MWBDW02] A.A. Makarenko, S.B. Williams, F. Bourgault, and H.F. Durrant-Whyte, *An experiment in integrated exploration*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), vol. 1, 2002, pp. 534 – 539.

[Nea98] Radford M. Neal, *Markov chain sampling methods for dirichlet process mixture models*, Tech. Report 9815, University of Toronto, sep 1998.

[NH86] Y. Nakamura and H. Hanafusa, *Optimal redundancy control of robot manipulators*, International Journal of Robotics Research **6** (1986), no. 1, 32–42.

[NHK+07] S. Nakaoka, S. Hattori, F. Kanehiro, S. Kajita, and H. Hirukawa, *Constraint-based dynamics simulator for humanoid robots with shock absorbing mechanisms*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2007, pp. 3641–3647.

[NIH+11] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon, *Kinectfusion: Real-time dense surface mapping and tracking*, International Symposium on Mixed and Augmented Reality, 2011, pp. 127–136.

[NK07] K. Nishiwaki and S. Kagami, *Walking control on uneven terrain with short cycle pattern generation*, Proc. 7th IEEE-RAS International Conference on Humanoid Robots, 2007, pp. 447–453.

[NK09] K. Nishiwaki and S. Kagami, *Online walking control system for humanoids with short cycle pattern generation*, International Journal of Robotics Research **28** (2009), no. 6, 729–742.

[NKK+07] K. Nishiwaki, J. Kuffner, S. Kagami, M. Inaba, and H. Inoue, *The experimental humanoid robot h7: a research platform for autonomous behaviour*, Philosophical Transactions of The Royal Society A **365** (2007), 79–107.

[NKS+04] K. Nagasaka, Y. Kuroki, S. Suzuki, Y. Itoh, and J. Yamaguchi, *Integrated motion control for walking, jumping and running on a small bipedal entertainment robot*, Proc. IEEE International Conference on Robotics and Automation ICRA '04, vol. 4, April 2004, pp. 3189–3194.

[NL08] A. Nakhaei and F. Lamiraux, *Motion planning for humanoid robots in environments modeled by vision*, IEEE International Conference on Humanoid Robots (Daejeon, Korea), December 2008.

[NSN+12] G. Nelson, A. Saunders, N. Neville, B. Swilling, J. Bondaryk, D. Billings, C. Lee, R. Playter, and M. Raibert, *Petman: A humanoid robot for testing chemical protective clothing*, Journal of the Robotics Society of Japan **30** (2012), no. 4, 372–377.

[NW00] J. Nocedal and S. J. Wright, *Numerical optimization 2nd edition*, Springer Series in Operations Research, 2000.

[NYK+05] E. S. Neo, K. Yokoi, S. Kajita, F. Kanehiro, and K. Tanie, *A switching command-based whole-body operation method for humanoid robots*, IEEE/ASME Transactions on Mechatronics **10** (2005), no. 5, 546–559.

[OEE+12] C. Ott, O. Eiberger, J. Englsberger, M. A. Roa, and A. Abu-Schäffer, *Hardware and control concept for an experimental bipedal robot with joint torque sensors*, Journal of Robotics Society of Japan **30** (2012), no. 4, 44–48.

[OEF+06] Ch. Ott, O. Eiberger, W. Friedl, B. Bäuml, U. Hillenbrand, Ch. Borst, A. Albu-Schäffer, B. Brunner, H. Hirschmuller, S. Kielhöfer, R. Konietschke, M. Suppa, T. Wimböck, F. Zacharias, , and G. Hirzinger, *A humanoid two-arm system for dexterous manipulation*, 2006, pp. 276–283.

[OKT+07] K. Okada, M. Kojima, S. Tokutsu, T. Maki, Y. Mori, and M. Inaba, *Multi-cue 3d object recognition in knowledge-based vision-guided humanoid robot system*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2007, pp. 1505–1506.

