A MULTI-BEHAVIOUR ALGORITHM FOR AUTO-GUIDED MOVEMENTS IN SURGEON ASSISTANCE

E. Bauzano, V.F. Muñoz and I. Garcia-Morales

University of Malaga, Spain

ABSTRACT

This paper focuses on autonomous movements to aid the surgeon to perform certain tasks. Robotic assistants have solved the drawbacks of Minimally Invasive Surgery (MIS) and provide additional skills to the surgeons. However, some authors argue that these systems could lengthen the operating time. The solution is the automation of certain maneuvers that help the surgeon during a surgical maneuver. This work proposes control architecture for a surgical robot capable of performing autonomous movements. In this way, a trajectory planner based on a behaviour concept computes the required velocity vector of the surgical instrument hold by the robot.

Keywords: Automatic Movements; Minimally Invasive Surgery; Robot Assistant

1 INTRODUCTION

The enormous complexity and costs limit the clinical impact of the robotic assistants for Minimally Invasive Surgery (MIS) despite of its well known advantages. Some authors argue that the use of robots in surgery, although providing more precision, could lengthen operating time [1].

One solution to this problem consists of automating certain surgical maneuvers. *Visual Servoing* is one of the most common techniques to perform automated tasks. Control of surgical instrument movements involves calculating instant linear and angular velocity references in each control period. These references are obtained by analyzing the images from a non-calibrated stereo vision system [2] or acquiring those images through a regular laparoscopic surgery camera [3]. Visual Servoing also enables safe movements of the laparoscopic tool, for example, on cardiac surgery, so the instruments are synced with the heart beat [4]-[5].

Other works are devoted to assist the surgeon with robots during the intervention. This way, some developments have performed autonomous stitching and knot tying procedures [6]; others automatically guide a robotic tip in colonoscopy [7], provide automatic transformations to a robot assistant from laparoscopic navigation to open-surgery motion [8], or give autonomous decisions on *teleoperation* with high communication latency or low bandwidth [9]. There are also more complex systems like EndoPAR which automates a knot tying procedure in heart surgery [10], and allows the surgeon to operate as if there was no heartbeat. Furthermore, there are also studies for human-machine skill transfer on robot assistants so they may perform automatic knot tying procedures [11], or automatic navigation on cholecystectomy without pre-operative information [12].

This paper proposes a solution for an automatic movement of the surgical instrument arm equipped with an active wrist. A passive wrist emulation (PWE) controller is needed in order to know the point of insertion so-called *fulcrum points*, as well as minimize the forces over the abdomen. This arm must also avoid the collision with the surgeon tool during its navigation. The main goal of this development is devoted to replacing a human assistant for the surgeon in certain laparoscopic surgery procedures. In this way, authors propose a control architecture for performing automated tasks which allows adapting the movements to the surgeon actions.

The structure of this article is divided into six sections. After this introduction, section 2 states the control architecture proposed to solve auto-guided movements on MIS robotic assistants, whereas section 3 resumes the control philosophy of the PWE to locate the fulcrum point. With this general solution, section 4 explains the developed methodology for moving the robot to one of the surgeon's

Contact author: Enrique Bauzano Nuñez[1]

[1]ebauzano@isa.uma.es.

tool tip. This technique has been applied to the task of taking gauze to the surgeon, as presented on section 5. Finally, section 6 discusses some possible improvements to the methodology, as well as related future works.

2 CONTROL ARCHITECTURE

This section introduces the architecture proposed to perform automatic movements. Firstly, a brief description of the surgical workspace is given to explain the situation where the robot has to develop the task, as well as all movement constraints that limit its freedom. Once the problem has been presented, the specifications the robot must accomplish are discussed. These requirements should let the robotic assistant navigate into the abdominal cavity in a safely way.

2.1 PROBLEM STATEMENT

The environment where the robot interacts with both the patient and the surgeon consists of a closed space, the abdominal cavity, as it is shown in Figure 1. The camera *C* and the instrument of the robot *R* are inserted through their respective fulcrum points G_C and G_R. Moreover, this environment also includes the surgeon's tools used for surgery procedures. One of them is considered the target for the robot *S*, whereas the other one is the obstacle *O*.

