Towards semi-autonomy in laparoscopic surgery through vision and force feedback control

Alexandre Krupa*, Christophe Doignon*, Jacques Gangloff*
Michel de Mathelin*, Luc Soler† and Guillaume Morel‡

*University of Strasbourg I, †IRCAD, ‡EDF R&D, France

Abstract: This paper shows ongoing research results on the development of automatic control modes for robotized laparoscopic surgery. We show how both force feedback and visual feedback can be used in an hybrid control scheme to autonomously perform basic surgical subtasks. Preliminary experimental results on an example clamping tasks are given.

Introduction

In laparoscopic surgery, small incision points are made in the human abdomen contrary to open surgery where a large incision is made. The surgeon puts a trocar at these incision points. Then, surgical instruments and an endoscopic camera are inserted through the trocars. Looking at the video signal, the surgeon can move the tools in order to perform the desired surgical task. The advantages of laparoscopic surgery are obvious: reduced pain and hospital stay, as well as a quicker recovery. The main inconvenience of this surgical technique is due to the stand of the surgeon which is very tiring and limits the duration of the surgical procedure as well as the surgeon's performance.

Figure 1. Example task : cleaning-suction process in laparoscopic surgery

Laparoscopic surgical robots have appeared recently. Several commercial systems for laparoscopic surgery exist today, e.g., ZEUS (Computer Motion, Inc.) or EndoWrist (Intuitive Surgical, Inc.). With these systems, robot arms are used to manipulate the instruments and the camera. The surgeon teleoperates the robots through master arms using the visual feedback from the laparoscope. This reduces the surgeon tiredness, and potentially increases his/her precision by the use of a high master/slave motion ratio.
Our research in this field is aimed at expanding the potentialities of such systems by providing "automatic modes" in which the system autonomously performs simple subtasks. In this case, the robot controller uses the visual feedback from the laparoscope to automatically drive the instruments, through a visual

servo loop, towards their desired location. Pioneer research in this field can be found in [1] or [2] where the laparoscope is programmed to automatically track a surgical instrument in its field of view, or in [3] were a 3 dof surgical robot is automatically placed from vision feedback.
In cooperation with IRCAD (Institut de Recherche sur le Cancer de l'Appareil Digestif, Strasbourg, France), we particularly focus on liver surgery. This surgery involves a number of repetitive gesture, such as the cleaning-suction process (Fig. 1) : first, the surgeon has to sweep a liquid projecting instrument over the surface to be cleaned up ; then, the same instrument (with the pump reversed) is used to suck up the remaining liquid. It shall be noticed that although this cleaning process is rather simple as compared to critical surgical gestures, it involves repetitive movements for the surgeon who drives the master devices. Providing semi-autonomy to the robot by the use of vision based control will then relieve the surgeon of tiredness induced by this kind of simple and repetitive tasks and allow a maximized concentration for the delicate phases of the operation. In this case, the surgeon delimits the surface to be cleaned up on the screen and the robot autonomously performs the cleaning-suction process.
The paper first shows how force feedback control can be used in order to limit the forces exerted on the patient through the trocar. Then, an application of combined vision and force control to automatic clamping is given, with preliminary experimental results.

1. The use of force control in laparoscopic manipulation

A particularity of laparoscopic manipulation lies in the presence of a trocar, which limits to 4 dof the surgical instruments motion.
This problem has been addressed over the past by the mean of mechanical design [4], [5], [6]. The proposed solutions use either a remote rotation center device, which requires a precise positioning of the robot in the trocar prior to the operation, or a 6 DOF robot having two passive joints in the wrist. This last solution suffers from a lack of precision due to backlash that may occur between the instrument and the trocar. Our proposition is to use a fully actuated 6 DOF robot, that provides both functional flexibility and precision. In order to avoid large forces to be exerted on the patient through the trocar, the robot tip is equipped with a force sensor.