[ORH11] C. Ott, M. A. Roa, and G. Herzinger, *Posture and balance control for biped robots based on contact force optimization*, IEEE/RAS Int. Conf. on Humanoid Robotics (Humanoids), 2011, pp. 26–33.

[OTK+05] R. Ozawa, Y. Takaoka, Y. Kida, K. Nishiwaki, J. Chestnutt, J. Kuffner, S. Kagami, H. Mizoguchi, and H. Inoue, *Using visual odometry to create 3d maps for online footstep planning*, Proceedings of IEEE International Conference on Systems, Man and Cybernetics (SMC), IEEE, 2005, pp. 2643–2648.

[PCDG] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, *Capture point: A step toward humanoid push recovery*, IEEE/RAS Int. Conf. on Humanoid Robotics (Humanoids).

[PEK08] J-R. Chardonnet P. Evrard, F. Keith and A. Kheddar, *Framework for haptic interaction with virtual avatars*, IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), 2008.

[PHAS09] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, *Learning and generalization of motor skills by learning from demonstration*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), 2009, pp. 763–768.

[Pit99] R. Pito, *A solution to the next best view problem for automated surface acquisition*, IEEE Transactions on Pattern Analysis and Machine Intelligence **21** (1999), no. 10, 1016–1030.

[PK06] J. Park and O. Khatib, *Contact consistent control framework for humanoid robots*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), 2006.

[Pow04] J.D. Powell, *The newuoa software for unconstrained optimization without derivatives, damtp report 2004/na05*, Tech. report, University of Cambridge, England, 2004.

[PSRA05] R.V. Patel, F. Shadpey, F. Ranjbaran, and J. Angeles, *A collision-avoidance scheme for redundant manipulators: theory and experiments*, Journal of Robotic Systems **22** (2005), no. 12, 737–757.

[Riv00] P. Rives, *Visual servoing based on epipolar geometry*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), vol. 1, 2000, pp. 602–607.

[Rum83] S.M. Rump, *Solving algebraic systems with high accuracy*, pp. 51–120, eds. U. Kulisch and W. Miranker Academic Press, 1983.

[SB04] P. Sardin and G. Bessonnet, *Forces acting on a biped robot. center of pressure - zero moment point*, IEEE Transactions on Systems Man and Cybernetics **34** (2004), no. 5, 630–637.

[Sch05] K. Schittkowski, *Ql: A fortran code for convex quadratic programming - user's guide, report, version 2.11*, Department of Mathematics, University of Bayreuth, 2005.

[SD05] Vivek A. Sujan and Steven Dubowsky, *Efficient information-based visual robotic mapping in unstructured environments*, The International Journal of Robotics Research **24** (2005), no. 4, 275–293.

[SF99] J. Sanchiz and R. Fisher, *A next-best-view algorithm for 3d scene recovery with 5 degrees of freedom*, British Machine Vision Conference, 1999, pp. 163–172.

[SGJG06] Hisashi Sugiura, Michael Gienger, Herbert Janssen, and Christian Goerick, *Real-time self collision avoidance for humanoids by means of nullspace criteria and task intervals*, IEEE/RSJ International Conference on Humanoid Robots, 2006, pp. 575–580.

[SIT02] Y. Sumi, Y.and Y. Ishiyama, and F. Tomita, *Hyper frame vision: A real-time vision system for 6-dof object localization*, International Conference on Pattern Recognition, 2002, pp. 577–580.

[SK06] L. Sentis and O. Khatib, *A whole-body control framework for humanoids operating in human environments*, Proceedings of the IEEE Internactional Conference in Robotics and Automation, 2006.

[SKTT02] Y. Sumi, Y. Kawai, T.Yoshimi, and T. Tomita, *3d object recognition in cluttered environments by segment-based stereovision*, International Journal of Computer Vision **6** (2002), 5–23.

[SKYT98] Y. Sumi, Y. Kawai, T. Yoshimi, and F. Tomita, *Recognition of 3d free-form objects using segment-based stereo vision*, International Conference on Computer Vision, 1998, pp. 668–674.

