Autonomous Robots manuscript No.
(will be inserted by the editor)

Reliable non-prehensile door opening through the combination of vision, tactile and force feedback

Mario Prats · Pedro J. Sanz · Angel P. del Pobil

Received: date / Accepted: date

Abstract Whereas vision and force feedback – either at the wrist or at the joint level – for robotic manipulation purposes has received considerable attention in the literature, the benefits that tactile sensors can provide when combined with vision and force have been rarely explored.

In fact, there are some situations in which vision and force feedback cannot guarantee robust manipulation. Vision is frequently subject to calibration errors, occlusions and outliers, whereas force feedback can only provide useful information on those directions that are constrained by the environment. In tasks where the visual feedback contains errors, and the contact configuration does not constrain all the cartesian degrees of freedom, vision and force sensors are not sufficient to guarantee a successful execution.

Many of the tasks performed in our daily life that do not require a firm grasp belong to this category. Therefore, it is important to develop strategies for robustly dealing with these situations. In this article, a new framework for combining tactile information with vision and force feedback is proposed and validated with the task of opening a sliding door. Results show how the vision-tactile-force approach outperforms vision-force and force-alone, in the sense that it allows to correct the vision errors at the same time that a suitable contact configuration is guaranteed.

M. Prats
Computer Science and Engineering Department
Jaume-I University, Castellón, Spain
E-mail: mprats@icc.uji.es

P.J. Sanz
Computer Science and Engineering Department
Jaume-I University, Castellón, Spain
E-mail: sanzp@icc.uji.es

A.P. del Pobil
Computer Science and Engineering Department
Jaume-I University, Castellón, Spain
and
Department of Interaction Science
Sungkyunkwan University, Seoul, South Korea
E-mail: pobil@icc.uji.es

1 INTRODUCTION

Robotic mobile manipulation is a field which has to deal with significant uncertainties when considering real scenarios. Except some specific cases (Petrovskaya and Ng, 2007), errors in sonar and laser-based localization are far from the precission needed for most manipulation tasks. Trying to reach an object just based on localization information would lead to unacceptable errors in the final hand position. There is a need for additional sensing capabilities that are able to provide a more accurate object pose with respect to the robot. Vision is the most suitable sensor for this purpose. It potentially allows to locate quite precisely the target object in the environment and to track its motion in the case of movable objects. Although at the contact level it is possible to precisely perform a manipulation task without the need for vision, it is still very helpful in order to perceive the effects of our interaction with the world. However, for most manipulation cases, only part of the object is visible in the image, which, in addition, can be easily occluded by the robot hand and subject to illumination changes. This normally leads to a poor visual feature set, from which it is almost impossible to estimate an accurate object pose. All of these errors result in a hand misalignment with respect to the object, which can lead the task to failure. The ability of a robot to properly use its sensors for dealing with such uncertainties is completely necessary for ensuring a successful manipulation. In particular, force and tactile feedback are the main sources of information that humans use for robust physical interaction.

In this paper, we develop an approach for combining vision, force and tactile feedback with applications to manipulation in household environments. We first analyze the accuracy of a vision-based pose estimation method for manipulation tasks, and conclude that significant errors in the estimated pose can appear in some singular cases. In order to deal with such errors, we propose to combine the visual information with real-time feedback coming from tactile and force sensors, leading to a vision-tactile-force control approach. The new approach provides the robot with a rich sensory experience which is used for performing a door opening task in a robust manner. The same task is performed when only vision and force feedback is available, and results show how the use of tactile information highly increases the robot performance.

In our approach, vision and tactile information is combined first, togheter with object-robot localization information, and then used as input for a stiffness force controller. The vision controller implements a position-based visual servoing approach (Martinet and Gallice, 1999), based on the Virtual Visual Servoing (VVS) method for model-based pose estimation and tracking of articulated objects (Marchand and Chaumette, 2002). The tactile controller is in charge of up to three cartesian degrees of freedom (d.o.f's), allowing to keep always a good contact. Force control is performed through a programmed stiffness, which allows to deal with external forces on all the cartesian directions. The redundant nature of the sensors allows to perform the task even if a sensor is not available or provides innacurate data. Throughout this article, the terms *sensor integration* and *sensor combination* will be used without distinction to refer to the control-level combination of different sensor feedback, in contrast to *sensor fusion* which is normally understood as the combination at the sensor level.

1.1 Related work

Many works have considered the use of vision for manipulation purposes (Kragic and Christensen, 2002; Stemmer et al, 2006; Dune et al, 2008). In general, vision is normally used in order to obtain an estimation of the object position and orientation that allows the robot to perform a specific action with it. Pose estimation techniques in robot vision can be classified in appearance-based or model-based approaches (Lepetit and Fua, 2005). Appearance-based methods work by comparing the 2D image of the object with those stored in a database containing previously acquired views from multiple angles. The main advantage of these methods is that they do not need a 3D object model, although a previous process must be performed in order to include a new object in the database. Model-based methods obtain better accuracy and robustness, because of the use of model information for anticipating events like object self-occlusions. Some approaches consider a combination of both methods, like (Kragic and Christensen, 2002), where an appearance-based method is used first for getting an initial pose estimation, which is then used as initialization for a model-based algorithm.

Force sensing has been also adopted for estimating the 6 d.o.f pose of objects in the environment, normally following state estimation techniques based on probabilistic approaches. For example, Bruyninckx et al (2003) adopted techniques already developed in the SLAM community for both mapping and localization of objects in the context of autonomous manipulation. A similar approach was adopted by Petrovskaya et al (2006), where a new approach called Scaling Series Particle Filter (SSPF) was developed for estimating the complete pose of polygonal objects, from contacts estimated from a force sensor. These methods have the advantage of estimating an accurate hand-object relative pose in a robust manner, even when the initial uncertainty is very high. Our approach differs from these methods mainly in that we focus on the task execution part. We propose a reactive controller in order to keep a good contact configuration while the task is being performed, without necessarily knowing the full pose of the manipulated object, and without any kind of replanning or model update. We assume that vision provides a initial pose estimation which is suitable enough for approaching the hand to the part of the object to be manipulated. In the cases where vision is not available or its information is highly innaccurate, localization methods as the ones previously mentioned could be adopted as a previous step to our approach, or even simultaneously.

In order to deal with the uncertainties inherent to vision processing, the use of vision for robotic manipulation is normally considered jointly with force feedback. Some approaches have adopted either passive compliance at the robot joints (Edsinger and Weber, 2004) or torque control on torque-controlled manipulators (Wyrobek et al, 2008; Albu-Schaffer et al, 2008). However, the most frequent method is still active force control, since it can be implemented on standard manipulators. To combine visual and force information at the control level, two main approaches (impedance-based and hybrid-based strategies) have been studied (Hosoda et al, 1996; Nelson and P.K.Khosla, 1996; Morel et al, 1998; Baeten et al, 2003). In these schemes the idea is merely to replace the classical position controller (Khalil and Dombre, 2002) 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 from both vision and force feedback. 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. Recently, external vision-force control was proposed (Mezouar et al, 2007; Prats et al, 2008), which makes the combination in sensor-space, allowing to control vision and force on all the degrees of freedom, whereas only the vision control law is directly connected to the robot.

Whereas vision and force integration techniques have been extensively studied in the literature, tactile information has been rarely considered jointly with those sensors. Touch is the ability to sense at the finger-object level (Howe, 1994), and it has been shown that people have difficulties to perform manipulation tasks when deprived of tactile feedback (Johansson and Westling, 1984). It is known that there are about 17000 mechanoreceptors distributed along the fingers and the palm of the human hand, which provide rich information, mainly about the contact distribution, limb motion and forces (Howe, 1994). Several attempts to mimick the human sense of touch exist (Tegin and Wikander, 2005), being the tactile array sensors – able to perceive the pressure distribution of the contact and the local shape – the most common approach.

Vision-tactile-force combination was already addressed in (Allen et al, 1999), where some guidelines for detecting useful manipulation events with these sensors were given, without addressing the robot control problem. In (Son et al, 1996), an approach for combining vision and tactile feedback in a control law was proposed, where force along one direction was also considered, and measured from the tactile sensors. A recent paper (Schmid et al, 2008), makes a comparison between tactile-alone, force-alone and force-tactile integration in the task of opening a door, where vision is used in a previous step in order to detect the door handle.

1.2 Summary of our approach

Our approach differs from the previous ones in the following aspects:

- First, instead of an ad-hoc vision processing algorithm, we make use of the VVS approach (Marchand and Chaumette, 2002) for model-based pose estimation and tracking of articulated objects. This makes our approach amenable to be used for different objects and tasks with little modification of the vision part, and provides a robust estimation, based on a well-established theory. We assume that this method provides a suitable initial pose estimation that allows to reach for the object without requiring contact-based localization techniques.
- Second, we consider a dedicated force/torque sensor providing 6 degrees of freedom, instead of a one-dimensional force computed from the tactile sensor (tactile-based force) or hand strain gauge. In addition, contacts are also detected with a dedicated tactile sensor, instead of estimating the contact position from the force information (Petrovskaya et al, 2006). This would not be possible in our case due to the lack of accurate models of the hand and fingers, and the high noise and low resolution of the force sensor signal.
- Finally, we consider full vision-tactile-force combination, in the sense that all the three sensors can be present at the same time. In addition, sensor combinations such as vision-force, tactile-force and vision-tactile are also allowed. The task is performed in a reactive manner, without any kind of replanning or model update.

The VVS pose estimation method is outlined in Section 2, and its suitability for two typical cases in mobile manipulation is analyzed. The first corresponds to the case where an object is fully visible in the image, either because it has a small size, or because it is seen from a far position. The second case consists of a big object seen from a close position, suitable for reaching and manipulation. It is shown how the second case normally lacks enough image information for a robust vision-based manipulation. Then, in Section 3, we propose to integrate the vision-based controller with other controllers built up from tactile and force feedback. As tactile sensors provide very accurate and robust contact information, they are used in order to correct the hand-object misalignments generated during vision-based reaching. In addition, force control is implemented as a programmable stiffness at the robot hand, allowing to deal with undesired forces generated by those misalignments. Some experiments are performed in Section 4, involving the opening of a cabinet door when image information is not enough for computing an accurate object pose. Results show how the addition of tactile information allows to robustly deal with situations that cannot be controlled with just vision and force feedback.