1.1. Kinematics

The kinematics of the laparoscopic manipulation is depicted in figure 2, where F_c is a frame attached to the tip of the tool handler, such that the z_c axis is colinear to the tool penetration axis ; F_s is the F/T sensor frame, with z_s colinear to z_c ; P is the point of the tool handler that instantaneously coincides with the trocar ; l is the distance between the origins of F_s and F_c and d is the distance from P to the origin of F_c.
As only 4 dof are available, we choose to use the following parametrization for the operational space velocity vector:

$$\dot{W} = \begin{pmatrix} \dot{d} & \omega_x & \omega_y & \omega_z \end{pmatrix}^T \tag{1}$$

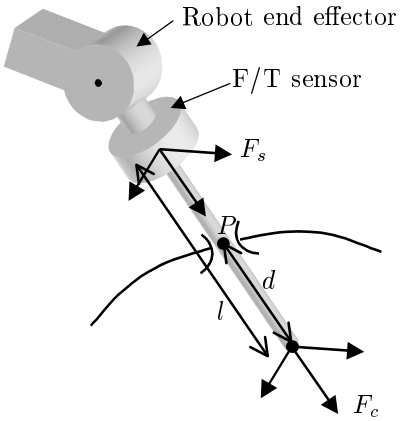

Figure 2. Manipulation through a trocar

where $\dot{d}$ is the translational velocity of the tool handler along z_c, and ω_x, ω_y and ω_z are projections of the absolute rotational velocity over x_c, y_c and z_c respectively.

Let $\dot{r}_p$ be a vector grouping the components, expressed in F_c, of the twist describing the absolute velocity of the tool handler at point P. Since the trocar does not allow any tangential motion of the tool handler at point P, we have:

$$\dot{r}_p = \begin{pmatrix} v(P)_{c/0} \\ \omega_{c/0} \end{pmatrix} = \begin{pmatrix} 0_{2\times 4} \\ I_{4\times 4} \end{pmatrix} \dot{W} = A\dot{W} \tag{2}$$

Then, we can express the same twist at the origin O_c of F_c:

$$\dot{r}_c = \begin{pmatrix} v(O_c)_{c/0} \\ \omega_{c/0} \end{pmatrix} = M(d)\dot{r}_p \quad \text{with:} \quad M(d) = \begin{pmatrix} 1 & 0 & 0 & 0 & d & 0 \\ 0 & 1 & 0 & -d & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{3}$$

where both $\dot{r}_p$ and $\dot{r}_c$ are expressed in F_c. Furthermore, as F_c is rigidly attached to the robot end-effector, standard kinematics can be used to provide the robot *natural* jacobian matrix J_N, such that $\dot{r}_c = J_N\dot{q}$, where $\dot{q}$ is the robot joint velocity. Finally, the inverse kinematic model is :

$$\dot{q} = J_N^{-1}(q)M(d)A\dot{W} \tag{4}$$

where the robot kinematics is supposed to be nonsingular.

1.2. Force feedback

If the penetration depth d was perfectly known, equation (4) could be used to drive the robot joint velocities while respecting the trocar constraint. However, in practice, due to the experimental conditions of a surgical operation, this assumption is not realistic, $M(d)$ is not perfectly known. As a consequence, using

equation (4) to control the robot motion will induce a lateral motion of P and forces could be applied on the patient through the trocar. To cope with this problem, a force feedback controller is added to the system in order to limit the lateral forces applied at the incision point.
Let f_x and f_y be the measured forces along x_s and y_s, respectively. Hybrid position/force control can be used to servo these two measures to zero. Assuming that the robot joints are velocity controlled, we get a conventional motion rate control :

$$\dot{q}^* = J_N^{-1}(q)M(\hat{d})\dot{r}_p^* = J_N^{-1}(q)M(\hat{d}) \begin{pmatrix} k\ f_x \\ k\ f_y \\ \dot{W}^* \end{pmatrix} \tag{5}$$

where $\dot{q}^*$ is the joint velocity control input, $\hat{d}$ is the estimation of the penetration depth d, k is a gain corresponding to the programmed apparent damping, and $\dot{W}^*$ is the operational space velocity control input, that can be provided either by the surgeon through master devices (teleoperation mode) or by a vision based controller (automatic mode). This control strategy is known to be very robust providing that the force loop bandwidth is low enough as compared to the joint velocity loop bandwidth.
Actually, the first experiments with small depth estimation errors exhibited good results in lateral force limitations. However, for larger estimation errors, the force loop was not fast enough to efficiently compensate for lateral motions at the trocar, and large forces occurred (see experimental results).
In order to increase the overall controller performance, it was then necessary to provide online estimation of the penetration depth d. Two drastically different strategies were experimented:

