Noname manuscript No.
(will be inserted by the editor)

Mario Prats · Philippe Martinet · Angel P. del Pobil · Sukhan Lee

Robotic Execution of Everyday Tasks by means of External Vision/Force Control

Received: date / Accepted: date

Abstract In this article, we present an integrated manipulation framework for a service robot, that allows to interact with articulated objects at home environments through the coupling of vision and force modalities. We consider a robot which is observing simultaneously his hand and the object to manipulate, by using an external camera (i.e. robot head). Task-oriented grasping algorithms [1] are used in order to plan a suitable grasp on the object according to the task to perform. A new vision/force coupling approach [2], based on external control, is used in order to, first, guide the robot hand towards the grasp position and, second, perform the task taking into account external forces. The coupling between these two complementary sensor modalities provides the robot with robustness against uncertainties in models and positioning. A position-based visual servoing control law has been designed in order to continuously align the robot hand with respect to the object that is being manipulated, independently of camera position. This allows to freely move the camera while the task is being executed and makes this approach amenable to be integrated in current humanoid robots without the need of hand-eye calibration. Experimental results on a real robot interacting with different kind of doors are presented.

Keywords Household robots · Vision/force control · Task-oriented grasping

Mario Prats and Angel P. del Pobil
Universitat Jaume I, Castellón, Spain.
E-mail: mprats@icc.uji.es

Philippe Martinet
LASMEA, Clermont-Ferrand, France.
E-mail: Philippe.MARTINET@lasmea.univ-bpclermont.fr

Mario Prats, Philippe Martinet and Sukhan Lee
Intelligent Systems Research Center, Sungkyunkwan University, Suwon, Korea.

1 INTRODUCTION

Most of our physical interaction with the world is mediated by the use of our hands. A domestic robot companion must be able to reliably perform simple manipulation tasks in everyday environments such as opening a door to enter another room, or pulling a drawer open to take something out. Most of today's robots exhibiting such abilities do it in an ad-hoc fashion and having little flexibility. See, for example, [3], where the authors present a mobile manipulator for opening doors without using force feedback. Instead, success in manipulation relies on an accurate localization algorithm and detailed models of the doors. This is also the case for sophisticated humanoid robots in which the effort has been put into locomotion, leaving manipulation unaddressed. See for instance [4] in which a Kawada Industries HRP2 humanoid robot is used for grasping ad-hoc cylinders by using its hands as pincers; or [5] in which a Sony QRIO humanoid is used which, even though it is endowed with 5-fingered hands, its manipulative abilities are limited to grasp a ball or spongy foam objects.

There is a need for fully autonomous robots that make use of different and complementary sensor modalities to perform a great variety of tasks under all kinds of uncertainties. In particular, vision and force are the most important sensors for task execution. Whereas vision can guide the hand towards the object and supervise the task, force feedback can locally adapt the hand trajectory according to task forces. When dealing with disparate sensors, a fundamental question stands: how to effectively combine the measurements provided by these sensors? An approach of this problem is to combine the measurements using multi-sensor fusion techniques [6]. However, as pointed out by several researchers, such method is not well adapted to vision and force sensors since the data they provide measure fundamentally different physical phenomena, while multi-sensory fusion is aimed in extracting a single information from disparate sensor

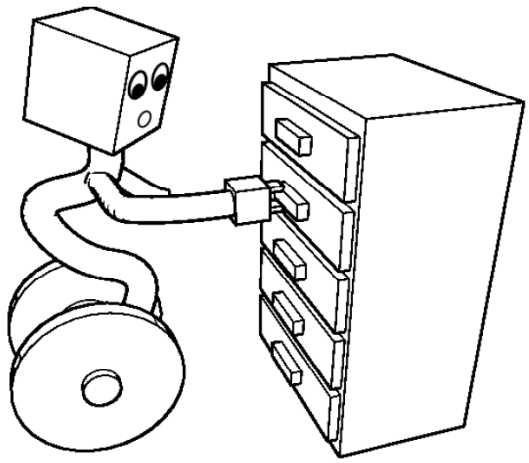

Fig. 1 Considered scenario.

data. Another approach to this problem is to combine visual and force data at the control level.

Some researchers have addressed the problem of vision/force control and two main approaches (impedance-based and hybrid-based strategies) have been studied [7–10]. In these approaches, the idea is merely to replace the classical position controller [11] by a vision-based controller. Hybrid control separates vision control and force control into two separate control loops, that operate in orthogonal directions. With this approach, it is not possible to control a direction simultaneously in vision and force. With the impedance-based control, the six degrees of freedom can be simultaneously vision- and force-controlled. However, coupling is done at the control level and local minima can appear during convergence.

In this article, we present a novel approach for sensor-guided robotic task execution that is amenable to be integrated in current mobile manipulators and humanoid robots. We consider a robot which is observing simultaneously his hand and the object to manipulate, by using an external camera (i.e. robot head, see Figure 1). Task-oriented grasping algorithms [1] are used in order to plan a suitable grasp on the object according to the task to perform. A new vision/force coupling approach [2] is used in order to, first, guide the robot hand towards the grasp position and, second, perform the task taking into account external forces.

The problem of hand/object alignment for grasping tasks has been addressed by other authors. In [12], a visual servoing framework for aligning the end-effector with an object was presented. Instead of working in the euclidean space, visual servoing was done on the projective space by doing projective reconstruction with a stereo camera, thus avoiding the need for camera calibration. The desired gripper-to-object relationship was learnt during an off-line procedure. In [13], an external position-based visual servoing approach was used on a humanoid robot in order to guide the hand towards the object. Hand pose was estimated by a kalman filter taking as input the stereo reconstruction of a set of LEDs attached on the robot hand.

As in [13], we also adopt a position-based visual servoing control law, because of the facilities that this approach offers for task specification. Instead of using a stereo camera and performing 3D reconstruction, we make use of a single camera and follow the virtual visual servoing approach for pose estimation [14]. The goal of the vision control loop is to align the gripper with respect to some part of the object (i.e, handle). As the pose of the gripper and the object is estimated on-line, the relative position between both can be computed at each iteration without the need of knowing the position of the camera with respect to the robot base. Therefore, the robot is still able to perform the task even in the presence of some camera motion. Task execution is independent of camera position. No extrinsic camera parameters are needed, which makes the integration of this approach into other robotic systems very easy, and opens the door to best-view planning algorithms for head control. In addition, instead of learning the grasp position during an offline stage like in [12], we make use of a task-oriented grasp planning algorithm [1] which autonomously computes which part of the object should be grasped in order to perform a given task. Finally, and in contrast with existing works, visual servoing does not finish when the robot grasps the object. Instead, a novel vision/force control framework is adopted in order to perform a given task on the object. Thus, visual servoing is not only used for hand-object alignment (reaching), but also for task execution and supervision (interaction). Our approach for vision/force coupling [2], based on the concept of external control [15], does the coupling in sensor-space, and not at the control level as classical impedance and hybrid approaches do [7–10]. This allows to control vision and force on all the degrees of freedom, whereas only the vision control law is controlling the robot. Note that in impedance and hybrid control, vision and force control outputs are added at the lowest level, making it possible to reach local minima when both vision and force control outputs are in conflict.

In summary, the main contributions of this work are:

- an external position-based visual servoing approach for aligning object and gripper, independently of camera motion, using virtual visual servoing pose estimation
- a novel method for vision/force coupling where the force control law modifies the visual reference, so that only the vision control law is connected to the robot, thus avoiding local minima.
- an integrated robotic manipulation system, using the above concepts, able to robustly perform common daily tasks by the coupling of visual and force feedback, after the automatic planning of the grasp according to the robot's purpose.

In Section 2, we describe the concept of everyday tasks as the kind of tasks that we want to perform with

our service robot. Section 3 introduces the theoretical framework of our work, consisting of the task-oriented grasp planning and the vision/force control scheme. In Section 4, the implementation on a real robot is described, and experimental results are presented. Finally, some conclusions and future lines are outlined in Section 6.