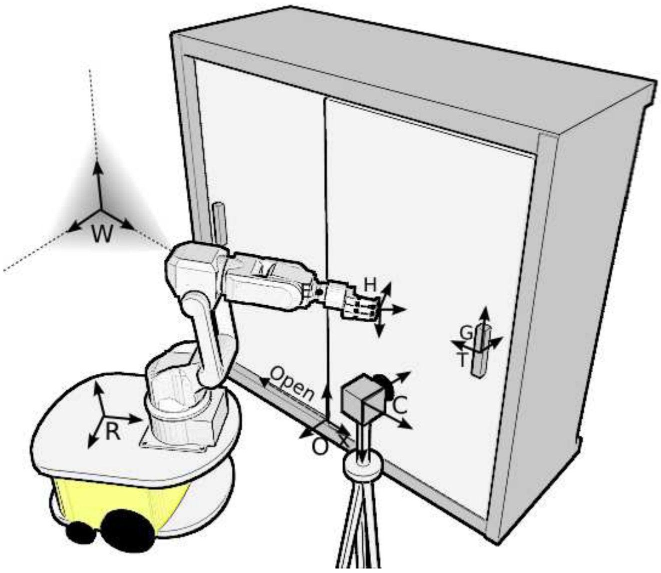


Fig. 1 The experimental environment consists of a mobile manipulator, an external camera and a cabinet with a sliding door that must be pushed open to the left. The mobile manipulator is located at frame R given with respect to the world frame, W. The camera is at frame C, and calibrated with respect to R. The object frame, O is supposed to be known in world coordinates. Finally, frames H, G and T denote the hand, grasp and task frames, and are given with respect to R and O respectively. They are used as auxiliary entities for grasp and task description and execution (Prats et al, 2010).

1.3 Mobile manipulation environment

Our mobile manipulation environment consists of a mobile robot with manipulation capabilities, placed in front of the object that has to be manipulated, as shown in Figure 1. The task considered in this article is to open a sliding door by pushing towards the left. This is a particularly interesting task for two reasons. The first is that it is a common task performed continuously in human environments, and, therefore, of great interest from the service robotics point of view. The second reason, more interesting from a scientific point of view, is that the task does not require a firm grasp, and, therefore, not all the cartesian d.o.f's are constrained. Thus, force sensors cannot report misalignments on the unconstrained directions, and, therefore, it is difficult to keep a suitable contact configuration just by vision and force feedback.

It is assumed that the mobile robot has map-based localization capabilities so that it is possible to know approximately the robot pose in the map world coordinates, which will be denoted by the homogeneous transformation matrix ${}^{W}\mathbf{M}_{R}$. It is out of the scope of this article to define a specific localization method. Instead, the reader is referred to the literature, where several robot localization approaches have been proposed, with different accuracy depending on the type of sensors used for measuring the environment features: sonar (Drumheller, 1987), laser (Castellanos et al, 1996), robot vision (Se et al, 2001), ceiling cameras (Broxvall et al, 2006), etc. Using one or a combination of these methods, the robot can be localized quite precisely inside a map of the environment, which can be acquired previously, or simultaneously to the localization process (Durrant-Whyte, 2006). In addition, it is also assumed that the target object is included inside the map, at ${}^{W}\mathbf{M}_{O}$, and that a CAD model of the object exists, including significant features such as vertex, edges and joints in the case of articulated objects.

We consider a camera, located at frame C, and linked with the robot base frame through ${}^{R}\mathbf{M}_{C}$. This homogeneous transformation is calibrated in our case, although it could also be computed from robot kinematics in the case of a humanoid head kinematically linked to the body, for example. It is also assumed that the camera intrinsic parameters are known from a previous calibration step.

We assume that a task-oriented grasp planning algorithm, based on our previous work (Prats et al, 2007b, 2010), exists, providing the following information:

- A task-oriented hand preshape (Prats et al, 2007b) suitable for performing the particular grasp and task.
- A hand frame, H, attached to the part of the hand used for the grasp, and known with respect to the manipulator end-effector frame through hand kinematics. This homogeneous transformation will be denoted by ${}^{E}\mathbf{M}_{H}$.
- A grasp frame, G, attached to the part of the object where the hand must be moved to, and expressed with respect to the object frame, O, through the homogeneous transformation matrix ${}^{O}\mathbf{M}_{G}$.

The manipulation task consists of two steps: reaching and interaction. Reaching is specified as a desired transformation to achieve between the hand and the grasp frame, denoted by ${}^{H}\mathbf{M}^{*}_{G}$, whereas interaction is determined by a force reference, ${}^{T}\mathbf{f}^{*}$, given in a task frame, T, which is related to the object with ${}^{O}\mathbf{M}_{T}$, and aligned with the natural object motion constraints, as specified in the Task Frame Formalism (Bruyninckx and Schutter, 1996). In this article, we will focus on the interaction part. For previous work on vision-based reaching, refer to (Prats et al, 2008).

2 Vision-based pose estimation for mobile manipulation

It is known that humans use 3D information rather than 2D features for vision-based manipulation (Hu et al, 1999). In addition, the use of 3D information allows to easily integrate the vision part with the task and grasp planning algorithms. For these reasons, we adopt a model-based approach for vision-based pose estimation and manipulation.

There are two main methods in the literature for model-based pose estimation and tracking of articulated objects, both based on full-scale non-linear optimization. The first, developed by Drummond and Cipolla (2002), is formulated from the Lie algebra point of view, whereas the second, proposed by Comport et al (2004b,a), is based on the Virtual Visual Servoing (VVS) method (Marchand and Chaumette, 2002). Both methods implement robust estimation techniques and have shown to be very suitable for real-time tracking of common articulated objects in real environments. A comparison between both approaches is reported in (Comport et al, 2005), where it is shown that both formulations are equivalent, although some differences in performance can appear at run time. In our mobile manipulator, we have implemented the VVS approach (Comport et al, 2004a; Marchand and Chaumette, 2002), mainly for its computational efficiency and because it is based on a solid background theory, i.e. 2D visual servoing, which convergence conditions, stability, robustness, etc. have been widely studied in the visual servoing community (Hutchinson et al, 1996). In addition, almost any kind of visual feature can be used and combined with this approach (points, lines, ellipses, etc.), as long as the corresponding interaction matrix can be computed. Different examples of the interaction matrix for the most common features are shown in (Espiau et al, 1992).

In the following sections, we study the advantages and limitations of the VVS approach for manipulation tasks, and propose a sensor integration scheme that allows to use force and tactile information to complement the vision sensor when the data it provides is not complete or inaccurate.

2.1 The concept

The concept of the VVS approach, developed in (Marchand and Chaumette, 2002), is to apply visual servoing techniques to a virtual camera, so that a set of object features projected in the virtual image from a model, match with those extracted from the real image. Under this approach, the pose estimation and tracking problem can be seen as equivalent to the problem of 2D visual servoing (Comport et al, 2004b), which has been extensively studied in the visual servoing community (Hutchinson et al, 1996). Taking as input an object model, and an initial estimation of the camera pose in object coordinates, denoted as a pose vector, $\mathbf{r}$, the idea is to project a set of 3D features of the object model into a virtual image of the object, taken from the virtual camera position, $\mathbf{r}$. This virtual image is compared with the real one, and a vector of visual features is generated, denoted by $\mathbf{s}(\mathbf{r})$.

In our particular implementation, we make use of the point-to-line distance feature, as in (Comport et al, 2004b), although any kind of geometric feature could be used as long as the interaction matrix can be computed. The edges of the object model, projected as lines in the virtual image, are sampled at regular intervals, and a search for a strong gradient is performed in the real image, in a direction perpendicular to the projected line, as shown in Figure 2. For each match, the point-to-line distance is computed and stored in the feature vector. The desired feature vector is given by $\mathbf{s}^* = 0$, which represents the case when all the edges of the object model are projected on strong gradients, and, ideally, the virtual camera position corresponds to the real one. The control law governing the virtual camera motion is given by:

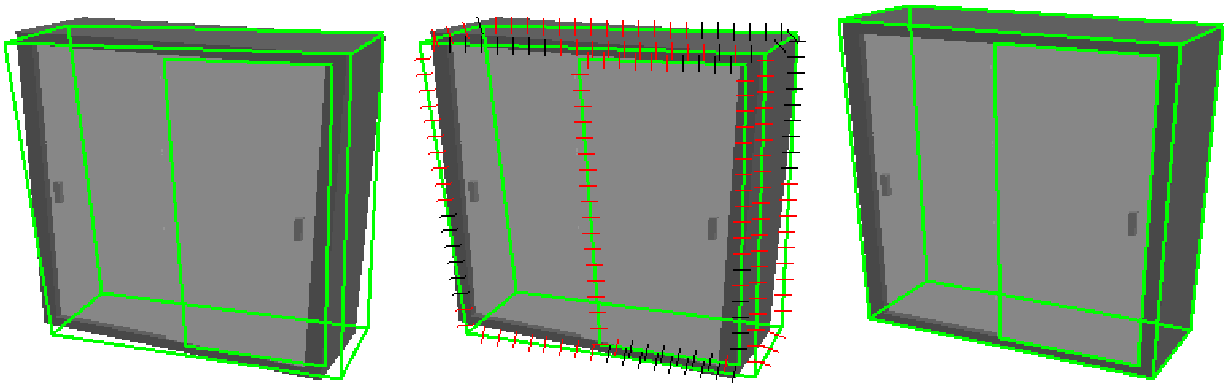
$$\mathbf{v}_r = -\lambda \left(\widehat{\mathbf{D}}\widehat{\mathbf{L}_s}\right)^+ \widehat{\mathbf{D}}(\mathbf{s}(\mathbf{r}) - \mathbf{s}^*) \tag{1}$$

where $\mathbf{v}_r$ is the virtual camera velocity, λ is a control gain, $\widehat{\mathbf{L}_s}$ is the interaction matrix for the point-to-line distance feature (Comport et al, 2004b), and $\widehat{\mathbf{D}}$ is a diagonal weighting matrix computed by iteratively reweighted least squares, which is a robust estimator for dealing with outliers.