The abdominal cavity where the robot instrument may move is defined by a cone-shaped view field, which contains the scene seen in the screen by the surgeon. The robot can just analyze this space, so both, its instrument *R* and the surgeon's tool *S* must remain inside of this cone. This situation may be fulfilled by following the surgeon's tool *S* with the camera and forcing the robot tool *R* to be within the view field. As an additional hypothesis, the surgeon is able to displace his or her tools *O* and *S* during any movement of the robot. This work supposes that the visual servoing problem is solved.

The fulcrum point G_R is a movement constraint for the robot additional instrument *R*. Therefore, movements inside the patient are limited to four degrees of freedom. In this way, in order to control the position of the surgical instrument, it is used a motion controller to take into account these holonomic constraints [13].

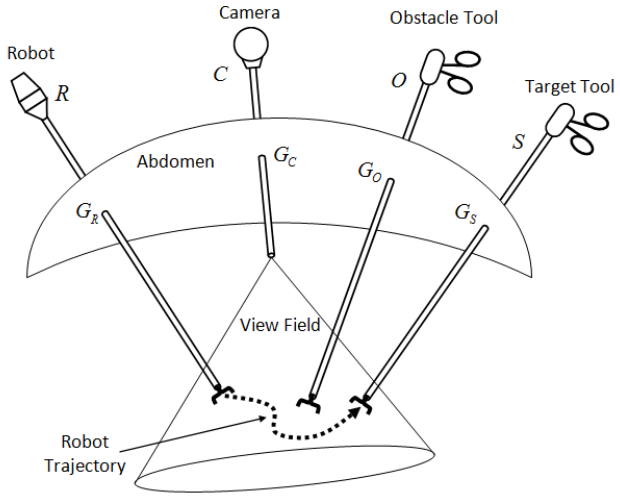

Figure 1 The camera is focused over the surgical workspace, while robot tool goes where the surgeon's target tool is located.

This paper suggests moving the robot instrument *R* to the surgeon's target tool *S* location. The surgeon tool *O* is defined as the unique obstacle to be avoided during an automatic task. The trajectory has to be calculated on-line, because the surgeon's tools are not static since he or she continues the intervention normally. All these considerations will be taken into account on the control architecture proposed on next subsection.

2.2 ARCHITECTURE SCHEME

Once the functionality needed for the robot has been described, an architecture scheme which resumes the capabilities of the system can be introduced as shown in Figure 2. The main element is the *Local Planner*, whose mission is to guide the robot instrument to the surgeon tool avoiding the obstacles.

For this purpose, some information is needed from the environment. In addition to the own kinematics of the robot given by the *Robot Location* feedback, the *Surgeon Location* must be known in order not only for the current tool position, but also for estimating its trajectory thanks to the velocity and acceleration parameters. This prediction is done by the *Surgeon Trajectory Estimator* through the obstacle tool velocity $\vec{v}_O$, and allows the robot to update its trajectory $\vec{v}_R$ in order to avoid possible collisions with the surgeon's obstacle tool *O*.

With all this data, the *Local Planner* may command the special instrument through the *Passive Wrist Emulation* Control. This element is necessary to know the fulcrum point, and generates the Planned Trajectories for the *Robot Arm* spherical navigation. As the robot has an active wrist, the Passive Wrist Emulation (PWE) allows minimizing the forces over the abdomen of the patient [13]. This controller is explained on next section.

3 PASSIVE WRIST EMULATION ON ACTIVE WRIST

Direct actuated wrists avoid the backlash introduced by the trocar-instrument interaction which appears on passive wrists solutions. Thus, the endoscope always reaches the desired spherical position in spite of inaccurate estimation of the fulcrum point. However, this uncertainty will eventually force the abdominal wall and could damage the patient [13]. This section will resume how simultaneously estimate the fulcrum point while reducing the force applied over the abdomen.