2 EVERYDAY TASKS

In this section, the kind of tasks our service robot has to perform are described and modelled according to a well-known task description formalism.

2.1 Considered tasks

We consider the general case of a mobile manipulator (or humanoid) working in a home environment. We assume that the robot is endowed with an object recognition module, so that it is able to recognize the object to manipulate and to retrieve its geometrical and structural model from a database. The robot is able to move in front of the object by using navigation capabilities such as mapping, localization, obstacle avoidance, etc.

With "execution of everyday tasks", we mean the robotic manipulation of articulated objects that can be commonly found in our everyday life, such as doors, drawers, windows, etc. For this, we need a formalism that, first, allows us to easily specify the tasks to the robot, and, second, allows the robot to compliantly execute the tasks under uncertainties. We make use of the Task Frame Formalism (TFF), first devised by Mason [16], and then reviewed in [17], because of its suitability for all kinds of force-controlled actions. We consider 1DOF mechanisms such as revolute joints (turning a knob, opening a door, etc.) and translational joints (opening a drawer, pushing a button, etc.). As shown in [17], this kind of tasks are well supported by the TFF. The programmer has to choose a suitable task frame (TF) so that some directions are velocity-controlled and some others are force-controlled, according to the natural constraints imposed by the environment (by the mechanism in our case).

2.2 Object and task modelization

Normally it is the programmer who specifies the TF in advance according to the task [18]. In our case, the robot chooses the most suitable TF autonomously by using a task-oriented grasp planning algorithm [1] that needs as input an object model including not only geometrical information, but also kinematic information, or a description of the object mechanism.

We describe an object as a set of different parts that are assembled together. Each part is defined on its own reference frame, which is independent from the other parts. A set of relations is defined between the parts, in terms of constrained and free degrees of freedom, i.e. a motion constraint is defined with each frame. Therefore, each of the frames defining the structure of the object can be used as the task frame.

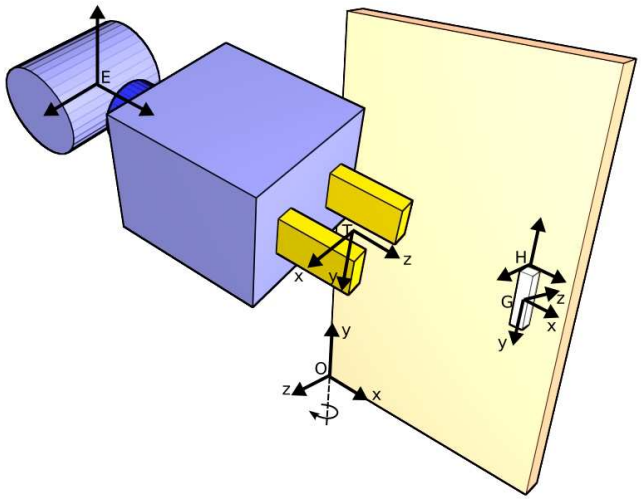

Fig. 2 Considered frames.

Figure 2 shows an example of a door representation. It is composed of two parts: the door table, defined in frame $\mathcal{O}$ -which is also the object reference frame- and the handle, defined in frame $\mathcal{H}$. The model, as described in [1], includes the relation between the different object parts. In this case, the relation between the handle and the door table is known, and represented as an homogeneous transformation matrix ${}^{\mathcal{O}}\mathbf{T}_{\mathcal{H}}$. The model also includes the degrees of freedom (motion constraint) for each part. In the example of Figure 2, the frame $\mathcal{H}$ is fixed with respect to $\mathcal{O}$, but the frame $\mathcal{O}$ has one degree of freedom: a rotation around Y axis, which corresponds to the task of opening the door. Thus, the task is specified to the robot by means of a frame (the task frame) and the degree of freedom that must be activated on it. For more details on the object representation, refer to [1].

2.3 Approaches to task execution

The task execution process for articulated objects can be divided into two stages:

- A reaching phase, where the hand of the robot must be moved towards the handle until the grasp is executed successfully.
- An interaction phase, where the hand is in contact with the object and the particular mechanism must be activated.

The reaching task can be performed in open loop if a good estimation of the object pose with respect to the robot is available. This is the approach followed in [3],

where the localization algorithm is able to provide the robot pose inside the map with 1mm of accuracy. However, this is not the general case, and, in the real life, the robot has to face with lots of uncertainties. Closed loop is more adequate if we want to deal with these uncertainties under not-structured environments. Normally, a visual servoing framework is adopted to close the loop during reaching [12, 13].

Regarding the interaction phase, it is worth noting that the robot hand is in contact with the environment, and any kind of uncertainty (errors in the models, bad pose estimation, etc.) may produce very big forces that can damage the environment or the robot. In [3] the authors still rely on the localization algorithm during the interaction phase, without using any kind of sensor feedback, which is very dangerous. When the robot is in contact with the environment, it is extremely important to design a controller that can deal with unpredicted forces and adapt the hand motion accordingly.

This is the reason why a vision/force coupling approach is adopted in this work. Vision feedback allows the robot to continuously track the object and to visually servo the hand for task execution. Force feedback allows to deal with errors in pose estimation and object models, so that any undesired external force can be compensated by modifying the control law, either at the control level as classical approaches do, or at the sensor level as we present in this work.

3 GENERAL FRAMEWORK

In this section, the theoretical description of the two main modules of our service robot application are described:

- Task-oriented grasp planning, in charge of choosing a grasp on the object, which is suitable for the task to perform.
- Vision/force control, in charge of visually guiding the robot hand towards the planned grasp, and then performing the task with vision and force feedback.

3.1 Task-oriented grasp planning

Task-oriented grasp planning deals with the problem of finding a grasp on an object which is suitable for a particular task. There are few works about grasping that take the task into account [19–21] and most of them do not consider the task during grasp planning. Instead, the task is considered on the grasp evaluation stage as a quality measure. In practice, lots of grasps would have to be generated and evaluated, making these approaches computationally unaffordable. In [1], we presented a task-oriented grasp planning algorithm based on hand preshapes [22].

The input to this algorithm is the object model (as described previously) and the task to perform, in terms of a mechanism (i.e. degree of freedom) to be activated on the object. The algorithm provides the following:

- A grasp frame, $\mathcal{G}$ in the example of Figure 2, where the hand has to be moved, which is related with the object reference frame, $\mathcal{O}$, by computing the homogeneous transformation matrix ${}^{\mathcal{O}}\mathbf{T}_{\mathcal{G}}$.
- A hand preshape suitable for the task, including a tool frame, $\mathcal{T}$ in Figure 2, attached to the hand, which determines the control strategy that will be followed for grasping [1]. The tool frame is related to the end-effector frame by the transformation ${}^{\mathcal{E}}\mathbf{T}_{\mathcal{T}}$

The original description of the task-oriented grasp planning algorithm was prepared for the Barrett Hand [1], but we have adapted it in order to deal with the parallel-jaw gripper that we use in this work. For this simple hand, we only consider the precision hand preshape [1]. The tool frame, $\mathcal{T}$, is set to the middle point between both fingertips as shown in Figure 2. The goal of the reaching phase is to move the tool frame $\mathcal{T}$ towards the grasp frame $\mathcal{G}$. This is done by visually computing the relative pose between both frames, and visual servoing the tool frame in order to reduce the pose error, as explained in Section 3.2. When the visual servoing control law has converged, the grasp frame is used as the task frame, where the task is defined in terms of the constrained motion. For more details, refer to [1].

3.2 Vision/force control for everyday tasks

For visually-guided reaching of the object, we propose a position-based visual servoing closed-loop approach where a robot head observes both the gripper and the object and tries to achieve a relative position between both, like in [12, 13]. Regarding the task execution, it is necessary to minimize external forces at the same time that the vision control law guarantees the whole task execution. In the following sections, the theoretical framework for our position-based visual servoing approach and vision/force control is presented.