[SLE91] C. Samson, M. Le Borgne, and B. Espiau, *Robot control: the task function approach*, Clarendon Press, Oxford, United Kingdom, 1991.

[SLY11] F. Seide, G. Li, and D. Yu, *Conversational speech transcription using context-dependent deep neural networks*, INTERSPEECH, 2011, pp. 437–440.

[SMD12a] H. Strasdat, J. M. M. Montiel, and A. J. Davison, *Visual slam: Why filter?*, Image Vision Comput. **30** (2012), no. 2, 65–77.

[SMD12b] Hauke Strasdat, J.M.M. Montiel, and Andrew J. Davison, *Visual slam: Why filter?*, Image and Vision Computing **30** (2012), no. 2, 65 – 77.

[SS91] B. Siciliano and J-J. Slotine, *A general framework for managing multiple tasks in highly redundant robotic systems*, IEEE Int. Conf. on Advanced Robotics (ICAR'91) (Pisa, Italy), 1991, pp. 1211–1216.

[SS02] D. Scharstein and R. Szeliski, *A taxomomy and evaluation of dense two-frame stereo correspondence algorithms*, International Journal of Computer Vision, IJCV **47** (2002), no. 1, 7–42.

[SS04a] Javier F. Seara and Georges Schmidt, *Intelligent gaze control for vision-guided humanoid walking: methodological aspects*, Robotics and Autonomous Systems **48** (2004), no. 4, 231–248.

[SS04b] A. J. Smola and B. Schölkopf, *A tutorial on support vector regression*, Statistics and Computing **14** (2004), no. 3, 199–222.

[SYLM07] W. Suleiman, E. Yoshida, J.-P. Laumond, and A. Monin, *On humanoid motion optimization*, IEEE/RAS Int. Conf. on Humanoid Robotics (Humanoids), 2007, pp. 180–187.

[SZ03] J. Sivic and A. Zisserman, *Video Google: A text retrieval approach to object matching in videos*, Proceedings of the International Conference on Computer Vision, vol. 2, October 2003, pp. 1470–1477.

[TAR03] Benoit Telle, Marie Josée Aldon, and Nacim Ramdani, *Guaranted 3d visual sensing based on interval analysis*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2003, pp. 1566–1571.

[TBF05] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*, 2005.

[Tel03] Benoit Telle, *Méthode ensembliste pour une reconstruction 3d garantie par stereo vision*, Ph.D. thesis, Université Montpellier II, 2003.

[TK06] G. Taylor and L. Kleeman, *Visual perception and robotic manipulation*, Springer, 2006.

[TMY09a] T. Takenaka, T. Matsumoto, and T. Yoshiike, *Real time motion generation and control for biped robot -1st report: Walking gait pattern generation-*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2009, pp. 1084 –1091.

[TMY09b] T. Takenaka, T. Matsumoto, and T. Yoshiike, *Real time motion generation and control for biped robot -3rd report: Dynamics error compensation-*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2009, pp. 1594 –1600.

[TMY+09c] T. Takenaka, T. Matsumoto, T. Yoshiike, T. Hasegawa, S. Shirokura, H. Kaneko, and A. Orita, *Real time motion generation and control for biped robot -4th report: Integrated balance control-*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2009, pp. 1601 –1608.

[TMYS09] T. Takenaka, T. Matsumoto, T. Yoshiike, and S. Shirokura, *Real time motion generation and control for biped robot -2nd report: Running gait pattern generation-*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), oct. 2009, pp. 1092 –1099.

[Tri98] B. Triggs, *Optimal estimation of matching constraint*, 3D structure from Multiple Images of Large-Scale environments, Berlin, Germany: Springer-Verlag", 1998, pp. 63–77.

[TTIA07] T. Takubo, T. Tanaka, K. Inoue, and T. Arai, *Emergent walking stop using 3-d zmp modification criteria map for humanoid robot*, IEEE/RAS International Conference on Robotics and Automation (ICRA), 2007, pp. 2676–2681.