Figure 3 illustrates an endoscope movement where the altitude angle β changes from a starting null position *A* to a final position *B*. The robot movement planner uses an initial estimation of the fulcrum position *C* for computing the arc shape trajectory, instead of the unknown real one I_0. In such way, the abdominal force $\vec{F}_a$ gives information about the direction where the fulcrum point *I* is displaced. Therefore,

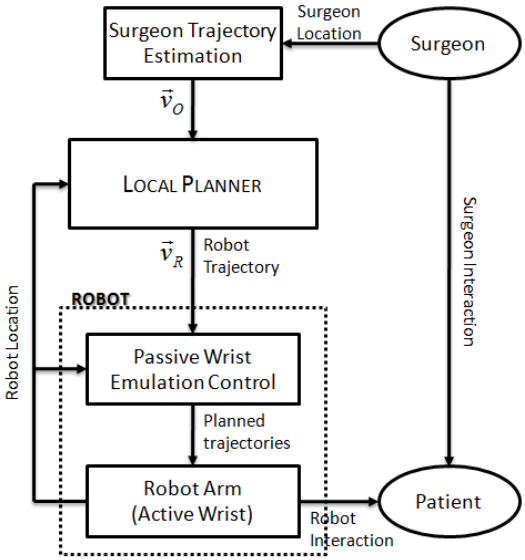

Figure 2 Architecture scheme for performing automatic tasks of a robot assistant.

this force is used for two purposes: i) if navigation is being performed in an accurate way; ii) any unexpected change implies a fulcrum location variation (i.e. respiratory motion or abdominal cavity air pressure variation).

The proposed approach to reduce the efforts applied to the abdominal wall is based on emulating the passive wrist behaviour. On the situation shown in Figure 3, a non-actuated wrist would keep its orientation according to the initial position at the fulcrum point I_0 by means of the angle φ. Thus, the motion controller computes φ thanks to a spring model of the abdomen where both, the force $\vec{F}_a$ and the abdomen stiffness are known [13].

4 AUTO-GUIDED METHODOLOGY

The automated movement proposed in this paper consists of reaching a goal position defined by the surgeon's target tool tip *S*, as it was stated in Figure 1. In this way, the other surgeon's tool *O* is considered an obstacle the robot must avoid. As an additional hypothesis, both surgical tools may be displaced during the robot movement.

Figure 4 shows the proposed *Local Planner* scheme stated in Figure 2 which has been used to compute the required velocity $\vec{v}_R$ for the robot tool (*R* in Figure 1) in order to reach the target tool *S* by avoiding the obstacle *O*. The current velocity of the robot $\vec{v}_R$ and the obstacle tool $\vec{v}_O$ are used on a Fuzzy Logic algorithm for deciding the best strategy to get closer to the target without collisions.

This work proposes the combination of three behaviours in order to plan the required robot velocity:

- The "*Go-to Target*" behaviour finds a trajectory to the target avoiding static obstacles using the Artificial Potential Field methodology (APF), but cannot deal with movable obstacles. However, it can react to changes on the target location.
- The "*Velocity Corrections*" behavior will change the robot velocity $\vec{v}_R$ depending on the obstacle velocity $\vec{v}_O$. These corrections will be more important with high velocities and when the robot is nearby the obstacle tool.
- The "*Backward Movement*" behaviour covers the possibility that the robot tool and the obstacle tool are so close they may collide. This situation is solved by adapting the robot movement to the obstacle in order to avoid the collision, as the surgeon will probably want to displace the robot to free his or her vision space.

As it has been already stated, the decision of which behavior is the best on each situation is taken by a Fuzzy algorithm. Depending on the directions of the robot velocity $\vec{v}_R$ and the obstacle one $\vec{v}_O$, the Fuzzy decision will use the best combination of the behaviors previously stated. The truth table of this Fuzzy decision appears on Table I.

Table I - Truth Table for Behavior Decision

$\vec{v}_O$ \ $\vec{v}_R$	1	2	3	4
1	A	FA	HC	C
2	FA	A	C	HC
3	HC	C	A	FA
4	C	HC	FA	A

A = go Away, FA = Far Away, C = get Closer, HC = High Close.