3.2.1 External position-based visual servoing

Several vision-based control laws have been proposed in the literature [23]. They are generally classified in three groups, namely position-based, image-based and hybrid-based control. The first one works in 3D cartesian space and requires, in most cases, a model of the object and the camera intrinsic parameters [24]. In contrast, image-based visual servoing works directly in the image space [25]. More recently, several researchers have explored hybrid approaches which combine euclidean and image information [26].

As already mentioned in [13], the natural space for specifying the task is the cartesian space, and there are

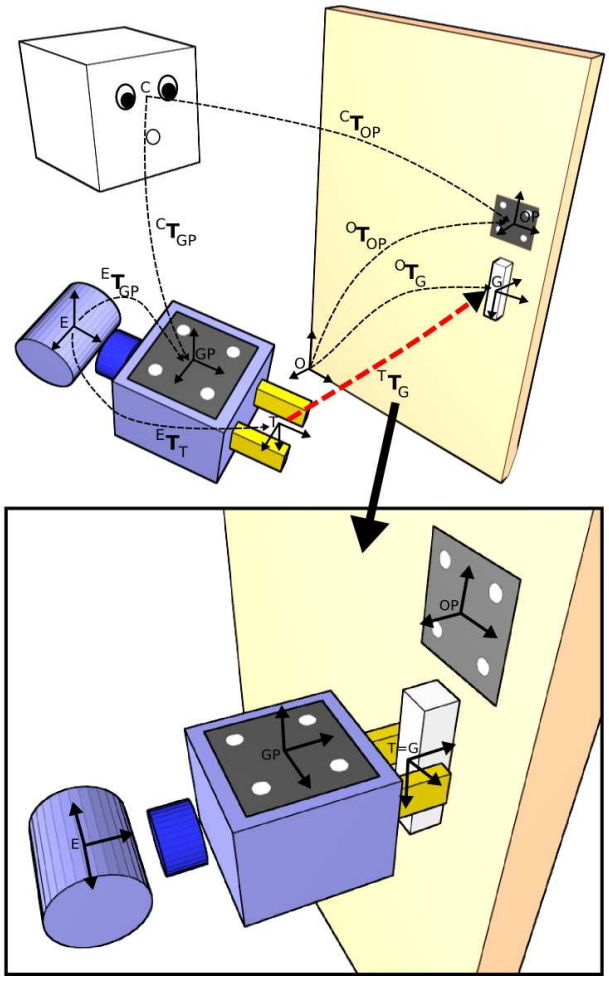

Fig. 3 The vision task is to guide frame $\mathcal{T}$ towards frame $\mathcal{G}$.

evidences that humans use 3D information for task planning [27]. Thus, we adopt a position-based visual servoing approach using an external camera which observes simultaneously the gripper and the object. Note that this is the common configuration in humanoid robots.

We set the vector $\mathbf{s}$ of visual features to:

$$\mathbf{s} = \begin{pmatrix} \mathbf{t} \\ \mathbf{u}\theta \end{pmatrix}$$

where $\mathbf{t}$ is the translational part of the homogeneous matrix ${}^{\mathcal{T}}\mathbf{T}_{\mathcal{G}}$, and $\mathbf{u}\theta$ is the axis/angle representation of the rotational part of ${}^{\mathcal{T}}\mathbf{T}_{\mathcal{G}}$.

The matrix ${}^{\mathcal{T}}\mathbf{T}_{\mathcal{G}}$, which relates hand and handle, is computed directly from the visual observation of the gripper and object, according to the following expression:

$$\left({}^{\mathcal{C}}\mathbf{T}_{\mathcal{GP}} \cdot {}^{\mathcal{E}}\mathbf{T}_{\mathcal{GP}}{}^{-1} \cdot {}^{\mathcal{E}}\mathbf{T}_{\mathcal{T}}\right)^{-1} \cdot {}^{\mathcal{C}}\mathbf{T}_{\mathcal{OP}} \cdot {}^{\mathcal{O}}\mathbf{T}_{\mathcal{OP}}{}^{-1} \cdot {}^{\mathcal{O}}\mathbf{T}_{\mathcal{G}} \quad (1)$$

where ${}^{\mathcal{C}}\mathbf{T}_{\mathcal{GP}}$ is an estimation of the pose of an arbitrary hand frame, expressed in camera frame. ${}^{\mathcal{C}}\mathbf{T}_{\mathcal{OP}}$ is an estimation of an arbitrary object frame pose, expressed in camera frame. We are currently estimating hand and object pose by virtual visual servoing [14], using a set of point features drawn on a pattern whose model is known. One pattern is attached to the gripper, in a known position ${}^{\mathcal{E}}\mathbf{T}_{\mathcal{GP}}$. Another pattern is attached to the object, also in a known position with respect to the object reference frame: ${}^{\mathcal{O}}\mathbf{T}_{\mathcal{OP}}$. As future lines we would like to implement a feature extraction algorithm in order to use natural features of the object instead of the markers. Note that this will not significantly affect the current implementation, as virtual visual servoing pose estimation can deal with different types of visual features [14]. Finally, the tool frame ${}^{\mathcal{E}}\mathbf{T}_{\mathcal{T}}$ and the grasp frame ${}^{\mathcal{O}}\mathbf{T}_{\mathcal{G}}$ are computed by the task-oriented grasp planning algorithm presented in section 3.1 and detailed in [1]. For a comprehensive description of the frames involved in the vision task, see Figure 3.

We compute the velocity in the tool frame $\tau_{\mathcal{T}}$ using a classical visual servoing control law:

$$\tau_{\mathcal{T}} = -\lambda \mathbf{e} + \widehat{\frac{\partial \mathbf{e}}{\partial t}} \quad (2)$$

where $\mathbf{e}(\mathbf{s}, \mathbf{s}^*) = \widehat{\mathbf{L}_{\mathbf{s}}^{+}}(\mathbf{s} - \mathbf{s}^*)$ (in our case, $\mathbf{s}^* = 0$). The last term, $\widehat{\frac{\partial \mathbf{e}}{\partial t}}$, is the estimation of how the visual features change over time. It is related to the object motion, and should be taken into account when the hand is in contact with the environment in order to reduce tracking errors. However, we can neglect it, because the use of force feedback allows us to cope with these small tracking errors, as long as the task velocity is small. The interaction matrix $\widehat{\mathbf{L}_{\mathbf{s}}}$ is set for the particular case of position-based visual servoing:

$$\widehat{\mathbf{L}_{\mathbf{s}}} = \begin{pmatrix} -\mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & -\mathbf{L}_w \end{pmatrix}$$

$$\mathbf{L}_w = \mathbf{I}_{3\times3} - \frac{\theta}{2}[\mathbf{u}]_\times + \left(1 - \frac{\mathrm{sinc}(\theta)}{\mathrm{sinc}^2(\frac{\theta}{2})}\right)[\mathbf{u}]_\times^2$$

where $[\mathbf{u}]_\times$ is the skew anti-symmetric matrix for the rotation axis $\mathbf{u}$. Finally, the joint velocities that are sent to the robot are computed as:

$$\dot{\mathbf{q}} = \mathbf{J}^{-1} \cdot \widehat{\mathbf{L}_\times} \cdot \begin{pmatrix} {}^{\mathcal{E}}\mathbf{R}_{\mathcal{T}} & [{}^{\mathcal{E}}\mathbf{t}_{\mathcal{T}}]_\times \cdot {}^{\mathcal{E}}\mathbf{R}_{\mathcal{T}} \\ \mathbf{0}_{3\times3} & {}^{\mathcal{E}}\mathbf{R}_{\mathcal{T}} \end{pmatrix} \cdot \tau_{\mathcal{T}}$$

where $\mathbf{J}$ is the robot jacobian and $\widehat{\mathbf{L}_\times}$ relates $\tau_{\mathcal{E}}$ and $\dot{\mathbf{X}}_{\mathcal{E}}$ according to $\dot{\mathbf{X}}_{\mathcal{E}} = \widehat{\mathbf{L}_\times} \cdot \tau_{\mathcal{E}}$ [24,28]. It is worth noting that, for very small displacements, $\widehat{\mathbf{L}_\times}$ can be taken as the identity matrix, and, thus, $\dot{\mathbf{X}}_{\mathcal{E}} = \tau_{\mathcal{E}}$. Finally, ${}^{\mathcal{E}}\mathbf{R}_{\mathcal{T}}$ and ${}^{\mathcal{E}}\mathbf{t}_{\mathcal{T}}$ are, respectively, the rotational and translational part of the homogeneous transformation matrix ${}^{\mathcal{E}}\mathbf{T}_{\mathcal{T}}$.