2.2 Virtual Visual Servoing on articulated objects

Comport et al (2004a) presented an approach for pose estimation and tracking of articulated objects based on the VVS method and the kinematic set concept. In their approach, the articulated pose is estimated directly from the visual observation of the object parts, leading to an efficient method that eliminates the propagation of errors through the kinematic chain. The only condidition is that joint parameters must be decoupled in the minimization of the objective function. This can be accomplished by performing the minimization in object joint coordinates instead of in the camera space. Let $\mathbf{s}_1(\mathbf{r}_1)$ and $\mathbf{s}_2(\mathbf{r}_2)$ represent the perceived visual features on both parts of an articulated object composed of two links and one joint, and $\mathbf{s}_1^*$ and $\mathbf{s}_2^*$ be the desired values for those features, with $\widehat{\mathbf{L}_{s1}}$ and $\widehat{\mathbf{L}_{s2}}$ representing the corresponding interaction matrices. Then, the articular pose can be estimated by applying the following image-based control law:

$$\begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{pmatrix} = -\lambda \widehat{\mathbf{A}} \left(\widehat{\mathbf{D}}\widehat{\mathbf{H}}\right)^+ \widehat{\mathbf{D}} \begin{pmatrix} \mathbf{s}_1(\mathbf{r}_1) - \mathbf{s}_1^* \\ \mathbf{s}_2(\mathbf{r}_2) - \mathbf{s}_2^* \end{pmatrix} \tag{2}$$



(a) Initial estimation

(b) Point-line distance minimization

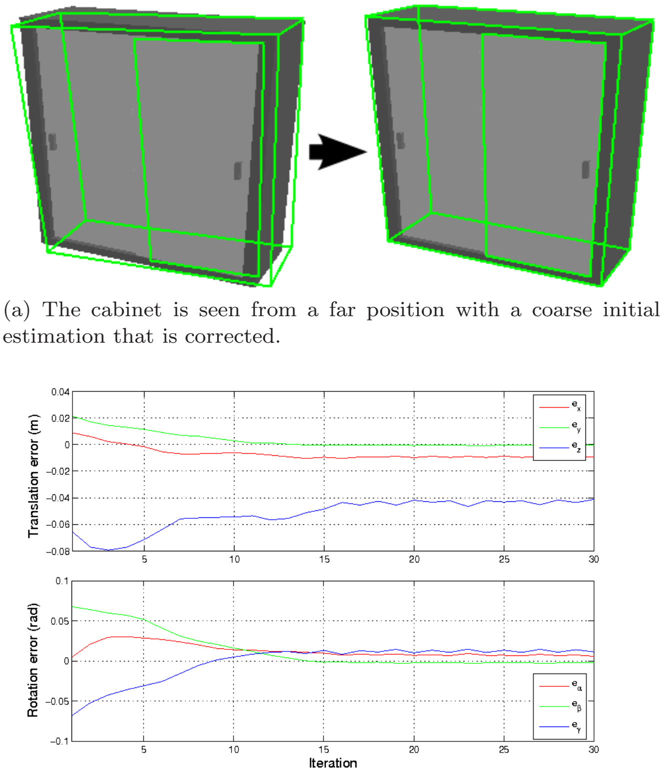
(c) Final estimation

Fig. 2 An outline of the VVS-based pose estimation approach based on the point-to-line distance feature. A feature vector is built from the distances between the projected edges and high-gradient points searched along the edge normals, at the sampling interval. The goal of the non-linear minimization is to reduce all the distances to zero.

$$\widehat{\mathbf{H}} = \begin{pmatrix} \widehat{\mathbf{L}_{s1}} & \mathbf{0} \\ \mathbf{0} & \widehat{\mathbf{L}_{s2}} \end{pmatrix} \widehat{\mathbf{A}}$$

$$\widehat{\mathbf{A}} = \begin{pmatrix} \widehat{{}^{C}\mathbf{W}_{O}}\mathbf{S} & \widehat{{}^{C}\mathbf{W}_{O}}\mathbf{S}^{\perp} & \mathbf{0} \\ \widehat{{}^{C}\mathbf{W}_{O}}\mathbf{S} & \mathbf{0} & \widehat{{}^{C}\mathbf{W}_{O}}\mathbf{S}^{\perp} \end{pmatrix}$$

where $\widehat{{}^{C}\mathbf{W}_{O}}$ represents the twist transformation matrix from the camera frame to the object joint frame, and $\mathbf{S}^{\perp}$ is a constraint matrix which depends on the type of joint (Comport et al, 2004a). Finally, the virtual camera velocities, one for each link, are given by $\mathbf{v}_1$ and $\mathbf{v}_2$.



(a) The cabinet is seen from a far position with a coarse initial estimation that is corrected.

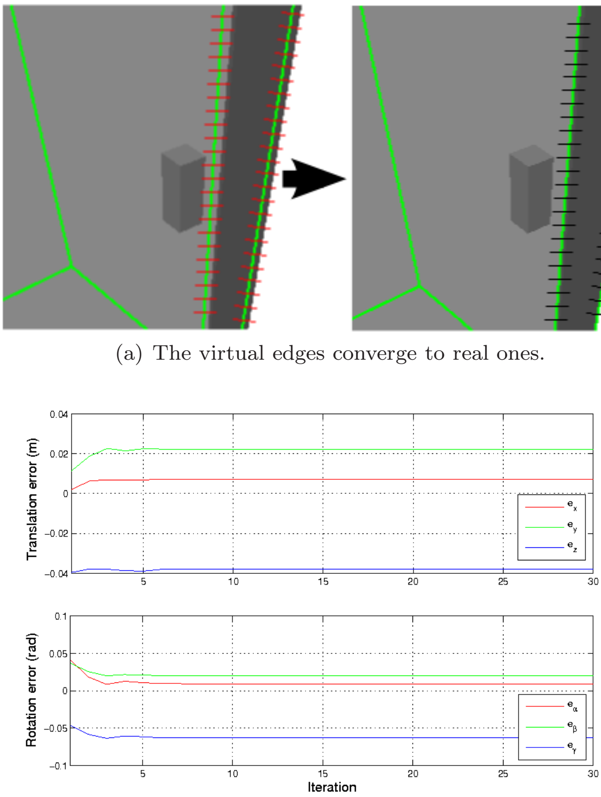
(b) The pose estimation converges to the real value up to a small error that corresponds to the typical accuracy given by vision on an object seen from a far position.

Fig. 3 VVS convergence in the case where there is enough information in the image and the interaction matrix is full rank.

2.3 Limitations

The main limiting factor affecting the convergence of the VVS method is local minima, which depends on the kind of features considered and its observability in the image. In general, the interaction matrix should be full rank in order to be able to compute a solution for equations 1 and 2. This is the typical case for objects that are fully observable from the camera point of view. Figure 3 shows a simulation of a cabinet seen from a far position. In order to study the convergence of the VVS method in this case, significant errors have been manually introduced in the initial camera position, simulating robot localization errors. Results show how VVS is able to improve the initial estimation, and converges to the real pose, up to a small error. The reason is that a rich set of visual features is available, which ensures the full rank of matrix $\widehat{\mathbf{L}_s}$.

However, this is difficult to achieve in a manipulation environment, where the robot is close to the target object, and only a small part of it is visible. As an example, Figure 4 shows the same cabinet seen from a position suitable for manipulation purposes. In this case, only the right edges of the cabinet and the door are visible. Although the handle features could also be considered in the simulation environment, we have not used them because of the difficulty to extract them robustly in a real case, apart from the fact that, during manip-



(a) The virtual edges converge to real ones.

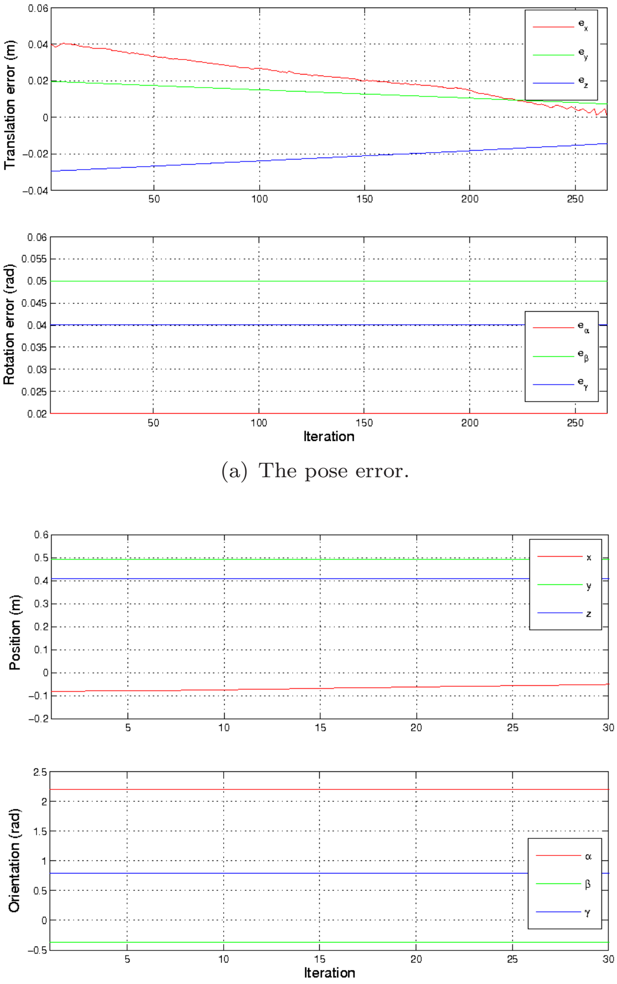
(b) The pose estimation does not converge.

Fig. 4 VVS convergence in the case where there is not enough information in the image and the interaction matrix is not full rank.

ulation, the handle features are normally occluded by the hand. In addition, the handle can be different from one door to another, whereas the door edges are always present. In this experiment, the interaction matrix computed from the set of point-to-line features extracted from the visible edges is not full-rank, which means that there is some ambiguity and multiple solutions are possible. It is worth noting that this is not a limitation of the method, but a result of the particular visual conditions. In fact, without considering the handle, it would be ambiguous also for the human eye. Figure 4 shows how the VVS method converges to a situation where some d.o.f's are even worse than the initial estimation, even though the projected edges correspond to the real ones.

Fortunately, this problem can be detected by continuously checking the rank of the interaction matrix. At the moment that some d.o.f's are lost, it is possible to fix the parent object pose and track only the articulated part which normally needs only one or two d.o.f's. If the parent object motion is not considered, equation 2 takes the following form:

$$\mathbf{v}_2 = -\lambda {}^{C}\widehat{\mathbf{W}}_O \mathbf{S}^{\perp} \left(\widehat{\mathbf{D}}\widehat{\mathbf{L}}_{s_2} {}^{C}\widehat{\mathbf{W}}_O \mathbf{S}^{\perp}\right)^{+} \widehat{\mathbf{D}}(\mathbf{s}(\mathbf{r}_2) - \mathbf{s}_2^*) \tag{3}$$



(a) The pose error.