- In the first one, we use the measured forces and torques in the xy plane to estimate the distance (that is the force/torque ratio). The algorithm, briefly depicted in appendix 1, uses a robust least square identification involving a forgetting factor, a sliding window and a threshold.
- In the second one (see appendix 2), an adaptive approach is proposed to estimate $\hat{d}$. It uses as inputs both the rotational velocities (ω_x^*, ω_y^*) sent to the robot, and the measured resulting tangential forces (f_x, f_y).

1.3. Experimental results

In order to evaluate the efficiency of the force feedback controller and the depth estimation strategies, an experimental testbed was built, consisting of an industrial 6 dof robot equipped with a force sensor, manipulating a rigid 40 cm bar, trough a trocar. During these experiments, we apply square reference signals ω_x^* and ω_y^* with a magnitude of ± 1.5 deg/s. In the first set of experiments, the penetration depth $d = 0.1m$ and its initial estimate is $d_0 = 0.3m$. The corresponding results are shown in the first plot column of Fig. 3. In the second set of experiments, the penetration depth $d = 0.2m$ and its initial estimate is $d_0 = 0.02m$. The corresponding results are shown in the second plot column of Fig. 3. Three strategies are compared. The first one consider a constant estimation of d, that is $\hat{d} = d_0$. The results (first line of Fig. 3) show

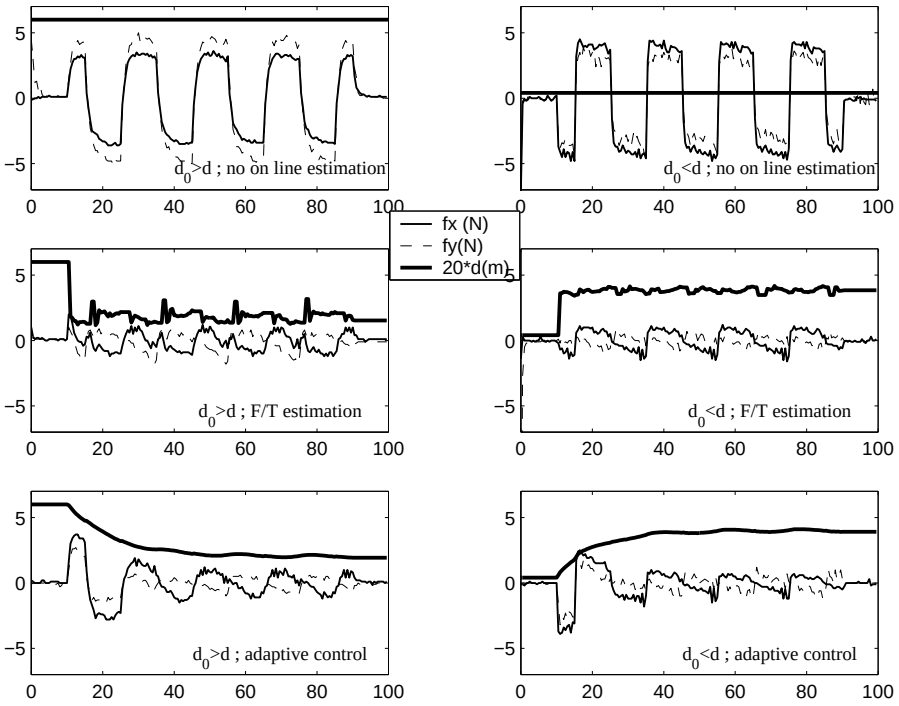

Figure 3. Force feedback experimental results *(x axis : time in seconds)*

significant forces (about ± 5 N) in both xy directions. For faster motions, these forces would clearly become unacceptable for the patient. The second strategy uses the direct estimation of d with forces and torques measurement. One can see (second line of Fig. 3) that the forces are limited to less than $\pm 2N$ and that the estimation of d converges rapidly towards the actual value. A small fluctuation of $\hat{d}$ remains, that does not significantly affect the force controller performance. Finally, the third strategy (last line, Fig. 3) uses the adaptive algorithm. This method exhibits a rather slower convergence of the depth estimation as compared to direct estimation of d. However, the estimated signal $\widehat{d}$ is smoother. This is due to the dynamics of the gradient law that acts on $\dot{\widehat{d}}$, whereas the least-squares algorithm gives instantaneously the estimation of $\widehat{d}$. Of course, once the estimation convergence is obtained, the force control performance is similar for the two on line estimation approaches.