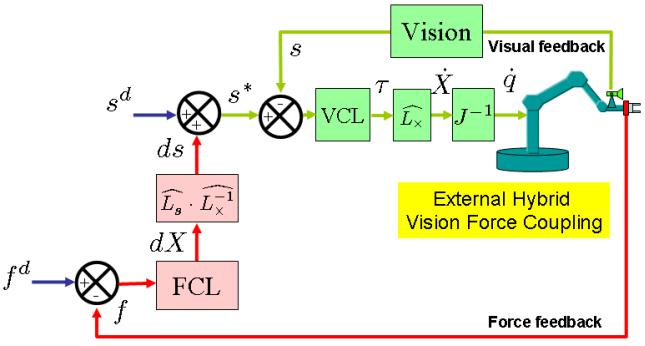

Fig. 4 External hybrid vision/force coupling.

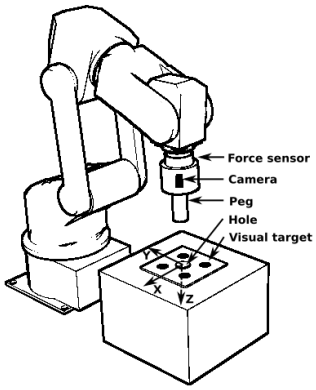

Fig. 5 General setup for vision/force control simulations

3.2.2 Vision/force control law

Computer vision can provide a powerful way of sensing the environment and can potentially reduce or avoid the need for environmental modeling. Vision allows accurate part alignment in partially unknown and/or dynamic environments without requiring contacts. Force sensor provides localized but accurate contact information. To combine visual and force information, two main approaches (impedance-based and hybrid-based strategies) have been studied [7–10]. In these approaches, the idea is merely to replace the classical position controller [11] by a vision-based controller. In both cases, the addition of the vision and force control outputs is done at the lowest level (control level). This can lead to local minima when both outputs are in conflict (same value and opposite signs).

In [2], we proposed a novel vision/force coupling approach, based on external control [15] where the force control loop is closed around an internal vision control loop in a hierarchical way (see Figure 4). The reference trajectory $\mathbf{s}^d$ used as original input of the vision-based controller is modified according to the external force control loop. The force control is performed by direct control: when the robot is moving $\mathrm{d}\mathbf{X}$ against a contact surface, the force measurement is proportional to the environment stiffness $\mathbf{K}$ and the displacement $\mathrm{d}\mathbf{X}$. Then, instead of adding the force control output to the vision control output as classical approaches do, the force control output is used to modify the desired vector of visual features $\mathbf{s}^d$, by projecting it on sensor space using the interaction matrix $\widehat{\mathbf{L_s}}$. Either if the end-effector is in contact with the environment or not, the robot is only controlled by the visual control law, and thus, the proposed control scheme has the same stability and convergence properties as the particular visual control law we choose [23]. In other words, force feedback does not introduce and problem of stability in the proposed control law.

In the control scheme, shown in Figure 4, the desired wrench $\mathbf{f}^d$ is added as input in the force feedback control loop. The stiffness is controlled by the force controller (FCL) according to a proportional control law:

$$d\mathbf{X} = \mathbf{K}^{-1}(\mathbf{f}^d - \mathbf{f})$$

Although we have chosen stiffness control for this work, the control scheme is general and it is possible to implement another more complex type of force control. Unlike existing approaches, the force controller does not modify the vision control output. Instead, it only modifies the reference trajectory of visual observations $\mathbf{s}^d$:

$$\mathbf{s}^* = \mathbf{s}^d + d\mathbf{s} \tag{3}$$

where $\mathbf{s}^*$ is the modified reference for visual features and $\mathrm{d}\mathbf{s}$ can be computed by projecting $\mathrm{d}\mathbf{X}$ by means of the interaction matrix as $d\mathbf{s} = \widehat{\mathbf{L_s}} \cdot \widehat{\mathbf{L}_x^{-1}} \cdot d\mathbf{X}$. It is worth noting that $d\mathbf{X}$ must be first transformed, from the force sensor frame, to the camera frame, via the corresponding screw transformation matrix.

The hierarchical juxtaposition of the force control loop on the vision control loop provides several advantages according to the existing methods [7, 9]: selection matrices and time-dependent geometric transformations are eliminated from the control loop leading to a controller design independent of the arm configuration. Since the force control only acts on the reference trajectory, conflicts between force and vision controllers are avoided. For a detailed analysis, refer to [2].

This is shown by simulation results in Figures 6, 7, 8 and 9, for a peg-in-hole task, where a robot with an eye-in-hand camera has to insert a peg of 10 cm length into a hole of the same depth. The difference between the peg and the hole diameters is 3 mm. The hole is in the center of a pattern composed by four circles forming a square, as shown in Figure 5. We assume that the hole, and thus also the pattern, are on the origin of the world frame, but rotated 10 degrees around Y axis (one of the axis contained in the pattern plane). The goal of the vision part is to reach a camera position so that the square is centered on the image at a given size (see Figure 6b). This desired position corresponds to the one when the

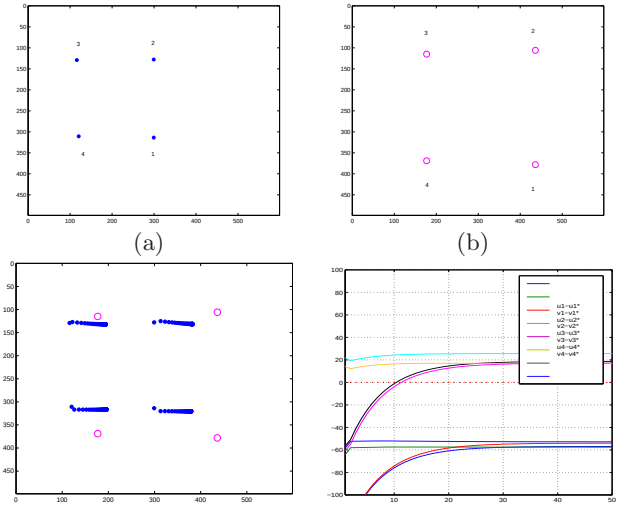

Fig. 6 Hybrid vision/force for a peg-in-hole task. Initial camera position is set to $\mathbf{X} = (0.02, -0.02, -0.36, -5, 5, 0)^T$ with respect to the world frame. (a) Initial image, (b) Desired image, (c) Features trajectory in the image plane, (d) Image error

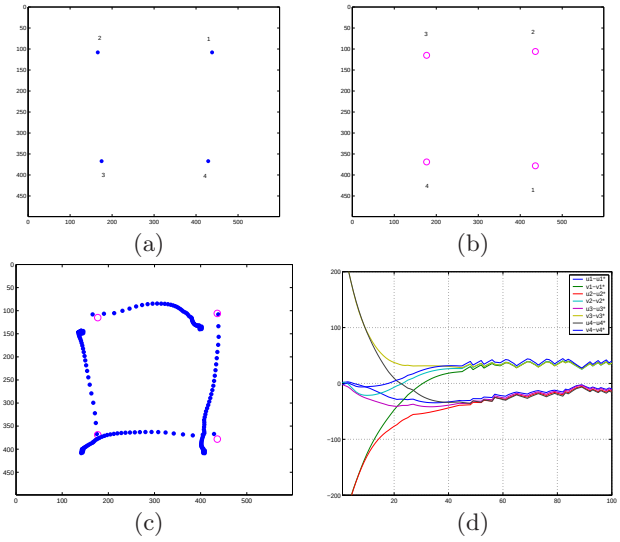

Fig. 7 Impedance vision/force for a peg-in-hole task. Initial camera position is set to $X = (0, 0, -0.25, 0, 0, 90)$ with respect to the world frame. (a) Initial image, (b) Desired image, (c) Features trajectory in the image plane, (d) Image error

peg is successfully inserted into the hole, and can be learnt during an off-line process.