(b) Evolution of the camera pose in object coordinates.

Fig. 6 Tracking of a sliding door along one DOF.

Figures 5 and 6 show the case where there is not enough information for estimating the full 6D pose, but it is still possible to track the articulated part along one translational d.o.f. Even though the articulated d.o.f. can be successfully tracked, the initial error on the rest of d.o.f's cannot be corrected, because the parent object edges are not considered in the minimization.

In summary, VVS can provide accurate 6D localization in cases where a big part of the target object is visible, so that there is enough information in the image for ensuring a full rank interaction matrix. However, in cases where only part of the object is visible, pose ambiguities can appear, leading to significant errors in the object pose estimation. Unfortunately, this is a common case in mobile manipulation, due to the proximity of the robot to the object.

At the moment of manipulation, these errors are manifested in misalignments between the hand and the object, which can lead the task to failure. In order to

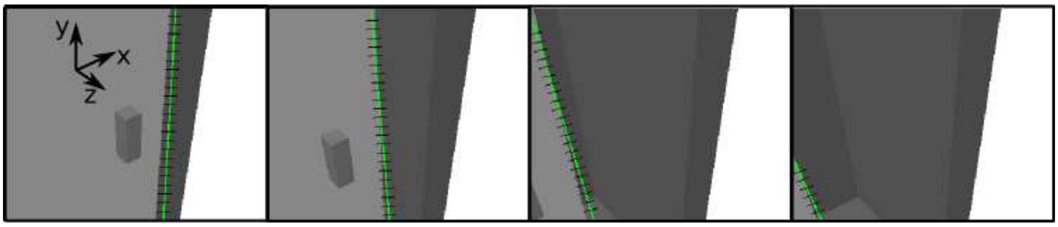


Fig. 5 Tracking of the articulated part when the parent object is considered fixed at a position with a manually introduced error. As the prismatic joint only involves 1 DOF, tracking can be performed even with a rank-deficient interaction matrix.

deal with those misalignments, force feedback is normally introduced. However, force sensors can only provide complete information about misalignments when the hand motion is fully constrained by the environment, i.e. under a firm grasp. In tasks like pushing where a firm grasp is not required, those directions that are tangent to the pushing direction and are not constrained, do not generate any force information (neglecting friction) that can be used for reducing the misalignment. In addition, vision is not accurate enough for detecting the lack of contact. There is a need of additional sensor information on these directions. In order to manage these situations, in the next section, we propose to complement the control signal coming from vision and force with that coming from tactile array sensors.

3 Vision-Tactile-Force Control

We propose a position-vision-tactile hybrid controller modified by a stiffness force control term, as shown in Figure 7. In the context of this scheme, position control is understood as that motion based on environment information obtained by mobile robot localization algorithms, either based on laser, sonar, odometry, intelligent environment, etc. Only one sensor between localization, vision and tactile sensors is used for a given cartesian direction. Our approach is to use the one which provides the most accurate and robust information for that direction.

Thus, we establish a sensor hierarchy where tactile information is preferred over vision feedback, which is also preferred over localization information. The reason is that tactile sensors provide the most robust and detailed information about the object position, although at the contact level, whereas vision provides more global, but less accurate data, and localization is normally the most innacurate source. The cartesian d.o.f's assigned to each sensor are set online by three selection matrices, $\mathbf{S}_p$, $\mathbf{S}_v$ and $\mathbf{S}_t$, which must be orthogonal each other, i.e. $\mathbf{S}_p \perp \mathbf{S}_v \perp \mathbf{S}_t \perp \mathbf{S}_p$. Then, if tactile information can be used for a given d.o.f, it is indicated in the corresponding diagonal element of the selection matrix, for example $\mathbf{S}_t = \mathbf{diag}\,(0,0,1,0,0,0)$ for the Z axis. If not, vision feedback will be adopted if possible. If neither tactile nor vision information is available, then the controller will rely on localization information. If a given cartesian direction must be explicitly controlled by force, it can be set to 0 on all the selection matrices, so that the force controller will fully take charge of it. Being ${}^{H}\mathbf{v}_p$, ${}^{H}\mathbf{v}_v$ and ${}^{H}\mathbf{v}_t$, the control velocity computed respectively by the position controller, the vision controller, and the tactile controller, all of them given in the hand frame, H, then the result of the preliminary sensor integration is given by:

$$ {}^{H}\mathbf{v}_{pvt} = \mathbf{S}_p \cdot {}^{H}\mathbf{v}_p + \mathbf{S}_v \cdot {}^{H}\mathbf{v}_v + \mathbf{S}_t \cdot {}^{H}\mathbf{v}_t \tag{4} $$

It is worth mentioning that, in our approach, the selection matrices act on the control velocities, and not on the input errors as in the original hybrid control concept. This is because the tactile and vision errors are not necessarily defined in the cartesian space, and thus, the selection matrices cannot be applied directly on them. Instead, they are applied after the corresponding controllers, where all the control signals are given in a common frame. Note that this is the common practice in hybrid vision-force control approaches (Nelson et al, 1995). The control velocity of expression 4 is then modified by a stiffness force controller which acts on all the degrees of freedom, ensuring that any force generated by a misalignment of the controlled frame, H, with respect to the environment will be kept inside a given range. If ${}^{H}\mathbf{v}_f$ is the hand velocity computed by the force controller, the final velocity signal, given in the robot end-effector frame can be computed as:

$$ {}^{E}\mathbf{v}_{pvtf} = {}^{E}\mathbf{W}_H \cdot \left({}^{H}\mathbf{v}_{pvt} + {}^{H}\mathbf{v}_f\right) \tag{5} $$

where ${}^{E}\mathbf{W}_H$ is the twist transformation matrix between the hand frame H, and the end-effector frame, E. This approach leads to a very natural behavior, where force is the most important sensor, followed by tactile, vision, and localization sensors. Under a blind situation, the task can still be performed by position-tactile-force integration. If tactile feedback is not available, as for example in the phase of reaching an object, then

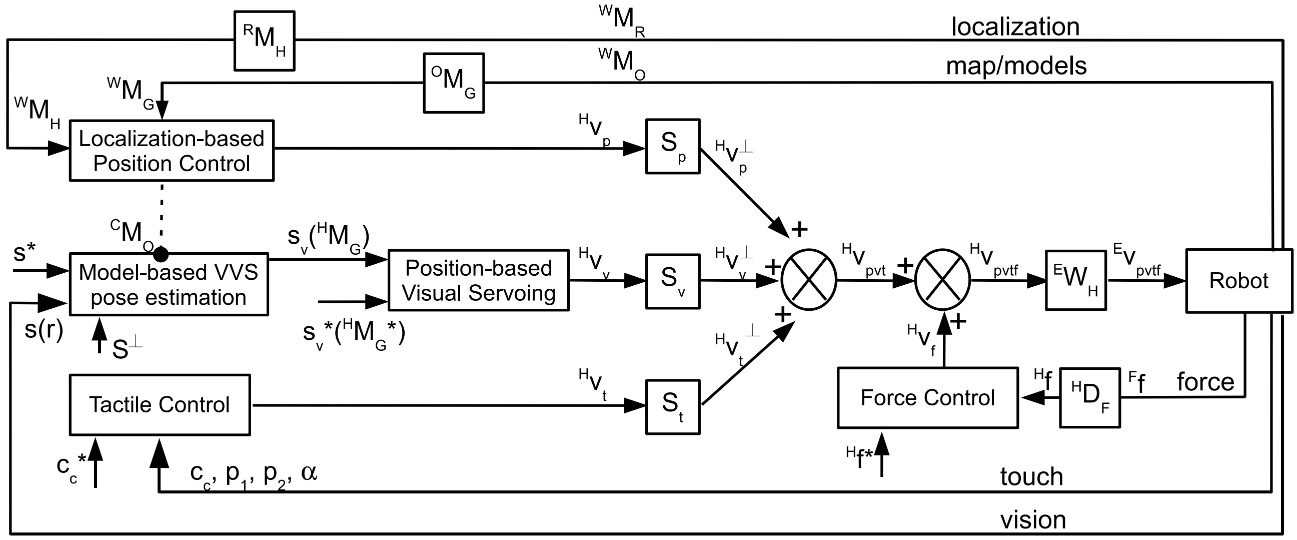


Fig. 7 Our control approach, integrating position, vision, tactile and force feedback.

position-vision-force can successfully guide the hand. In the worst case where tactile and vision are unavailable, position-force can still be used. In the following, the particular position, vision, tactile and force controllers used in our experiment are described.

3.1 Position Controller

In our approach, the localization-based position controller is in charge of computing the required hand velocity, based on world-to-robot and world-to-object information provided by localization algorithms. First, the hand frame is transformed to world coordinates as ${}^W\mathbf{M}_H = {}^W\mathbf{M}_R \cdot {}^R\mathbf{M}_E \cdot {}^E\mathbf{M}_H$, where ${}^R\mathbf{M}_E$ is the end-effector frame pose with respect to the robot base, which is assumed to be known from the robot kinematic model. The grasp frame is also transformed to world coordinates as ${}^W\mathbf{M}_G = {}^W\mathbf{M}_O \cdot {}^O\mathbf{M}_G$. Then, the hand-to-grasp relationship can be computed as:

$$ {}^H\mathbf{M}_G = \left({}^W\mathbf{M}_H\right)^{-1} \cdot {}^W\mathbf{M}_G \tag{6} $$

and a proportional position-based control can be performed with the following equation, where λ_p is the control gain, and ${}^H\mathbf{h}^*$ is a pose vector build from the homogeneous matrix ${}^H\mathbf{M}_H^*$ (i.e. ${}^H\mathbf{M}_H^* = {}^H\mathbf{M}_G \cdot \left({}^H\mathbf{M}_G^*\right)^{-1}$):

$$ {}^H\mathbf{v}_p = \lambda_p {}^H\mathbf{h}^* \tag{7} $$

This simple control law drives the hand in a straight line in order to reach the desired relative pose between the hand frame and the localization-based estimated pose of the grasp frame on the target object.