2. Vision based control of the instrument movements

In "conventional" laparoscopic telemanipulation, the four degrees of freedom of the instrument are directly controlled by the surgeon through master devices, using the video feedback of the laparoscope. Rather, we want to provide autonomy to the system, that is to use the video feedback in a control algorithm in order to achieve surgical tasks.

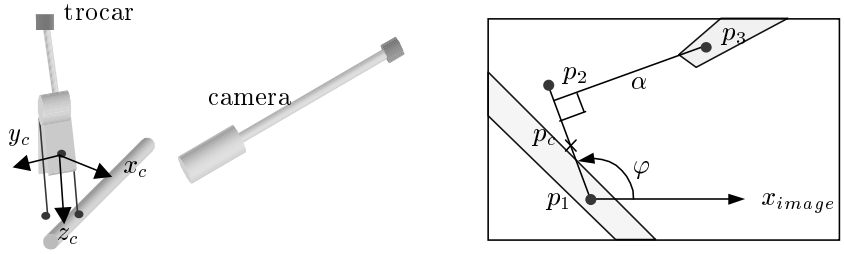

Figure 4. Vessel clamping task

2.1. The use of optical markers

Of course, a major diffulty lies in the poor structuration of the observed scene. To cope with this problem, structured lightening can be added by the mean of laser pointers attached to the tool and/or the camera. A first attempt was shown in [3] where a laser spot was used for depth estimation in a surgical robot. In this work, only one degree of freedom of the surgical instrument was driven with this optical marker.

We propose here to generalize this approach. We consider a clamping task, depicted in figure 4. In this experiment, we use two laser pointers mounted on a surgical tool. The laser beams are parallel, colinear to z_c, in the (z_c, y_c) plane. We also add an optical marker on the tip of the tool. This marker lies in the x_c axis. Note that the camera is fixed in this experiment, although in the future, we intend to mount it on a second robot, in the ZEUS system. An interesting property of this setup is that robust extraction of the optical marker images can be obtained by comparing an image with and without the markers on.

From the image coordinates of the three spots (p_1, p_2, p_3), we built the following image feature vector (see figure 4):

$$s = (u_c,\ v_c,\ \alpha,\ \varphi)^T \tag{6}$$

where (u_c, v_c) are the coordinates of $pc = \frac{1}{2}(p_1 + p_2)$, φ is angle from the x image axis to the vector joining p_1 to p_2, and α is the distance from p_3 to the line p_1p_2. The desired value s^* for s is supposed to be known. In the final scenario, the surgeon will indicate on a tactile screen the location of both the vessel to be clamped and the point where the clamping should be done. This will automatically set u_c^*, v_c^* and φ^*. Also, the clamping position is characterized by a known value of α^*.

2.2. Visual servoing

Due to the complexity of the scene, there is no way to precisely model the exact image jacobian matrix, that maps the instrument velocity into $\dot{s}$. Rather, we built a simplified jacobian matrix J_i for a nominal configuration, for which the lightened scene is planar and z_c is perpendicular to this plane. The following properties can be demonstrated: *i*) the velocity $\dot{d}$ only affects $\dot{\alpha}$; *ii*) the velocity ω_z does not affect $\dot{u}_c$ and $\dot{v}_c$; *iii*) the velocities ω_x and ω_y do not affect $\dot{\varphi}$; Thus

the image jacobian matrix J_i is given by:

$$\dot{s} = J_i \dot{W} \text{ with, } J_i = \begin{pmatrix} 0 & J_{i_{12}} & J_{i_{13}} & 0 \\ 0 & J_{i_{22}} & J_{i_{23}} & 0 \\ J_{i_{31}} & J_{i_{32}} & J_{i_{33}} & J_{i_{34}} \\ 0 & 0 & 0 & J_{i_{44}} \end{pmatrix} \tag{7}$$