Figures 6 and 7 show two different cases where the hybrid vision/force control and the impedance-based control fail (as shown in Figures 6c, 6d, 7c and 7d, the desired goal is never reached) . Figures 8 and 9 show the convergence of our proposed vision/force control law in the same conditions. For more details on these results, refer to [2].

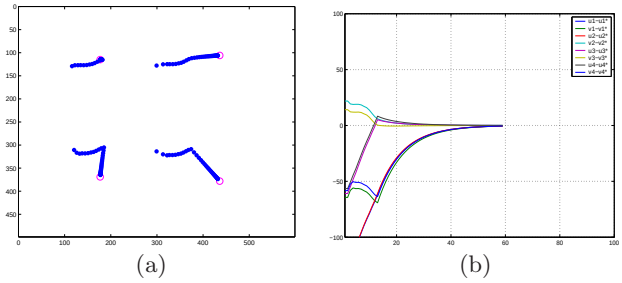

Fig. 8 External control for a peg-in-hole task. Initial camera position is set to $\mathbf{X} = (0.02, -0.02, -0.36, -5, 5, 0)^T$ with respect to the world frame. (a) Features Trajectory in the image plane, (b) Image error

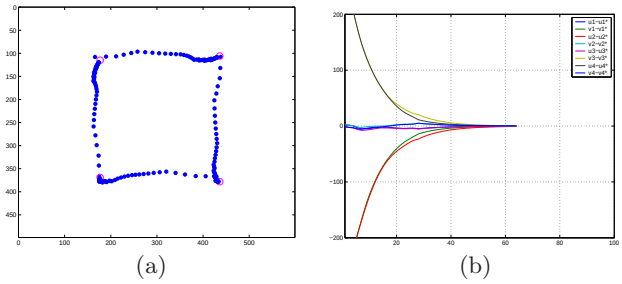

Fig. 9 External control for a peg-in-hole task. Initial camera position is set to $X = (0, 0, -0.25, 0, 0, 90)$ with respect to the world frame. (a) Features Trajectory in the image plane, (b) Image error

4 APPLICATION AND IMPLEMENTATION

The theoretical development of the previous section has been applied to a real robot. We have used a mobile manipulator composed of an Amtec 7DOF ultra light weight robot arm mounted on an ActivMedia PowerBot mobile robot. The hand of the robot is a PowerCube parallel jaw gripper. This robot belongs to the Intelligent Systems Research Center (Sungkyunkwan University, South Korea), and is already endowed with recognition and navigation capabilities [29] [30], so that it is able to recognize the object to manipulate and to retrieve its geometrical and structural model from a database. The robot is able to move in front of the object by using navigation capabilities such as mapping, localization, obstacle avoidance, etc.

As already mentioned, our goal is to interact with the different furniture and articulated objects that can be found in home environments, such as doors, windows, wardrobes, drawers, lights, etc. Our task starts when the mobile manipulator has navigated in front of the object that is going to be manipulated and has a view of both the object and its hand. For the experimental validation we have chosen a door opening task, because it is the most common task in home environments. Concretely, experimental results are presented with two very differ-

ent doors: a closet and a refrigerator. Both doors are very different in size, and have different handles. But both tasks can be described in terms of the structural model (Figure 2) as applying a rotational velocity around Y axis of frame $\mathcal{O}$. The task-oriented grasp planning algorithm [1] computes the grasp frame $\mathcal{G}$ and the tool frame $\mathcal{T}$ for each particular case, and relates them to the object reference frame with ${}^{\mathcal{O}}\mathbf{T}_{\mathcal{G}}$, and to the end-effector frame with ${}^{\mathcal{E}}\mathbf{T}_{\mathcal{T}}$, respectively (see Figure 3). It is worth noting that only the door model changes from one execution to the other. The grasp and the vision/force references are computed automatically taking the model as input, which makes this approach suitable for any kind of door as long as it has been recognized by the robot (in order to retrieve the model).

The whole manipulation process is divided into two steps: reaching a handle with a task-suitable grasp, and interacting with the environment (performing the task).

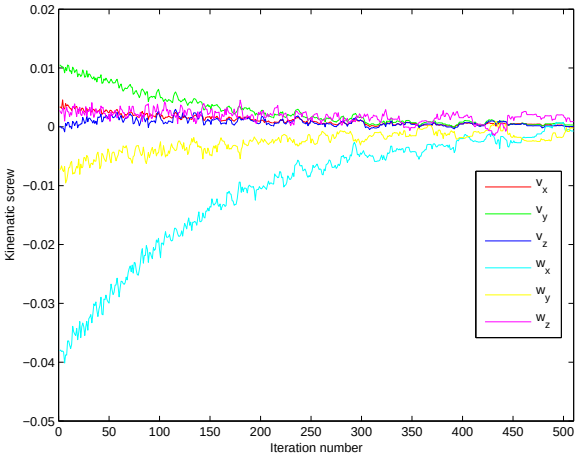

Fig. 10 Kinematic screw for reaching the pre-grasp position (closet task).

4.1 Reaching

The reaching task is divided into three different subtasks: reaching a pre-grasp position, reaching the grasp position and performing the grasp. The robot switches from one subtask to another when the resulting velocity of the vision/force controller is close to zero (i.e, the desired references have been reached).

4.1.1 Reaching a pre-grasp position

A pre-grasp frame, $\mathcal{P}$, is computed by the task-oriented grasp planning algorithm [1], and it is related to the object reference frame with the transformation ${}^{\mathcal{O}}\mathbf{T}_{\mathcal{P}}$. The pre-grasp position is used in order to adopt an initial configuration with respect to the final grasp frame so that the robot can reach the handle from a good direction.

The transformation between the tool frame and the pre-grasp frame ${}^{\mathcal{T}}\mathbf{T}_{\mathcal{P}}$ is computed at each iteration by equation 1, and then used for building the visual features vector $\mathbf{s}$. During this step there is no contact with the environment. Thus, the force loop in the vision/force control law is not modifying the visual reference. This means that the system behaves according to the vision control law of section 3.2.1

Figure 10 shows the evolution of the visual velocity for each degree of freedom (only for the case of the closet door). Initially, the tool frame is far from the pre-grasp frame (see Figure 11a), so that there is a large visual error. The visual control law makes this error converge to zero, which corresponds to the situation where the tool frame matches with the pre-grasp frame (see Figure 11b).

4.1.2 Reaching the grasp position

During this step, the tool frame is moved, from the pre-grasp position towards the grasp frame, as shown in Figure 11c. A new vector of visual features is computed according to equation 1. Thus, the handle is reached from the reaching direction established by the transformation ${}^{\mathcal{P}}\mathbf{T}_{\mathcal{G}}$. At the end of this step, the grasp frame and the tool frame are the same (up to modelling errors), which means that the handle is situated between the robot fingertips.

4.1.3 Performing the grasp

The last step of the reaching stage is to grasp the handle. The previous step guarantees that the handle is between the robot fingertips. Thus, the robot gripper is closed in order to grasp it, as shown in Figure 11d.