3.2 Vision controller

The object pose estimation provided by the VVS process, is used to compute a more accurate hand-to-grasp relationship, as:

$$ {}^H\mathbf{M}_G = \left({}^C\mathbf{M}_H\right)^{-1} \cdot {}^C\mathbf{M}_O \cdot {}^O\mathbf{M}_G \tag{8} $$

where ${}^C\mathbf{M}_H$ is assumed to be known from camera external calibration and robot kinematics (i.e. ${}^C\mathbf{M}_H = {}^C\mathbf{M}_R \cdot {}^R\mathbf{M}_E \cdot {}^E\mathbf{M}_H$), whereas ${}^C\mathbf{M}_O$ is the object pose estimation computed by the method described in the previous section. As full 3D information is available, we have opted for a position-based visual servoing (Martinet and Gallice, 1999) in order to obtain straight trajectories in cartesian space. The ${}^H\mathbf{M}_G$ matrix is given as input and used for setting the visual feature vector to $\mathbf{s}_v = (\mathbf{t} \quad \mathbf{u}\theta)^T$, where $\mathbf{t}$ is the translational part of the ${}^H\mathbf{M}_G$ homogeneous matrix, and $\mathbf{u}\theta$ is the axis/angle representation of the rotational part. Similarly, the desired feature vector $\mathbf{s}_v^*$ is set from the desired hand-to-grasp relationship ${}^H\mathbf{M}_G^*$, which can be either planned or learnt. The hand velocity, as computed by the position-based visual servoing control law, is given by:

$$ {}^H\mathbf{v}_v = -\lambda_v \widehat{\mathbf{L}_{\mathbf{s}_v}^{+}}(\mathbf{s}_v - \mathbf{s}_v^*) \tag{9} $$

where the following interaction matrix is chosen for the particular case of position-based visual servoing (Martinet and Gallice, 1999):

$$ \widehat{\mathbf{L}_{\mathbf{s}_v}} = \begin{pmatrix} -\mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & -\mathbf{L}_w \end{pmatrix} \tag{10} $$

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

$[\mathbf{u}]_{\times}$ denotes the skew-symmetric matrix associated to the rotation axis $\mathbf{u}$. It is worth noting that the vision controller is independent of the method used for estimating the object pose. Even though the VVS method has been adopted in this work, another different method could be used as long as it is suitable for inclusion in a control loop. Regarding the stability conditions of the position-based visual servoing approach, the reader is referred to (Martinet and Gallice, 1999).

3.3 Tactile Controller

The tactile controller designed for our experiments looks for the alignment between the robot fingertips and a planar surface, such as a handle. Although it has been specifically designed for our particular system and task, it could be easily adapted to a different case. The goal is to provide a control velocity for the set of cartesian directions which can be robustly and accurately controlled with tactile information. Depending on the sophistication of the tactile sensors, the sensor distribution, the hand configuration, and the task, more or less directions could be controlled.

We consider a Barrett Hand with one tactile array sensor on each fingertip, providing pressure distribution and magnitude information in a 8x15 pressure matrix, as shown in Figure 8. First, the biggest contact blob on sensors 1 and 2 is selected and its centroid is computed, giving the points $\mathbf{c}_1 = \left(\mathbf{c}_{1_x}, \mathbf{c}_{1_y}\right)$ and $\mathbf{c}_2 = \left(\mathbf{c}_{2_x}, \mathbf{c}_{2_y}\right)$ in the sensor frame. The maximum pressure sensed on each of the two contact blobs are denoted as p_1 and p_2. The point $\mathbf{c}_c = \left(\mathbf{c}_{c_x}, \mathbf{c}_{c_y}\right)$ is computed as the middle point between $\mathbf{c}_1$ and $\mathbf{c}_2$. Finally, α is computed as the angle between the line joining $\mathbf{c}_1$ and $\mathbf{c}_2$ and the vertical.

Three cartesian d.o.f's at the hand frame (H in Figure 8) are controlled in order to accomplish three goals:

– First, rotation around X axis is controlled in order to guarantee that the pressure is equally distributed between the tactile sensors, thus ensuring that all the tactile sensors keep the contact:

$$^{H}\mathbf{v}_{r_x} = K_p\left(p_2 - p_1\right) \tag{11}$$

– Second, rotation around Z axis is also controlled in order to regulate α to zero. The goal is to be aligned with the handle.

$$^{H}\mathbf{v}_{r_z} = K_\alpha \alpha \tag{12}$$

– Finally, translation along X axis is controlled in order to bring the point $\mathbf{c}_c$ towards a reference $\mathbf{c}_c^* = \left(\mathbf{c}_{c_x}^*, \mathbf{c}_{c_y}^*\right)$, which indicates the part of the tactile sensor where to keep the contact:

$$^{H}\mathbf{v}_{t_x} = -K_c\left(\mathbf{c}_{c_x} - \mathbf{c}_{c_x}^*\right) \tag{13}$$

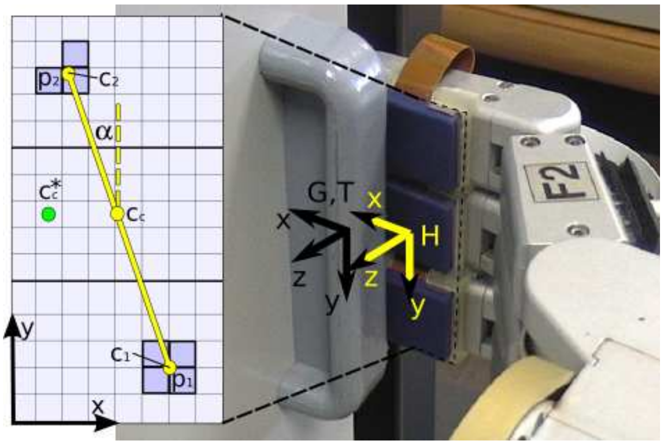


Fig. 8 The Barrett Hand in a hook precision preshape, with the tactile sensors installed at the fingertips. H is the hand frame, and G denotes the grasp frame. The biggest contact blob on sensors 1 and 2 is selected, and the centroid of these contacts are computed, together with the maximum pressure on each sensor and the angle between the contact line and the vertical.

K_p, K_α and K_c are the control gains for each controlled direction. The velocity on the rest of directions is set to zero:

$$^{H}\mathbf{v}_t = \left(^{H}\mathbf{v}_{t_x}, 0, 0, {}^{H}\mathbf{v}_{r_x}, 0, {}^{H}\mathbf{v}_{r_z}\right) \tag{14}$$

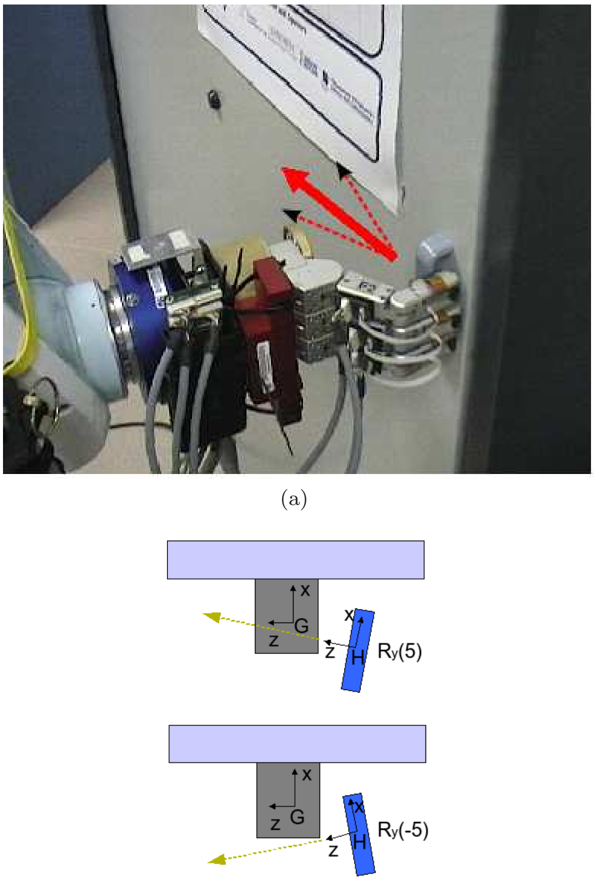
The selection matrix for the tactile controller is set to $\mathbf{S}_t = \mathbf{diag}\left(1,0,0,1,0,1\right)$ for our particular case. In the cases where tactile information is not available, such as in the phase of reaching, $\mathbf{S}_t$ can be set to zero so that the hand is controlled by position-vision-force integration.

In conclusion, when enough tactile information is available, tactile control can ensure that an accurate alignment between the hand and the handle is kept, by controlling just three cartesian d.o.f's, although it would be possible to control additional d.o.f's in the case of more advanced tactile sensors or different alignment tasks. It is worth mentioning that by observing contact over time it would be possible to control additional d.o.f's with our tactile sensors, such as rotation in Y axis.

3.4 Force Controller

Finally, an active stiffness control (Salisbury, 1980) is performed on top of the other controllers, where a force reference has been included in the feedback loop without affecting the control law stability, as it can be viewed as a reference trajectory modifier (Morel and Bidaud, 1996; Lasky and Hsia, 1991):

$$^{H}\mathbf{v}_f = \mathbf{K}_f^{-1}(^{H}\mathbf{D}_F \cdot {}^{F}\mathbf{f} - {}^{H}\mathbf{f}^*) \tag{15}$$


(a)

(b)

Fig. 9 The task is to push open a sliding door under manually introduced errors in the initial estimation of the grasp link. For a rotational positive error around Y axis of the grasp frame, G, the hand motion has a positive component along the X axis of the real grasp frame, which finally leads to a frontal contact if not corrected. For a negative error, the robot pushes along a direction which has a negative component along the X axis of the real grasp frame.

where $\mathbf{K}_f$ is the stiffness matrix, ${}^F\mathbf{f}$ represents the force measured at each iteration at the force sensor frame, F, and ${}^H\mathbf{f}^*$ is the force reference, used for pushing in the task direction. ${}^H\mathbf{D}_F$ represents the wrench transformation matrix between frames F and H (i.e. ${}^H\mathbf{D}_F = {}^H\mathbf{W}_F^T$)(Khalil and Dombre, 2002). The force reference, ${}^H\mathbf{f}^*$, is computed from the task reference, ${}^T\mathbf{f}^*$ (i.e. ${}^H\mathbf{f}^* = {}^H\mathbf{D}_T {}^T\mathbf{f}^*$), which can be set to zero on those directions where a passive behavior is desired, but must take a value for the task direction.