It is intersting to notice that we have a quasi triangular system, apart from the bloc mapping (ω_x, ω_y) into $(\dot{u}_c, \dot{v}_c)$.
As a number of unknown geometrical and optical parameters is involved in the computation of J_i, a first identification stage can be run at the beginning of any experiment. Constant velocities ω_x, ω_y, ω_z and $\dot{d}$ are applied independently during a short time T. The variations of s are computed and the jacobian matrix components are estimated by:

$$\left\{ \begin{array}{ccccc} \widehat{J}_{i_{12}} = \frac{\Delta u_c}{\omega_x T} & \widehat{J}_{i_{13}} = \frac{\Delta u_c}{\omega_y T} & \widehat{J}_{i_{22}} = \frac{\Delta v_c}{\omega_x T} & \widehat{J}_{i_{23}} = \frac{\Delta v_c}{\omega_y T} & \\ \widehat{J}_{i_{31}} = \frac{\Delta \alpha}{dT} & \widehat{J}_{i_{32}} = \frac{\Delta \alpha}{\omega_x T} & \widehat{J}_{i_{33}} = \frac{\Delta \alpha}{\omega_y T} & \widehat{J}_{i_{34}} = \frac{\Delta \alpha}{\omega_z T} & \widehat{J}_{i_{44}} = \frac{\Delta \varphi}{\omega_z T} \end{array} \right. \tag{8}$$

The visual servoing then consists of the quasi decoupling control law:

$$\dot{W}^* = \lambda \widehat{J}_i^{-1} (s^* - s) \tag{9}$$

Of course, the remaining degrees of freedom of the robot are still controlled using force feedback, and the overall controller consists of an hybrid vision/force controller that combines equation (5) and (9), together with an on-line estimation of $\hat{d}$.
The well known stability condition is that $J_i \widehat{J}_i^{-1}$ remains positive definite. Due to the complexity of the scene, the stability properties cannot be formally derived. Notice that a number of techniques have been proposed in the past using constant image jacobian matrices, exhibiting good experimental stability robustness properties (see e.g. [10]).

2.3. Planning

Vessel clamping, is decomposed in three stages (Fig. 5):

1. the angle φ is servoed to its desired value φ^*. The other components are not servoed. Then, the resulting motion is a pure rotation of the instrument around z_c.

2. the image coordinates u_c, v_c are servoed towards u_c^*, v_c^*, while φ is maintained at φ^*. This phase involves mainly ω_x and ω_y motions.

3. the distance α is servoed to α^*, while the other components of s are kept to their desired value. This phase mainly consist of a final z_c translation.

With such a decomposition, the image jacobian matrix components are not all identified at the beginning of the experiment. Rather, they are estimated progressively during the clamping, when they are required. $\widehat{J}_{i_{44}}$ is identified before the phase 1, $\widehat{J}_{i_{12}}$, $\widehat{J}_{i_{13}}$, $\widehat{J}_{i_{22}}$ and $\widehat{J}_{i_{23}}$, before the phase 2 and $\widehat{J}_{i_{3k}}$, $k = 1..4$, before phase 3. Note that in practice, only $J_{i_{31}}$ is actually identified, since, in the final approach configuration, $J_{i_{32}} \approx 0$, $J_{i_{33}} \approx 0$ and $J_{i_{34}} \approx 0$.