During this step the first contacts appear. Thus, at the same time that the gripper is closed, the vision/force control law is active. The reference for the vision control law is to match the grasp and tool frames (i.e, keep the handle in the middle point between both fingertips). The reference for the force control law is to minimize external forces ($\mathbf{f}^d = 0$). If, due to modelling errors, the handle is not perfectly placed in the middle point between the fingertips, then one finger will make contact before the other. This will generate a force that the force control law will try to regulate to zero by modifying the vision reference. During this step, the stiffness coefficient on Y direction of frame $\mathcal{E}$ is set to a small value in order to make the robot highly compliant in this direction. When the grasp is finished, the task frame is set to the tool frame in order to specify the task in terms of the constrained motion.

The real behavior is shown in Figure 12 (only for the case of the closet door). Due to a premature contact on one of the fingers, it appears a force in Y direction and a torque in X axis (expressed in effector frame $\mathcal{E}$). These forces modify the visual reference (i.e. the grasp frame

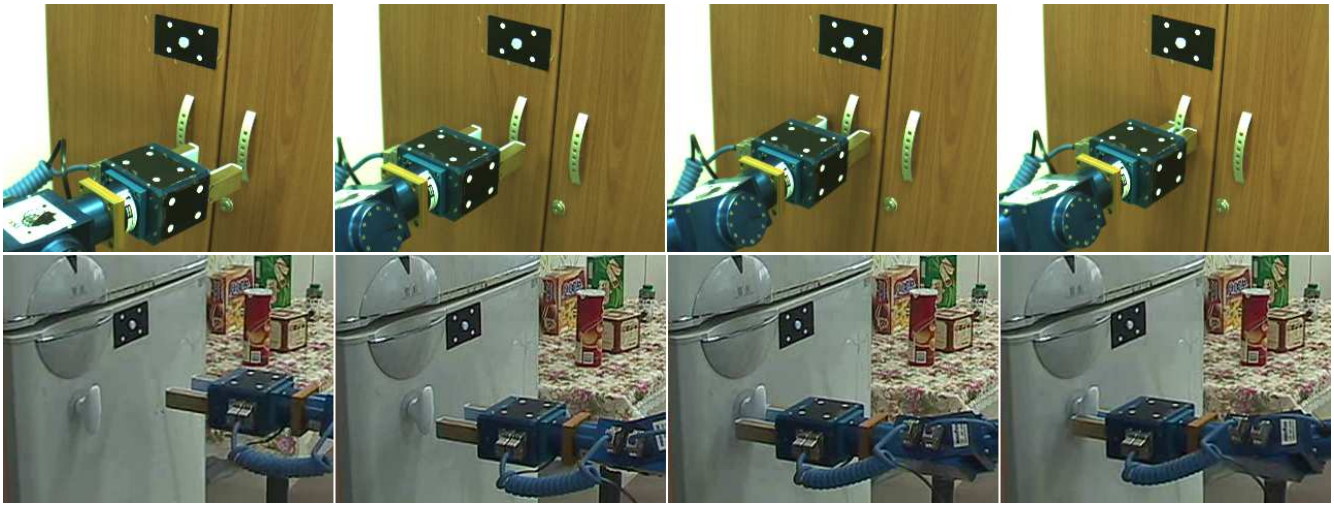

Fig. 11 Reaching the handle. Top row: closet. Bottom row: refrigerator. From left to right: (a) Initial position (b) Reaching the pre-grasp position (c) Reaching the grasp position (d) Grasping.

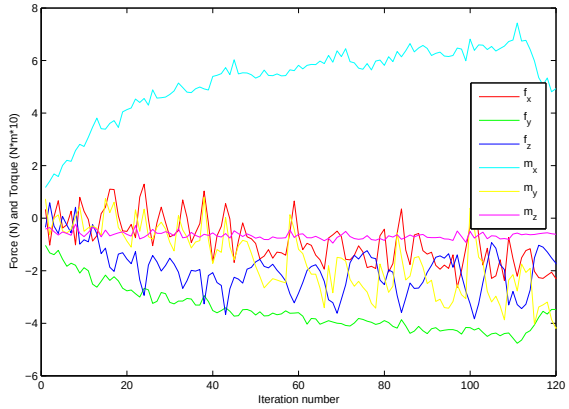

Fig. 12 Forces during grasping (closet task).

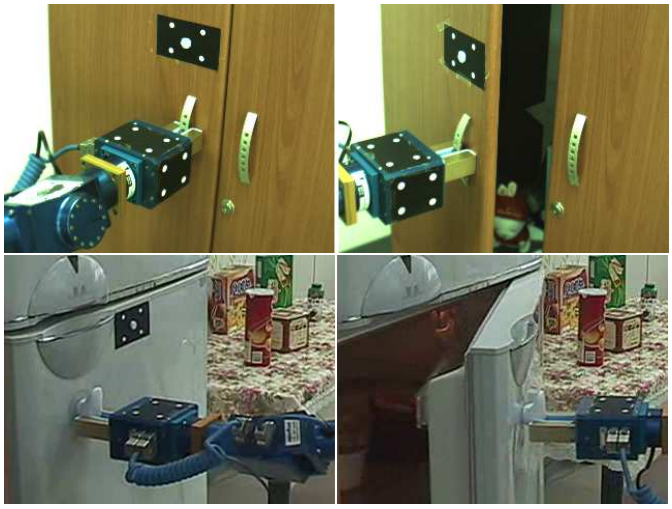

Fig. 13 Interaction phase. Top row: closet. Bottom row: refrigerator

pose w.r.t tool frame) according to equation 3, and then the robot is visually-guided in order to reduce the force. Note that it would be impossible to correct this visual positioning error without using force feedback.

4.2 Interaction

Once the handle has been reached, the robot computes the direction of the force that must be applied at the contact point (the motion constraint in the task frame), depending on the motion that must be applied to the object [1]. Thus, the reference for the force control law $\mathbf{f}^d$ is set on-line, depending on the task. However, the reference for the vision control law is not modified, because the contact (and, thus, the relative position between object and gripper) must be kept during the task execution.

The new force reference will modify the vision reference, so that the robot will move in a direction suitable for the task guided by the vision task. The natural object mechanism will generate forces on the robot hand that the force control law will try to minimize, making the robot hand to adapt to the object motion. Simultaneously, as the object pose is being observed at each iteration, any misalignment between the hand and the handle will be detected and corrected by the vision loop. Thus, both force and vision will work simultaneously for a common goal: performing the task while the relative position between hand and handle is kept constant.

Experimental results on the interaction phase can be seen in Figures 14 and 15 (only for the closet task). Figure 15 shows the evolution of the visual reference modifier $d\mathbf{s}$, which depends directly on the task forces according to equation 3. The visual reference is modified mainly in translation in Y and Z axis, and in rotation

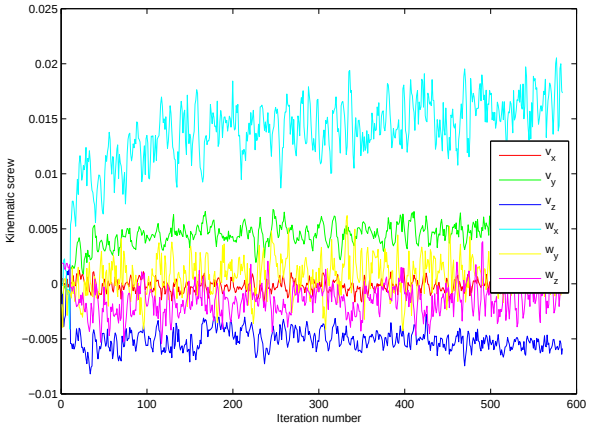

Fig. 14 Kinematic screw computed by the vision control law during the interaction phase (closet task).