In this table, velocities have labels assigned. Each label indicates the quadrant of a circle where the velocity $\vec{v}_R$ and $\vec{v}_O$ are orientated in terms of a reference axes. For example, if both velocities are 1 then their directions are the same, whereas if $\vec{v}_R$ is 1 and $\vec{v}_O$ is 3 mean that those directions are

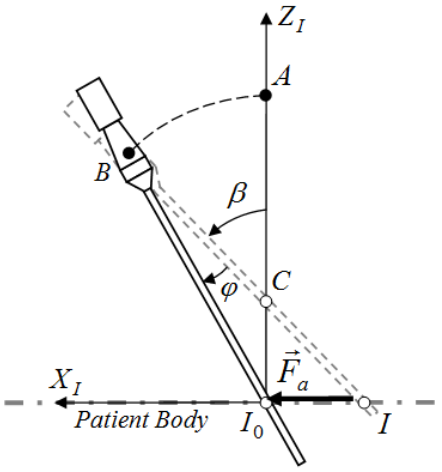

Figure 3 Passive Wrist Emulation behaviour.

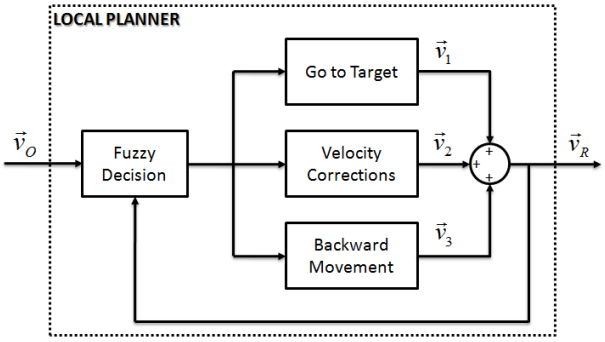

Figure 4 Local Planner scheme proposed to avoid dynamic obstacles.

opposed. The output of Table I is a fuzzy value that is used to select the importance of each behaviour.

4.1 THE "GO-TO TARGET" BEHAVIOUR

This behaviour is devoted to find a path to the target tool avoiding static obstacles. This work has applied the Artificial Potential Field (APF) methodology to fulfill this task. APF associates a repulsive potential field U_{rep} for each obstacle of the environment, as well as an attractive potential field to the target U_{att}. Thus, expression (1) states that the resulting potential field generates a virtual force $\vec{F}_1$ which both, attracts the robot to the goal with force $\vec{F}_{att}$ and repels it from the collision with the obstacles through $\vec{F}_{rep}$ force.

$$\vec{F}_1 = -\nabla U_{att} - \nabla U_{rep} = \vec{F}_{att} + \vec{F}_{rep} \quad (1)$$

The virtual potential functions of this work are based on the expressions of [14]. They have a high value only on surrounding obstacle area, whereas the attractor has a potential field which is proportional to the distance from the robot to the target. The gradient of these expressions gives the force generated by the potential field.

The main problem of APF methodology appears by means of local minima points that may be found on the workspace. This work extends the potential field of [14] by applying the solution proposed by [15] and uses a force field capable of escape from these local minima points.

The APF methodology is commonly applied to path planning of point mobile robots. However, MIS problem requires that the robot R is not just a point, but a line which departs from the fulcrum point G_R to its tool tip, as it has already been shown in Figure 1. One consequence is that the target also becomes a line which departs from the robot fulcrum point G_R and ends on the surgeon target tool tip. Furthermore, as both instruments are long enough to be considered like one-dimensional objects, they may only collide in one point. As shown in Figure 5, the minimal distance ρ between the robot tool and the obstacle tool defines the point over each tool M_R and M_O that would collide in case they approach. Point M_R is called the guide point, because it is used to move the robot to the target.

Secondly, the default potential function generates equipotential surfaces with a cylindrical shape. However, this work has chosen the use of conical surfaces with their vertices on the fulcrum point of the obstacle, as it can be shown in Figure 5. The reason is that the movements are faster near the tool tips, so higher potentials are needed to maintain the distance between robot and obstacle tool.