4 Experiments

In order to study the benefits that tactile feedback provides when complemented with vision and force information, a manipulation task is performed with an articulated object when there is not enough information in the image features for an accurate vision-based 6D pose estimation.

More concretely, the task is to open a cabinet door (of sliding type) further than 25 cm. The robot was manually moved in front of a cabinet as shown in Figures 1 and 9. The camera (with focal length $(344.00, 334.23)$ and image center $(140.46, 126.74)$ at a resolution of 320×240) was placed in order to get a view of the cabinet door, and a coarse estimation of the homogeneous matrix describing the relationship between the camera frame and the robot base frame (${}^R\mathbf{M}_C$) was calibrated by attaching a pattern to the robot hand and computing its pose with the Dementhon algorithm (Dementhon and Davis, 1995), as in (Prats et al, 2007a, 2008), and then making use of the robot kinematic model. Note that this step would not be necessary in a humanoid system, for example, where the eye-to-hand relationship can be approximately computed through robot kinematics.

The initial door pose in the camera frame, ${}^C\mathbf{M}_O$, was coarsely calibrated in our case, although it could be also computed from robot laser and sonar-based localization algorithms. Figure 10 shows a sequence captured by the robot camera during the execution. Note that only the left and right edges of the door are visible from the camera position, leading to a feature set which is not rich enough for getting a full rank interaction matrix. For this reason, the parent object is considered as fixed, and only the door is tracked along one translational d.o.f, by setting $\mathbf{S}^{\perp} = \mathbf{diag}(1,0,0,0,0,0)$ in equation 3. The vision selection matrix is set to $\mathbf{S}_v = \mathbf{diag}\,(0,1,1,0,1,0)$, whereas $\mathbf{S}_p = \mathbf{0}$.

For opening the door, a hook precission preshape (Prats et al, 2007b) was adopted, as shown in Figure 8. The hand frame, H was set to the inner part of the robot fingertips, whereas the grasp frame, G, was set to the handle, according to our previous work on a framework for specifying physical interaction tasks (Prats et al, 2010).

Apart from the errors generated by the poor calibration of the initial camera-robot and camera-object transformation, rotational errors of up to 5 degrees were manually added in the initial estimation of the object pose, on each of the three cartesian axis. Even under these significant errors, vision-based tracking of the articulated part along the articulated d.o.f succeeded in all the cases, when considering the parent object as fixed. Obviously, the errors on non-articulated d.o.f's were propagated as explained in section 2.3.

For each error (positive and negative), on each axis, the task was executed, first by using only the force sensor, then adding the vision modality, and finally by a combination of vision, force and tactile sensors.

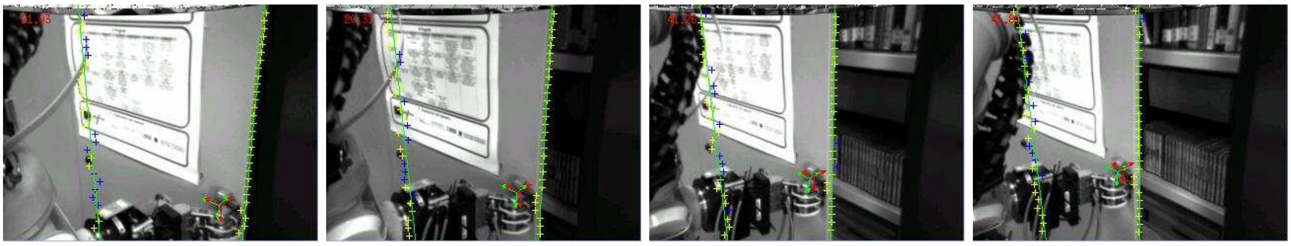


Fig. 10 The vision effects of introducing an error around Y axis in the pose of the grasp frame. The pose estimation method is able to track the articulated pose, but errors on the rest of directions cannot be corrected (note that the left edge estimation does not correspond to the real one).

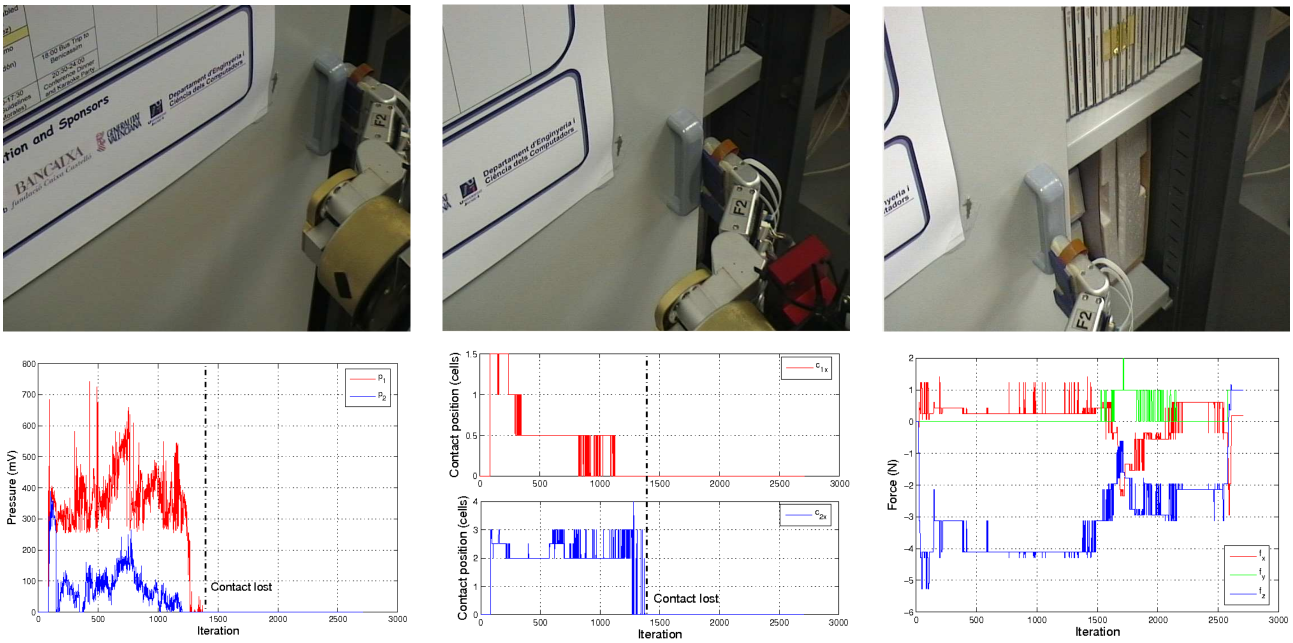


Fig. 11 Vision-force performance for an error of -5 degrees in Y axis of the handle pose. Top row: three snapshots of the interaction task, where it is shown how contact is lost during execution. Bottom row, from left to right: pressures at fingertips (p_1 and p_2), X component of the contact centroids (c_{1_x} and c_{2_x}) and forces in the hand frame (${}^{H}\mathbf{f}$)

Therefore, a total of 18 trials were performed, 6 for force-alone, 6 for vision-force and 6 for vision-force-tactile, and a trial was considered as a failure when the robot was unable to open the door further than 25 cm. When only the force controller was activated, the experiments succeeded in just 3 out of 6 trials. Vision-force completed the task in 5 experiments, and vision-force-tactile performed well in all the 6 cases. In addition, the vision-force-tactile combination was the only one able to avoid undesired forces in directions other than the task direction. In all the failures, the reason was the missing of contact between the hand and the handle, due to rotation misalignments that generated hand motion on directions tangential to the pushing direction.

Detailed results for the interesting case of a rotation error around Y axis are shown in Figures 11, 12 and 13, where the control parameters were chosen experimentally as follows: $\lambda_p = \lambda_v = 0.3$, $K_p = 5 \cdot 10^{-5}$, $K_\alpha = 0.1$, $K_c = 0.004$, $c^*_{c_x} = 1.5$, $\mathbf{f}^* = (0, 0, -5N, 0, 0, 0)$, $\mathbf{K}_f^{-1} = \mathbf{diag}\left(5 \cdot 10^{-4}, 5 \cdot 10^{-4}, 15 \cdot 10^{-4}, 0, 0, 0\right)$.

In the case of Figures 11 and 12, the introduced error is manifested in a misalignment that makes the robot push along a direction which has a negative component along the X axis of the real grasp frame, G, as shown in Figure 9. In this configuration, position constraints exist only along the frontal direction (X axis of the hand frame) and the opening direction (Z axis of the hand frame). As the rest of directions are not position-constrained, misalignments on these axis do not generate external forces, and, thus, cannot be detected and controlled with force feedback. Similarly, the vision part is running with an initial estimation which is wrong, and thus, the articulated pose estimation still contains the initialization error. Thus, an opening strategy using only vision and force sensors would easily