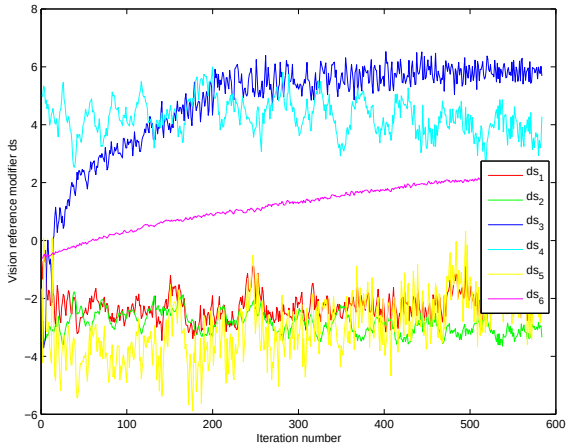

Fig. 15 Visual reference modifier based on interaction forces (closet task).

in X axis, due to the existence of important forces in these directions. Force in Z direction (of the end-effector frame $\mathcal{E}$) is regulated to a positive value according to the force reference $\mathbf{f}^d$. This force corresponds to the resistance of the particular object mechanism and is the one which is really acting on the task direction. The rest of forces appear on constrained directions and must be regulated to zero. The force in Y direction and torque in X direction are generated by the particular trajectory when opening the door. The force control law updates the vision reference so that the robot hand adapts to the natural trajectory (see the velocity in Y and the rotational velocity in X in Figure 14). It is worth noting that the hand trajectory is never planified. Instead, the vision/force control law adapts the hand motion automatically to the particular object mechanism.

5 DISCUSSION

An initial step towards vision/force-guided autonomous robotic manipulation of articulated objects has been presented. First, we have shown an object representation which is suitable for the definition of tasks under the Task Frame Formalism, and enables the use of task-oriented grasp planning algorithms. The object representation does not include a detailed geometrical model of the object. Instead, a simplified model, using bounding boxes, is used, which needs lower storage requirements, and makes grasp planning faster. However, we still have not addressed the problem of object recognition using such model.

Regarding vision and force sensors, a novel control law for coupling both modalities has been developed, based on external control [15]. The main advantage of this scheme is that the force control law is used to modify the vision reference, and not the vision control output, so that only the vision control law is moving the robot, thus avoiding problems of local minima that appear with other approaches. Although we have applied this control law to the particular case of position-based visual servoing, it can also be used with other kinds of visual servoing such as image-based or hybrid, as long as we know the interaction matrix. It is worth noting that the control law can still be improved by considering robot dynamics in the force loop, which is one of our future lines. We would also like to add more sensor modalities such as tactile sensors or proximity sensors that could add robustness to manipulation. Tactile sensors can detect contacts, even if they generate very small force, and could be used in order to correct any misalignment during grasping. Proximity sensors could be used for the same goal, but before making contact.

We have applied task-oriented grasp planning and vision/force control to a robot that must perform daily chores in a home environment. Instead of putting the camera on the hand, which may cause some problems of visibility when the hand is close to the object, an external camera has been used, which allows to have a suitable view of the object even when contact is made. In addition, this is the common configuration in current humanoid robots, where the camera is placed on the head.

An external camera also allows us to visually track the robot hand pose and to specify the grasp (and task) in terms of a desired relative pose between the hand and the object. Tracking the hand could be avoided by using joint encoders to get the hand pose with respect to the camera, assuming that the pose of the camera with respect to the robot base is known, which is a difficult calibration problem (specially when the camera is not fixed). In practice, modelling errors would generate important errors in the hand pose estimation, making this approach unfeasible. It is for this reason that we compute simultaneously the object and the hand pose and work with the relative pose between both. The main ad-

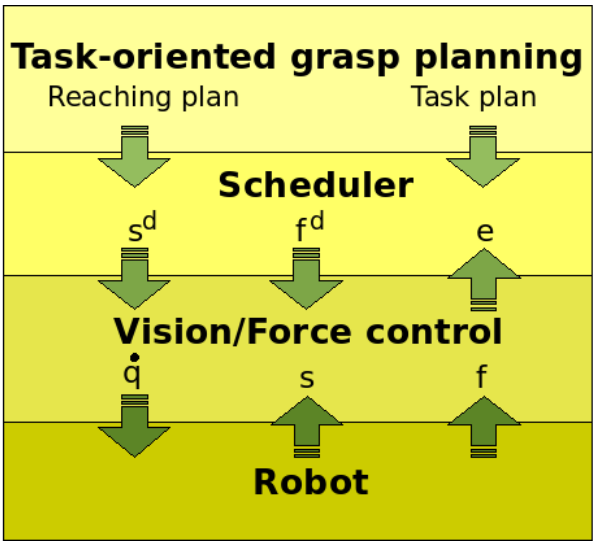

Fig. 16 General framework of the task scheduler.

vantage is that the camera can be moved freely without affecting the task execution. No external calibration is needed. At this moment, we are estimating the object and hand pose by using markers which is a quite robust and easy way for pose estimation. However, it has several disadvantages. First, whereas it may be acceptable to put a marker on the robot hand, it is not appropriate to put a marker to each object the robot has to manipulate. In addition, for certain hand-object configurations, it may be difficult to have a good view of all the points, and they can easily go out of the image. We plan to solve these problems by using natural object features for pose estimation, like in [31] or [32]. We also propose to take advantage of the independence between task execution and camera motion, by developping head control algorithms in order to move the robot eyes so that a suitable view of the hand and object is always available, according to some optimization criteria, following an active vision approach.

Regarding the visual servoing control law, the position-based approach was chosen for our experiments, mainly because it works on the cartesian space, where the grasp and task are also defined, making easier to generate the visual references from the grasp and task planning algorithms. However, it would be also possible to control the hand trajectory in cartesian space even with image-based visual servoing [33]. During the interaction phase, the robot is applying a motion on the object, and, thus, the visual features are in motion. We are currently neglecting the term that models this motion (see equation 2), and, therefore we have a tracking error, although force control can deal with it for small task velocities. It is worth noting that, due to image acquisition and processing times, the vision control frequency will be usually much smaller than the force control frequency. Thus, it is desired to run the global control law at the force sensor rate, even if the visual features are not updated at this high frequency. With this, we give priority to the force sensor feedback, and are able to detect and regulate contact at force sensor frequency, independently of the vision rate, which can vary from 25 Hz for ordinary cameras, up to 1 KHz for high-speed cameras.

Switching between the different tasks (reaching and interaction phase) is done by a task scheduler (see Figure 16) according to the error function of the visual control law ($\mathbf{e}(\mathbf{s}, \mathbf{s}^*)$). When the error is close to zero, we assume that the current step has finished, and update the vision and force references in order to perform the following step given by the task-oriented grasp planner. This clearly has the disadvantage of a discontinuity in the velocity signal when switching from one task to another. Our aim is to integrate current developments into a general control architecture. We have already worked on a control architecture for compliant execution of manipulation tasks [34]. The tasks presented in this article, could be implemented as behaviors into this architecture, so that the robot could make use of them according to a global plan. Another interesting approach is the task sequencing paradigm [35], which allows to activate/deactivate a set of small subtasks (such as avoiding obstacles, joint limits, maximizing manipulability, etc.) in order to reach a global task by taking profit of the robot redundancy.

6 CONCLUSIONS

An integrated sensor-guided robotic manipulation system for common everyday tasks has been presented. The system combines a task-oriented grasp planning algorithm with advanced visual/force servoing capabilities. The task-oriented grasp planning module computes a grasp on the object taking into account the task to perform. An external position-based visual servoing approach is used in order to visually guide the hand of the robot towards the object to grasp. During this step, the robot's head is continuously tracking the hand and the object. The relative pose between both is computed at each iteration independently of the camera position, which makes our approach amenable to be integrated into current humanoid robots without hand-eye calibration. Finally, the task is executed by means of a novel vision/force coupling approach which avoids control problems by making the integration in sensor space. Both vision and force feedback cooperate during task execution in order to keep the relative pose between gripper and object at the same time that the natural object mechanism is tracked. As future work, we would like to develop a feature extraction module in order to use the natural object features as input to the virtual visual servoing pose estimator. We would also like to work on head control algorithms in order to keep always a good view of the gripper and object during task execution. Task scheduling can also

be improved for taking into account joint limits, obstacles, and other kind of task-relevant criteria. Finally, we want to test the system on many different objects and mechanisms that future humanoid robots will have to deal with.