[UGC04] A. Ude, C. Gaskett, and G. Cheng, *Support vector machine and gabord kernels for object recognition on a humanoid with active foveated vision*, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2004, pp. 668–673.

[van04] Gino van den Bergen, *Collision detection in interactive 3D environments*, The Morgan Kaufmann Series in Interactive 3D Technology, Morgan Kaufmann Publishers, 2004.

[vdWCS06] J. van de Weijer and C. Cordelia Schmid, *Coloring local feature extraction*, ECCV'06, 2006, pp. 334–348.

[VJ01] P. Viola and M. Jones, *Rapid object detection using a boosted cascade of simple features*, Proceedings IEEE Conf. on Computer Vision and Pattern Recognition, 2001, p. 511.

[VS06] Andrea Vedaldi and Stefano Soatto, *Local features, all grown up*, cvpr **02** (2006), 1753–1760.

[WC06] P.-B. Wieber and C. Chevallereau, *Online adaptation of reference trajectories for the control of walking systems*, Robotics and Autonomous Systems **54** (2006), no. 7, 559–566.

[Wie02] P.-B. Wieber, *On the stability of walking systems*, Proceedings of the International Workshop on Humanoid and Human Friendly Robotics, 2002.

[Wie05] ———, *Somme comments on the structure of the dynamics of articulated motion*, Fast Motions in Biomechanics and Robotics, 2005.

[Wie06] ———, *Trajectory free linear model predictive control for stable walking in the presence of strong perturbations*, 2006, pp. 137–142.

[Wix94] L. E. Wixson, *Gaze selection for visual search*, Ph.D. thesis, Department of Computer Science, Univ. of Rochester, 1994.

[WOU+06] K. Welke, E. Oztop, A. Ude, R. Dillmann, and G. Cheng, *Learning feature representations for an object recognition system*, IEEE/RAS Int. Conf. on Humanoids Robots, 2006, pp. 290–295.

[WRSR03] G. Roth W. R. Scott and J.F. Rivest, *View planning for automated three-dimensional object reconstruction and inspection*, ACM Computing Surveys **35** (2003), no. 1, 64–96.

[WSG+07] H. Wersing, S.Kirstein, M. Goetting, H. Brandl, M. Dunn, I. Mikhailova, C. Goerick, J.J. Steil, H. Ritter, and E. Krner, *Online learning of objects in a biologically motivated architecture*, International Journal of Neural Systems **17** (2007), 219–230.

[YHI+03a] 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*, IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), vol. 3, September 14–19, 2003, pp. 2985–2991.

[YHI+03b] K. Yokoyama, H. Handa, T. Isozumi, Y. Fukase, K. Kaneko, F. Kanehiro, Y. Kawai, F. Tomita, and H. Hirukawa, *Video: Carring a table with people*, 2003.

[YK09] T. Yoshikawa and O. Khatib, *Compliant humanoid robot control by the torque transformer*, IROS, 2009, pp. 3011–3018.

[YT96] Yiming Ye and John K. Tsotsos, *Sensor planning in 3d object search: its formulation and complexity*, Fourth International Symposium on Artificial Intelligence and Mathematics (Florida, U.S.A.), January 3-5 1996.

[YT99] ———, *Sensor planning for 3d object search*, Computer Vision and Image Understanding **73** (1999), no. 2, 145–168.

[ZF96] Cyril Zeller and Olivier Faugeras, *Camera self-calibration from video sequences: The kruppa equations revisited*, Tech. Report RR2793, INRIA, 1996.

[Zha98] Z. Zhang, *Determining the epipolar geometry and its uncertainty: A review*, International Journal of Computer Vision **27(2)** (1998), 161–198.

[Zha00] ———, *A flexible new technique for camera calibration*, IEEE Transaction on Pattern Analysis and Machine Intelligence **22(11)** (2000), no. MSR-TR-98-71, 1330–1334.