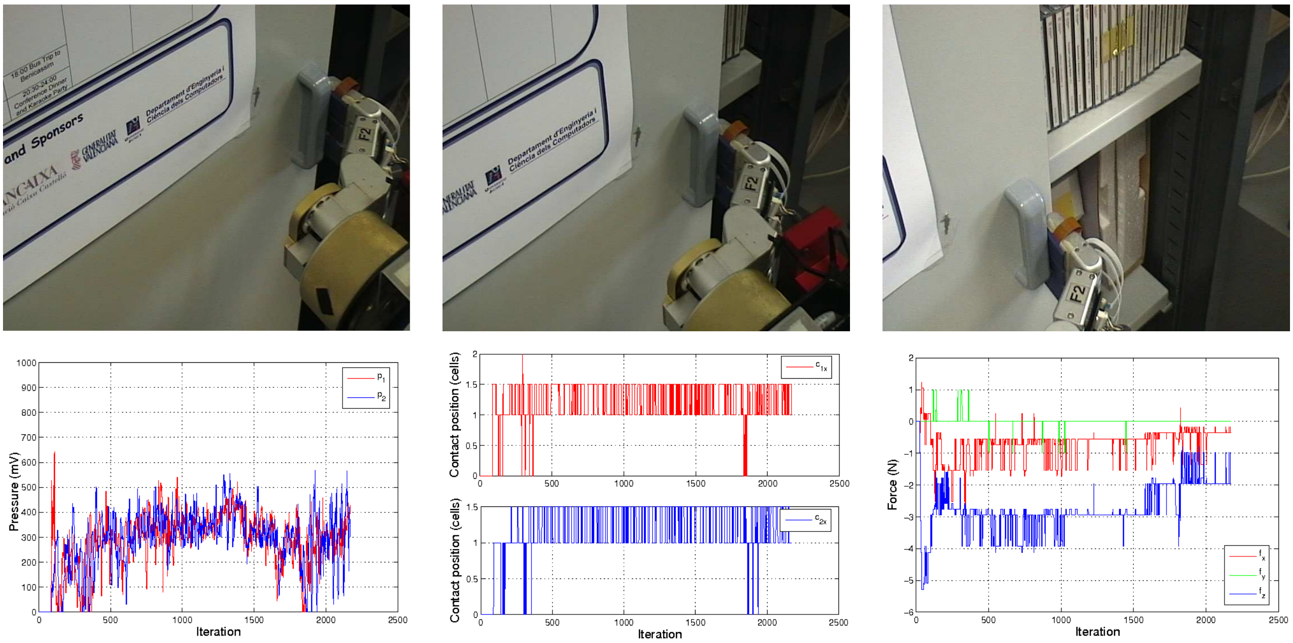


Fig. 12 Vision-force-tactile performance for an error of -5 degrees in Y axis of the handle pose. Pressure is balanced and contact is kept until the end of the task. Top row: three snapshots of the interaction task. Bottom row, from left to right: pressures at fingertips (p_1 and p_2), X component of the contact centroids (c_{1_x} and c_{2_x}) and forces in the hand frame (${}^{H}\mathbf{f}$)

lose contact, as shown in Figure 11. Vision-force-tactile, however, is able to perceive contact information, and controls the robot so that a contact is always present at the desired location in the fingertip (Figure 12). The force disturbances in the case of vision-force are due to a frontal contact of the finger with the handle, which generates a small frontal force, appearing at the moment of losing the contact.

It is also worth noting that vision-force-tactile is able to balance the pressure between the fingertips, whereas vision-force is not able to detect and control this issue. This is clearly shown in Figure 13, which shows the case of a positive rotation error around Y axis in the localization of frame G. In this case, the pushing direction has a small positive component in X axis, which slowly drives the fingertip towards the door, as shown in Figure 9. If only vision sensors were considered, frontal collision could not be detected, causing damage to the robot and the door. Figure 13 shows the behavior of vision-force and vision-force-tactile control in this case. Note that, under vision-force, there is contact only with one fingertip since the very beginning (13.a-b), and vision-force is not able to correct this misalignment. Consequently, the whole task force is made by only one finger, which has to support a high pressure, increasing the risk of sensor or mechanics damage, and decreasing the overall reliability. Vision-force-tactile, however, is able to balance the pressure, ensuring contact with all the fingers (Figure 13.d-e). Even if vision-force is finally able to complete the task, note that, as a consequence of the initial introduced error, the fingertip finally makes frontal contact with the door, leading to a high force in the frontal direction that exists almost from the beginning of the execution (13.c). As expected, vision-force-tactile avoids this situation, keeping the pressure level, contact position and forces inside a normal range.

Figure 14 shows a sequence of the vision-tactile-force execution [1]. It is shown how, starting from an initial position with significant alignment errors, vision-tactile-force integration is able to correct them and converge to a robust and safe configuration where all the fingertips are in contact and aligned with the handle. The main factor affecting the success of the vision-force-tactile strategy is the accuracy in the initial hand-object positionning after reaching. As long as there is contact with one of the tactile sensors, the tactile controller is activated and, thus, the error is reduced. However, if the initial error is so large that contact is not generated, the approach would fail. These situations can appear, for example, when the mobile platform localization is the only available information, with typical errors of several centimeters. In these situations, state estimation techniques could be adopted for improving the initial positioning (Bruyninckx et al, 2003).

[1] A video illustrating the different experiments performed is available at `http://www.robot.uji.es/lab/plone/Members/mprats/clips`

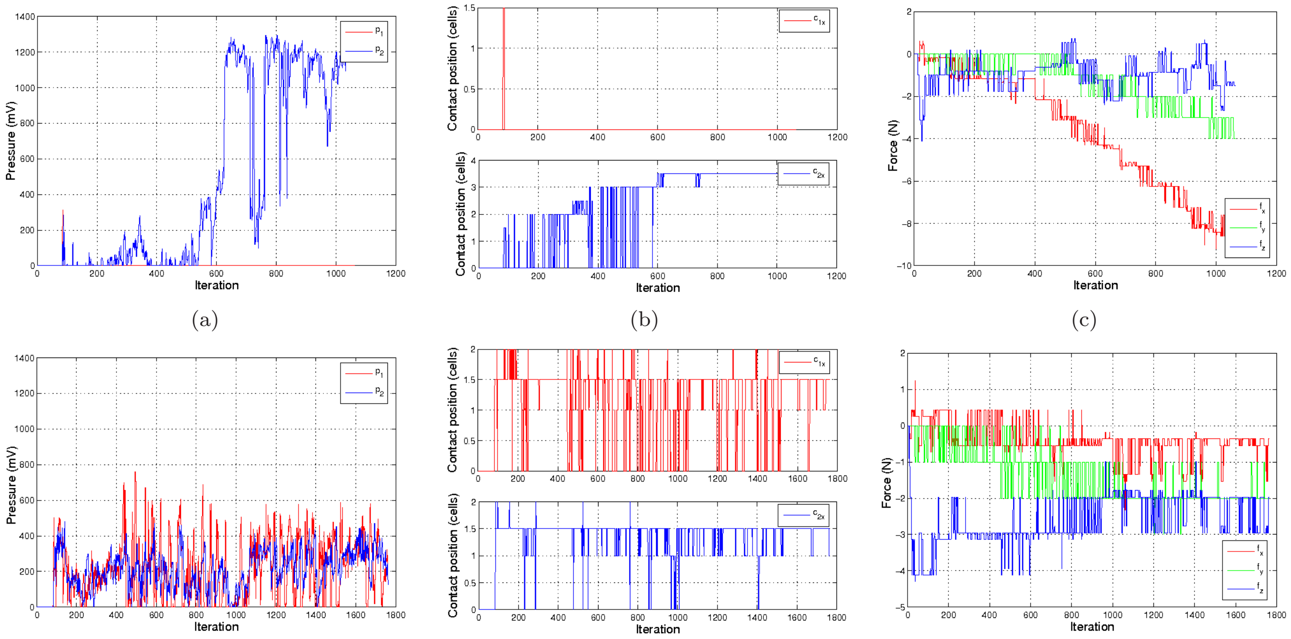


Fig. 13 Results for an error of +5 degrees in Y axis of the handle pose. Vision-force control is not able to completely align the hand. Vision-force-tactile aligns the hand successfully and distributes the pressure between the fingertips. Top row: vision-force; Bottom row: vision-tactile-force. From left to right column: pressures at fingertips (p_1 and p_2), X component of the contact centroids (c_{1_x} and c_{2_x}) and forces in the hand frame (${}^H\mathbf{f}$).

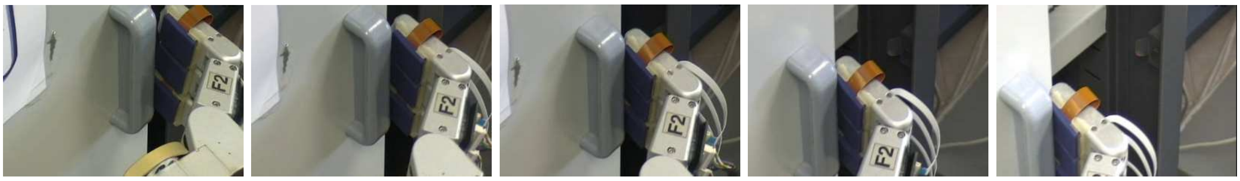


Fig. 14 A sequence showing the vision-tactile-force alignment process during task execution, starting from a coarse initial position.

5 Discussion

We have presented a new approach for integrating tactile feedback with vision and force information, with views to reliable manipulation in household environments. A position-based visual servoing approach takes the output of a vision-based articular pose estimation algorithm and visually guides the robot hand for the given task. A stiffness force controller locally modifies the hand trajectory in order to minimize external forces due to small misalignments. Finally, a tactile controller is in charge of continuously looking for a stable contact configuration by controlling 3 cartesian d.o.f's.

In our particular experiments, we give priority to tactile sensors over vision and position information. However, it could be different in other cases, like (Petrovskaya and Ng, 2007), for example, where localization algorithms provide very accurate pose information. In order to select which sensor controls each d.o.f, the respective selection matrices $\mathbf{S}_p$, $\mathbf{S}_v$ and $\mathbf{S}_t$ can be set accordingly. Due to the limitations of our tactile sensors, and the kind of tasks considered, only 3 d.o.f's can be accurately controlled by our tactile controller in a completely reactive manner, although additional d.o.f's could be controlled through contact monitoring over time. The rest is controlled by vision, or by the position controller in case vision is not available. However, the particular tactile, vision, position and force controllers are independent of the global scheme, meaning that it would be possible to design a new tactile controller able to control 6 d.o.f's in the case of using tactile sensors which provide enough information

It is worth noting the possibility to modify the sensor assignation at run time. If, for example, vision processing fails at some time, it would be possible to remove the d.o.f's assigned to the vision controller, and assign them to the position controller, or tactile controller in case they have enough information to control them. However, this introduces the problem of identifying the sensor suitability for a given cartesian direction.

This could be managed automatically following, for example, the *sensor resolvability* approach (Nelson and P.K.Khosla, 1996), that allows to identify which sensor provides the most suitable estimation for controlling a particular cartesian direction.

Regarding the vision controller, it would be desirable to control all the 6 cartesian d.o.f's, or, at least, those which are not already controlled by the tactile controller. We have discussed in section 2.3 the difficulties to estimate a full 6D pose when the robot is ready for manipulation and the camera is close to the object. In these cases, the interaction matrix can take a very low rank, and local minima can appear in the VVS minimization process. This part could be improved by considering a wide-angle camera, or additional visual features, apart from the point-to-line distance. However, we cannot assume that a rich feature set is always available. Therefore, errors in the visual estimation can always appear, making it still necessary to use force and tactile sensors for dealing with them.