7 ACKNOWLEDGEMENTS

This work is supported by the Intelligent Robotics Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Commerce, Industry and Energy of Korea. This work is also supported in part by the Science and Technology Program of Gyenggi Province as well as in part by the Sungkyunkwan University. And this work was also partly supported by Brain Korea 21 (BK21) project and by the Korean Ministry of Information and Communication, as well as by Generalitat Valenciana (Spain), under grant CTBPRB/2004/052. The authors want to thank the INRIA Lagadic team for the ViSP software [36].

References

1. M. Prats, A.P. del Pobil, and P.J. Sanz. Task-oriented grasping using hand preshapes and task frames. In *Proc. of IEEE International Conference on Robotics and Automation*, pages 1794–1799, Rome, Italy, April 2007.
2. Y. Mezouar, M. Prats, and P. Martinet. External hybrid vision/force control. In *Intl. Conference on Advanced Robotics (ICAR'07)*, Jeju, Korea, 2007.
3. A. Petrovskaya and A.Y. Ng. Probabilistic mobile manipulation in dynamic environments with application to opening doors. In *International Joint Conference on Artificial Intelligence (IJCAI'07)*, Hyderabad, India, January 2007.
4. P.F. Dominey, A. Mallet, and E. Yoshida. Progress in programming the hrp-2 humanoid using spoken language. In *International Conference on Robotics and Automation (ICRA'07)*, Rome, Italy, April 2007.
5. M. Ito, K. Noda, Y. Hoshino, and J. Tani. Dynamic and interactive generation of object handling behaviors by a small humanoid robot using a dynamic neural network model. *Neural Networks*, 19(3):323–337, 2006. ISSN: 0893-6080.
6. R. Bajcsy. *Integrating vision and touch for robotic applications*. Trends and Applications of AI in Business, ed. W. Reitman, Ablex Publ. Co, 1984.
7. K. Hosoda, K. Igarashi, and M. Asada. Hybrid visual servoing / force control in unknown environment. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1097–1103, Osaka, Japan, 1996.
8. B.J. Nelson and P.K. Khosla. Force and vision resolvability for assimilating disparate sensory feedback. *IEEE Trans. on Robotics and Automation*, 12(5):714–731, 1996. ISSN: 1042-296X.
9. G. Morel, E. Malis, and S. Boudet. Impedance based combination of visual and force control. In *IEEE International Conference on Robotics and Automation (ICRA'98)*, volume 2, pages 1743–1748, Leuven, Belgium, May 1998.
10. J. Baeten and J. De Schutter. *Integrated Visual Servoing and Force Control: The Task Frame Approach*. Springer, 2003. ISBN: 3-540-40475-9.
11. W. Khalil and E. Dombre. *Modeling identification and control of robots*. Hermes Penton Science, 2002. ISBN: 1-9039-9613-9.
12. R.P. Horaud, F. Dornaika, and B. Espiau. Visually guided object grasping. *IEEE Transactions on Robotics and Automation*, 14(4):525–532, August 1998. ISSN: 1042-296X.
13. G. Taylor and L. Kleeman. Flexible self-calibrated visual servoing for a humanoid robot. In *Proc. of the Australian Conference on Robotics and Automation*, pages 79–84, Sydney, Australia, November 2001.
14. E. Marchand and F. Chaumette. Virtual visual servoing: a framework for real-time augmented reality. In *EUROGRAPHICS 2002*, volume 21(3), pages 289–298, Saarebrücken, Germany, September 2002.
15. V. Perdereau and M. Drouin. A new scheme for hybrid force-position control. *Robotica*, 11:453–464, 1993. ISSN: 0263-5747.
16. M. Mason. Compliance and force control for computer-controlled manipulators. *IEEE Trans on Systems, Man, and Cybernetics*, 11(6):418–432, 1981.
17. H. Bruyninckx and J. De Schutter. Specification of force-controlled actions in the 'task frame formalism': A synthesis. *IEEE Transactions on Robotics and Automation*, 12(5):581–589, 1996. ISSN: 1042-296X.
18. T. Kröger, B. Finkemeyer, U. Thomas, and F.M. Wahl. Compliant motion programming: The task frame formalism revisited. In *Mechatronics & Robotics*, Aachen, Germany, September 2004.
19. Z. Li and S. Sastry. Task oriented optimal grasping by multifingered robot hands. In *Proc. IEEE Intl. Conference on Robotics and Automation*, pages 389–394 vol.4, 1987.
20. Ch. Borst, M. Fischer, and G. Hirzinger. Grasp Planning: How to Choose a Suitable Task Wrench Space. In *Proc. IEEE Intl. Conference on Robotics and Automation*, pages 319–325, New Orleans, LA, USA, April 2004.
21. R. Haschke, J.J. Steil, I. Steuwer, and H. Ritter. Task-oriented quality measures for dextrous grasping. In *IEEE Conference on Computational Intelligence in Robotics and Automation*, Espoo, 2005.
22. A.T. Miller, S. Knoop, H.I. Christensen, and P.K. Allen. Automatic grasp planning using shape primitives. In *Proc. IEEE Intl. Conference on Robotics and Automation*, pages 1824–1829, Taipei, Taiwan, September 2003.
23. S. Hutchinson, G.D. Hager, and P.I. Corke. A tutorial on visual servo control. *IEEE Trans. on Robotics and Automation*, 12(5):651–670, October 1996. ISSN: 1042-296X.
24. P. Martinet and J. Gallice. Position based visual servoing using a nonlinear approach. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 531–536, Kyongju, Korea, October 1999.
25. B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation*, 8(3):313–326, June 1992. ISSN: 1042-296X.
26. E. Malis, F. Chaumette, and S. Boudet. 2 1/2 d visual servoing. *IEEE Trans. on Robotics and Automation*, 15(2):238–250, April 1999. ISSN: 1042-296X.
27. Y. Hu, R. Eagleson, and M.A. Goodale. Human visual servoing for reaching and grasping: The role of 3-d geometric features. In *Proc. IEEE Intl. Conference on Robotics and Automation*, pages 3209–3216, Detroit, Michigan, USA, 1999.
28. B. Thuilot, P. Martinet, L. Cordesses, , and J. Gallice. Position based visual servoing : keeping the object in the field of vision. In *IEEE International Conference on*

Robotics and Automation (ICRA'02), pages 1624–1629, Washington DC, USA, May 2002.
29. H. Jang, H. Moradi, S. Hong, S. Lee, and J. Han. Spatial reasoning for real-time robotic manipulation. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2632 – 2637, Beijing, China, October 2006.
30. S. Lee, S. Lee, J. Lee, D. Moon, E. Kim, and J. Seo. Robust recognition and pose estimation of 3d objects based on evidence fusion in a sequence of images. In *Proc. of IEEE International Conference on Robotics and Automation*, Rome, Italy, April 2007.
31. A. Comport, E. Marchand, and F. Chaumette. Complex articulated object tracking. *Electronic Letters on Computer Vision and Image Analysis*, 5(3):21–31, 2005.
32. T. Drummond and R. Cipolla. Real-time visual tracking of complex structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):932–946, 2002. ISSN: 0162-8828.
33. Y. Mezouar and F. Chaumette. Optimal camera trajectory with image-based control. *Int. Journal of Robotics Research*, 22(10):781–804, October 2003.
34. M. Prats, A.P. del Pobil, and P.J. Sanz. A control architecture for compliant execution of manipulation tasks. In *Proc. of International Conference on Intelligent Robots and Systems*, pages 4472–4477, Beijing, China, October 2006.
35. N. Mansard and F. Chaumette. Task sequencing for high level sensor-based control. *IEEE Trans. on Robotics*, 23(1):60–72, February 2007. ISSN: 1042-296X.
36. 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(4):40–52, December 2005.