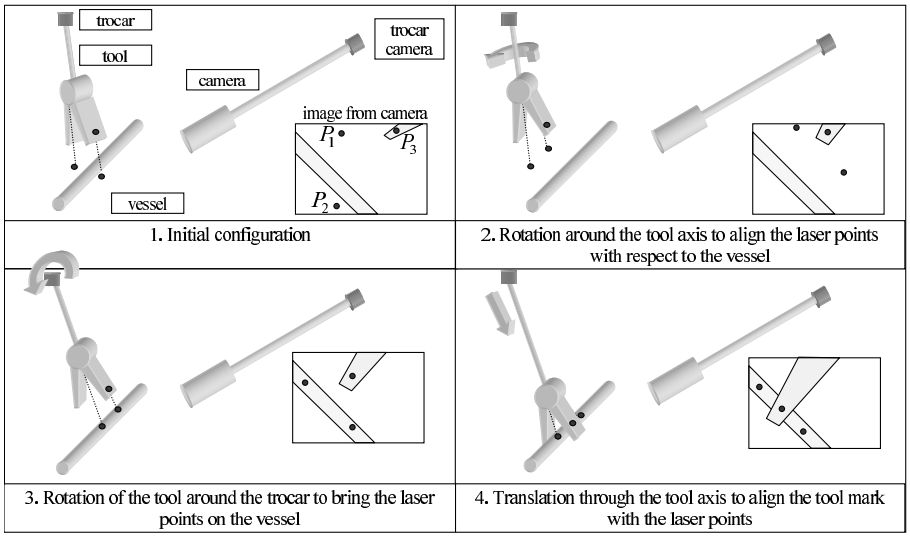

Figure 5. Planning a clamping task

2.4. Experimental results

This strategy was experimented on a lab testbed. A surgical instrument was equipped with to laser beams and an LED. In these preliminary experiments, the desired values of s are set manually to a value determined during a learning phase. $u^* = 350$ pixels, $v^* = 300$ pixels, $\varphi^* = -20$ deg. and $\alpha^* = 30$ pixels. A number of experiments with different initial configurations has been conducted. The experimental results are depicted in Figure (7), where the different phases can be observed.

During the first 3 seconds, an open loop motion aimed at identifying $\frac{\dot{\varphi}}{\omega_z}$ is performed. Then the first phase is run, providing the convergence of φ. Note that since large initial errors are involved, the controller output ω_z^* is saturated, which explains the linear convergence. Also, during this first phase, the three other components of s are not controlled, and may exhibit divergence.

At $t \approx 15$s, after the convergence of φ, a second identification phase is run to estimate the mapping between (ω_x, ω_y) and $(\dot{u}_c, \dot{v}_c)$. This produces a variation of the different components of s. The exponential convergence of u_c, v_c is then observed (from $t \approx 20$s to $t \approx 40$s, while φ is maintained to its desired final value.

The final phase consists then of the exponential decrease of α, that is the final clamping z_c translational motion.

Conclusion

Two main results are provided in this paper. The first one concerns the use of force feedback control to limit the forces exerted on a trocar in laparoscopic manipulation. This increases the manipulation performance over existing solutions, that use 4 dof mechanical devices, exhibiting either backlash (under-

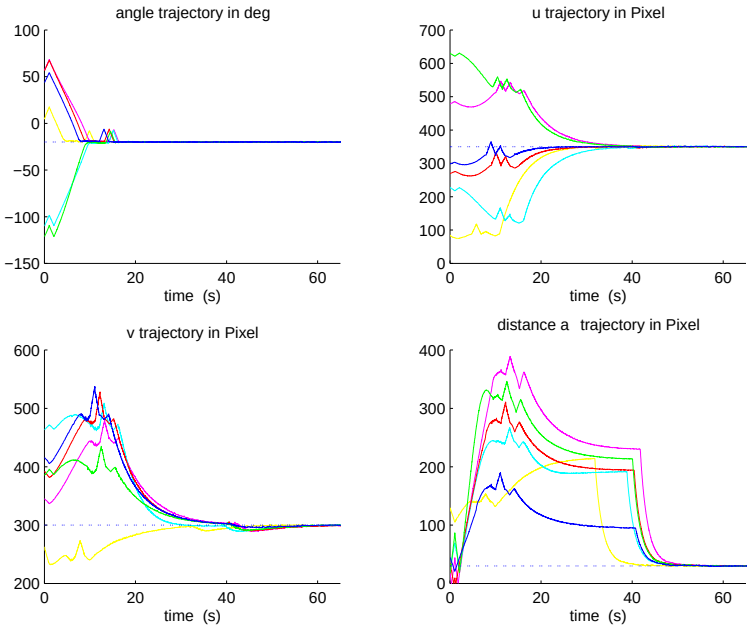

Figure 6. Visual servoing results.

actuated 6 dof robots) or a lack of flexibility (remote rotation center 4 dof robots). The second result concern the automatic realization of a clamping task using visual servoing. A simple strategy was implemented, exhibiting a robust behavior during preliminary lab experiments. Live experiments on pig liver surgery are to be programmed to validate this kind of approach.