Finally, it is worth mentioning that, in the current implementation, the whole controller is running at video rate, which is about 33 Hz in our experiments (on a standard Pentium IV at 3GHz). Although this frequency is sufficient for performing a safe force control, it could be desirable to achieve a higher rate, specially when high velocities are needed. This would also allow to take dynamic effects into account. It is part of the future work to adapt the current implementation in order to run at the force sensor rate.

6 Conclusion

A vision-tactile-force integration approach has been proposed and validated in a real manipulation environment. A door opening task is executed through the combination of the control signals provided by a position controller, which has an initial coarse estimation of the object pose, a vision controller based on an articular object pose estimator, a tactile controller, which looks for not losing the contact during manipulation, and a stiffness force controller, in charge of pushing along the task direction at the same time that the force is regulated on the rest of directions. Different sensor combinations, such as force-alone, vision-force or tactile-force, are also possible in case that one or more sensors become unavailable. In order to study the advantages of adding tactile feedback to vision and force-based manipulation, several experiments have been carried out with the task of opening a sliding door under manually introduced errors. Results show how the proposed vision-tactile-force approach is able to correct the hand-object misalignments generated by an innacurate vision-based reaching, and offers a more reliable execution than that obtained when only vision and force are used.

Acknowledgements This research was partly supported by the Korea Science and Engineering Foundation under the WCU (World Class University) program funded by the Ministry of Education, Science and Technology, S. Korea (Grant No. R31-2008-000-10062-0), by the European Commission's Seventh Framework Programme FP7/2007-2013 under grant agreements 217077 (EYESHOTS project), and 248497(TRIDENT Project), by Ministerio de Ciencia e Innovación (DPI-2008-06636; and DPI2008-06548-C03-01), by Fundació Caixa Castelló-Bancaixa (P1-1B2008-51; and P1-1B2009-50) and by Universitat Jaume I. The vision part of this work was implemented using the open source ViSP library (IRISA/INRIA).

References

Albu-Schaffer A, Eiberger O, Grebenstein M, Haddadin S, Ott C, Wimbock T, Wolf S, Hirzinger G (2008) Soft robotics: From torque feedback controlled light-weight robots to intrinsically compliant systems. IEEE Robotics & Automation Magazine 15(3):20–30, DOI 10.1109/MRA.2008.927979

Allen PK, Miller A, Oh PY, Leibowitz B (1999) Integration of vision, force and tactile sensing for grasping. International Journal of Intelligent Machines 4(1):129–149

Baeten J, Bruyninckx H, Schutter JD (2003) Integrated vision/force robotic servoing in the task frame formalism. Int Journal of Robotics Research 22(10-11):941–954

Broxvall M, Gritti M, Saffiotti A, Seo B, Cho Y (2006) PEIS ecology: Integrating robots into smart environments. In: IEEE International Conference on Robotics and Automation, Orlando, FL, pp 212–218

Bruyninckx H, Schutter JD (1996) Specification of force-controlled actions in the 'task frame formalism': A synthesis. IEEE Trans on Robotics and Automation 12(5):581–589

Bruyninckx H, Schutter JD, Lefebvre T, Gadeyne K, Soetens P, Rutgeerts J, Slaets P, Meeussen W (2003) Building blocks for slam in autonomous compliant motion. In: International Symposium of Robotics Research, Siena, Italy

Castellanos J, Neira J, Strauss O, Tardos J (1996) Detecting high level features for mobile robot localization. In: IEEE International Conference on Multisensor Fusion and Integration, Washington, DC, USA, pp 611–618

Comport AI, Marchand E, Chaumette F (2004a) Object-based visual 3d tracking of articulated objects via kinematic sets. In: CVPRW '04: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 1, IEEE Computer Society, Washington, DC, USA, p 2

Comport AI, Marchand E, Chaumette F (2004b) Robust model-based tracking for robot vision. In: In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS04, pp 692–697

Comport AI, Kragic D, Marchand E, Chaumette F (2005) Robust real-time visual tracking: Comparison, theoretical analysis and performance evaluation. In: Proc. IEEE Intl. Conference on Robotics and Automation, Barcelona, Spain, pp 2852–2857

Dementhon D, Davis L (1995) Model-based object pose in 25 lines of code. International Journal of Computer Vision 15(1/2):123–141

Drumheller M (1987) Mobile robot localization using sonar. IEEE Transactions on Pattern Analysis and Machine Intelligence 9:325–332

Drummond T, Cipolla R (2002) Real-time visual tracking of complex structures. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7):932–946

Dune C, Marchand E, Collewet C, Leroux C (2008) Active rough shape estimation of unknown objects. In: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Nice, France, pp 3622–3627

Durrant-Whyte T H; Bailey (2006) Simultaneous localization and mapping: part i. IEEE Robotics & Automation Magazine 13(2):99–110

Edsinger A, Weber J (2004) Domo: a force sensing humanoid robot for manipulation research. In: Proc. 4th IEEE/RAS International Conference on Humanoid Robots, vol 1, pp 273–291 Vol. 1

Espiau B, Chaumette F, Rives P (1992) A new approach to visual servoing in robotics. IEEE Transactions on Robotics and Automation 8(3):313–326

Hosoda K, Igarashi K, Asada M (1996) Hybrid visual servoing / force control in unknown environment. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, Osaka, Japan, pp 1097–1103

Howe R (1994) Tactile sensing and control of robotic manipulation. Journal of Advanced Robotics 8(3):245–261

Hu Y, Eagleson R, Goodale M (1999) Human visual servoing for reaching and grasping: The role of 3-d geometric features. In: Proc. IEEE Intl. Conference on Robotics and Automation, Detroit, Michigan, USA, pp 3209–3216

Hutchinson S, Hager G, Corke P (1996) A tutorial on visual servo control. IEEE Transactions on Robotics and Automation 12(5):651–670

Johansson R, Westling G (1984) Roles of glabrous skin receptors and sensorimotor memory in automatic control of precision grip when lifting rougher or more slippery objects. Experimental Brain Research 56:550–564

Khalil W, Dombre E (2002) Modeling identification and control of robots. Hermes Penton Science

Kragic D, Christensen H (2002) Model based techniques for robotic servoing and grasping. IEEE/RSJ International Conference on Intelligent Robots and Systems 1:299–304

Lasky T, Hsia T (1991) On force-tracking impedance control of robot manipulators. In: IEEE International Conference on Robotics and Automation, Sacramento, California, vol 1, pp 274–280, DOI 10.1109/ROBOT.1991.131587

Lepetit V, Fua P (2005) Monocular model-based 3d tracking of rigid objects. Foundations and Trends in Computer Graphics and Vision 1(1):1–89

Marchand E, Chaumette F (2002) Virtual visual servoing: a framework for real-time augmented reality. In: EUROGRAPHICS 2002, Saarebrücken, Germany, vol 21(3), pp 289–298

Martinet P, Gallice J (1999) Position based visual servoing using a nonlinear approach. In: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Kyongju, Korea, vol 1, pp 531–536

Mezouar Y, Prats M, Martinet P (2007) External hybrid vision/force control. In: Intl. Conference on Advanced Robotics (ICAR'07), Jeju, Korea

Morel G, Bidaud P (1996) A reactive external force loop approach to control manipulators in the presence of environmental disturbances. In: IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, USA, vol 2, pp 1229–1234, DOI 10.1109/ROBOT.1996.506875

Morel G, Malis E, Boudet S (1998) Impedance based combination of visual and force control. In: IEEE International Conference on Robotics and Automation (ICRA'98), Leuven, Belgium, vol 2, pp 1743–1748

Nelson B, PKKhosla (1996) Force and vision resolvability for assimilating disparate sensory feedback. IEEE Trans on Robotics and Automation 12(5):714–731

Nelson B, Morrow J, Khosla P (1995) Improved force control through visual servoing. In: American Control Conference, 1995. Proceedings of the, vol 1, pp 380–386vol.1

Petrovskaya A, Ng A (2007) Probabilistic mobile manipulation in dynamic environments with application to opening doors. In: Int. Joint Conf. on Artificial Intelligence, Hyderabad, India

Petrovskaya A, Khatib O, Thrun S, Ng A (2006) Bayesian estimation for autonomous object manipulation based on tactile sensors. In: Proc. IEEE International Conference on Robotics and Automation ICRA 2006, pp 707–714, DOI 10.1109/ROBOT.2006.1641793

Prats M, Martinet P, del Pobil AP, Lee S (2007a) Vision/force control in task-oriented grasping and manipulation. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, USA, pp 1320–1325

Prats M, del Pobil AP, Sanz PJ (2007b) Task-oriented grasping using hand preshapes and task frames. In: Proc. of IEEE International Conference on Robotics and Automation, Rome, Italy, pp 1794–1799

Prats M, Martinet P, del Pobil AP, Lee S (2008) Robotic execution of everyday tasks by means of external vision/force control. Intelligent Service Robotics 1(3):253–266

Prats M, Sanz PJ, del Pobil A (2010) A framework for compliant physical interaction - the grasp meets the task. Autonomous Robots 28(1):89–111

Salisbury J (1980) Active stiffness control of a manipulator in cartesian coordinates. In: IEEE International Conference on Decision and Control, Albuquerque, USA, pp 95–100

Schmid AJ, Gorges N, Göger D, Wörn H (2008) Opening a door with a humanoid robot using multi-sensory tactile feedback. In: IEEE International Conference on Robotics and Automation, Pasadena, CA, pp 285–291

Se S, Lowe D, Little J (2001) Vision-based mobile robot localization and mapping using scale-invariant features. In: IEEE International Conference on Robotics and Automation, Seoul, Korea, pp 2051–2058

Son JS, Howe R, Wang J, Hager G (1996) Preliminary results on grasping with vision and touch. In: In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS96, Osaka, Japan, vol 3, pp 1068–1075

Stemmer A, Schreiber G, Arbter K, Albu-Schäffer A (2006) Robust assembly of complex shaped planar parts using vision and force. In: IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, Heidelberg, Germany, pp 493–500

Tegin J, Wikander J (2005) Tactile sensing in intelligent robotic manipulation - a review. Industrial Robot: An International Journal 32(1):64–70

Wyrobek K, Berger E, Van der Loos H, Salisbury J (2008) Towards a personal robotics development platform: Rationale and design of an intrinsically safe personal robot. In: IEEE International Conference on Robotics and Automation, pp 2165–2170, DOI 10.1109/ROBOT.2008.4543527