References

[1] G.-Q. Wei, K. Arbter and G. Hirzinger. Real-Time Visual Servoing for Laparoscopic Surgery. *IEEE Engineering in Medicine and Biology*, 16(1), pp. 40-45, 1997.

[2] A. Casals, J. Amat, D. Prats and E. Laporte. Vision Guided Robotic System for Laparoscopic Surgery. *Proc. of the IFAC International Congress on Advanced Robotics*. Barcelona, Spain, 1995.

[3] M. Mitsuishi, S. Tomasaki, T. Yoshidome, H. Hashizume and K. Fujiyara Telemicro-surgery system with intelligent user interface. *Proc. of the 2000 IEEE International Conference on Robotics and Automation*. pp 1607-1614, San Francisco, CA, 2000.

[4] A. Faraz and S. Payandeh. A Robotic Case Study: Optimal Design for Laparoscopic Positioning Stands. *Proc. of the 1997 IEEE International Conference on Robotics and Automation* ,pp. 1553-1560. Albuquerque, New Mexico, April 1997.

[5] A. Madhani, G. Niemeyer and J.K. Salisbury. The Black Falcon: A Teleoperated Surgical Instrument for Minimally Invasive Surgery. *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, Victoria B.C., Canada, October, 1998.

[6] V.F. Munoz, C. Vara-Thorbeck, J.C. DeGabriel, J.F. Lozano, E. Sanchez-

Badajoz, A. Garcia-Cerezo, R. Toscano and A. Jimenez-Garrido. A Medical Robotic Assistant for Minimally Invasive Surgery. *Proc. of the* 25^{th} *IEEE International Conference on Robotics and Automation.* San Francisco, CA, pp. 2901-2906, April 2000.

[7] M. de Mathelin and R. Lozano. Robust adaptive identification of slowly time-varying parameters with bounded disturbances. *Automatica*, vol. 35, pp. 1291-1305, July 1999.

[8] F. Chaumette, P. Rives, B. Espiau. Classification and realization of the different vision-based tasks. *Visual servoing*, Koichi Hashimoto, pp. 199-228. World Scientific Press, 1993.

[9] F. Chaumette, *La relation vision-commande: théorie et application à des tâches robotiques.* Thèse, Université de Rennes, France, juillet 1990.

[10] B. Espiau, F. Chaumette, P. Rives, *A New Approach to Visual Servoing in Robotics.* IEEE Trans. on Robotics and Automation, vol 8 no 3, june 1992

[11] A. Krupa, M. de Mathelin, G. Morel. The use of force control in laparoscopic surgery *Technical Report*, Accessible on http://www.gravir.u-strasbg.fr/, 2000

Appendix 1

The distance, d, can be identified from the measurements of the forces f_x, f_y and the torques T_x, T_y. If we define m as the distance between the force/torque sensor's center and the incision point, i.e., $m = l - d$, then:

$$m = \frac{\sqrt{T_x^2 + T_y^2}}{\sqrt{f_x^2 + f_y^2}} = \frac{T_r}{f_r} \tag{10}$$

In order to provide a robust identification the following cost function was used with sliding window and forgetting factor:

$$J(t, t_0) = \int_{max(t-T, t_0)}^{t} e^{-\lambda(t-\tau)} (T_r(\tau) - f_r(\tau)\widehat{m}(t))^2 d\tau \tag{11}$$

where $\lambda > 0$ is a forgetting factor and $T > 0$ is the size of the sliding window. The least-squares estimate $\widehat{m}(t)$ that minimizes $J(t, t_0)$ is equal to:

$$\widehat{m}(t) = R(t, t_0)^{-1} Q(t, t_0) \tag{12}$$

$$\text{with} : \begin{cases} R(t, t_0) = \int_{max(t-T, t_0)}^{t} e^{-\lambda(t-\tau)} f_r^2(\tau) d\tau \\ Q(t, t_0) = \int_{max(t-T, t_0)}^{t} e^{-\lambda(t-\tau)} f_r(\tau) T_r(\tau) d\tau \end{cases} \tag{13}$$

However, equation (12) cannot be used directly to estimate m, particularly when the force signals are close to zero, bringing a high noise to signal ratio. Therefore, dead-zone is added (cf. [7]). If $f_r(t)$ or $T_r(t)$ decrease below some threshold value f_{th} or T_{th}, the computation of the least-squares estimate is frozen, i.e., only the reference tool velocity $\dot{d}^*$ along z_c is taken into account. Consequently, the robust estimation algorithm is defined as follows:

$$\widehat{d}(t) = \begin{cases} l - R(t, t_k)^{-1} Q(t, t_k) & \text{if } f_r(t) > f_{th} \\ & \text{and } T_r(t) > T_{th} \\ & \text{and } t \geq t_k + T_0 \\ \widehat{d}(T_k) + \int_{T_k}^{t} \dot{d}^*(\tau) d\tau & \text{otherwise} \end{cases} \tag{14}$$

where t_k is the last time instant when f_r and T_r left the dead-zone area, and T_k is the last time instant when f_r or T_r entered the dead-zone area. Details on this algorithm can be found in [11].

Appendix 2

The alternative adaptive approach used to estimate $\hat{d}$ is based on a model of the interaction between the robot and the patient at the incision point, that is:

$$f_x = -g.x_p \quad f_y = -g.y_p \tag{15}$$

where x_p and y_p are the lateral displacement of point P with respect to its equilibrium position and g is the stiffness of the abdominal wall of the patient. Neglecting the robot joint dynamics (which, again, is supposed to be very fast as compared to the force loop dynamics), the closed loop behavior is :

$$\left\{ \begin{array}{rcl} -kf_x & = & (\hat{d}-d)\omega_y^* + \frac{\dot{f}_x}{g} \\ -kf_y & = & -(\hat{d}-d)\omega_x^* + \frac{\dot{f}_y}{g} \\ \dot{d} & = & \dot{d}^* \end{array} \right. \tag{16}$$

These equations are linear with respect to the parametrization error $(d - \widehat{d})$. If a good estimate of g is known, the following normalized gradient algorithm to estimate d can be used:

$$\dot{\hat{d}} = \dot{d}^* + k_1(\dot{f}_x + g\ kf_x)\frac{\omega_y^*}{\epsilon + \omega_x^{*2} + \omega_y^{*2}} - k_1(\dot{f}_y + g\ kf_y)\frac{\omega_x^*}{\epsilon + \omega_x^{*2} + \omega_y^{*2}} \tag{17}$$

where $k_1 > 0$ is the gain of this gradient algorithm and $\epsilon > 0$ is a normalization coefficient. The stability and convergence properties of this estimation algorithm are given in [11]: the algorithm is stable and the convergence of the parameter error to zero is obtained if there is enough excitation (i.e., enough rotational velocities around axis x_c and y_c).
Furthermore, If g is not known, we can rewrite equation (16) as :

$$\left\{ \begin{array}{rcl} \dot{f}_x & = & -g\ kf_x - g(\hat{d}-d)\omega_y^* \\ \dot{f}_y & = & -g\ kf_y + g(\hat{d}-d)\omega_x^* \end{array} \right. \tag{18}$$

which is linear with respect to the stiffness coefficient g. Thus, the following gradient algorithm can be used to estimate both parameters g and d:

$$\left\{ \begin{array}{rcl} \dot{\hat{d}} & = & \dot{d}^* + k_1(\dot{f}_x + \widehat{g}\ kf_x)\omega_y^* - k_1(\dot{f}_y + \widehat{g}\ kf_y)\omega_x^* \\ \dot{\hat{g}} & = & -k_1(\dot{f}_x + \widehat{g}\ kf_x)kf_x - k_1(\dot{f}_y + \widehat{g}\ kf_y)kf_y \end{array} \right. \tag{19}$$

where $k_1 > 0$ is the gain of this gradient algorithm. The stability and convergence properties of this estimation algorithm are given in [11]: the parameter convergence to zero is obtained, assuming enough excitation.