Therefore, the APF methodology can be just applied to the guide point M_R instead all the robot tool longitude. This point has a target associated to its final position M_R^f (see Figure 5) when the instrument has reached the target. Thus, to calculate the next trajectory location the algorithm steps are:

Procedure for Automatic Movements

1) Locate the minimal distance point M_R

2) Calculate its target M_R^f

3) Apply the forces given by (1) *in order to know the needed velocity of the guide point*

4) Move the robot instrument to fit the new location and the fulcrum point constraint

End Procedure

First step can be geometrically deduced by solving the equation system of two lines that cross themselves, whereas second step just calculates the position of M_R^f by proportional distances (see Figure 5). As for third step, the expressions of attraction force $\vec{F}_{att}$ and repulse force $\vec{F}_{rep}$ are given by the Evolution of Artificial Potential Field (EAPF) stated by [15]:

$$\vec{F}_{att} = 2K_a \vec{r}_{goal}$$

$$\vec{F}_{rep} = \begin{cases} -K_r \left(\frac{1}{\rho} - \frac{1}{\rho_0} \right)^2 \left(\frac{\vec{r}_{goal}}{\rho^2} + \frac{1}{2}\hat{n} \right) & \text{if } \rho < \rho_0 \\ 0 & \text{if } \rho \geq \rho_0 \end{cases} \quad (2)$$

Figure 5 shows both forces presented in (2), being K_a and K_r their respective gains. Repulsive force $\vec{F}_{rep}$ only acts on a bound surrounding the obstacle below distance ρ_0 and has

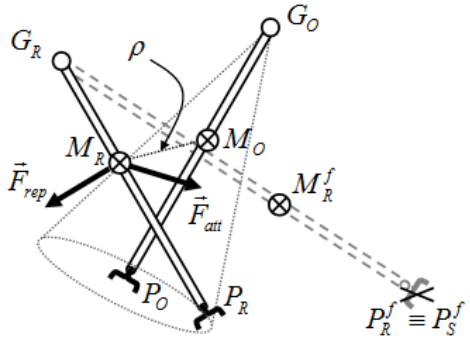

Figure 5 Virtual force for generating trajectories to the target.

two components: one part depends on the distance to the goal $\vec{r}_{goal}$ as well as the distance to the obstacle ρ, and the other is a constant with a direction $\hat{n} = (\hat{r}_{goal} \times \hat{\rho}) \times \hat{\rho}$ perpendicular to $\hat{\rho}$ and always pointing to the target. As a result, this constant forces the robot to move even if it has reached a local minima situation.

To adapt the potential field of the obstacle tool to a cone a new gain K'_r may be defined which depends on the relation between the obstacle tool total length L and the distance from its fulcrum point G_O and the point of minimal distance with the robot M_O (see Figure 5):

$$K'_r = \frac{\overline{G_O M_O}}{\overline{G_O P_O}} K_r = \frac{l}{L} K_r \tag{3}$$

Finally, the expression of $\vec{v}_1$ is deduced in expression (4) from the virtual force $\vec{F}_1$ of (1), where virtual mass m has been considered to be 1:

$$\vec{v}_1 = \int \frac{\vec{F}_1}{m} dt \rightarrow \vec{v}_1(t + \Delta t) = \vec{v}_1(t) + \frac{\vec{F}_1}{m} \Delta t \tag{4}$$

4.2 VELOCITY CORRECTIONS

APF behaviour loses its efficiency in dynamic environments because it gives a trajectory just based on the actual state. This way, a fuzzy algorithm is included to take velocity changes and path predictions into account. For example, if the robot moves on the same direction as the obstacle, there will be little or no variation on the velocity calculated by the APF method. However, if the robot moves nearby the obstacle and it moves in a perpendicular way, then it is possible that both future trajectories cross themselves. The fuzzy system will change the robot velocity and try to avoid the collision by reducing the speed in order to allow the surgeon to follow his or her original trajectory [16]. All possible situations are described by the truth table II.

The outputs on this table are the speed corrections $\vec{v}_2$ previously presented on Figure 4 produced by each combination of antecedent data. These rules adapt the speed of the robot tool so that the obstacle passes before; thus, the cells at the bottom-left of the table, which corresponds to an obstacle that will cross a close point in the trajectory after a long period of time, produce very slow commands. A value of S (Stop) means that $\vec{v}_2 = -\vec{v}_1$, whereas MM (Max Movement) indicates that $\vec{v}_2 = \vec{0}$, which fits with a maximum velocity generated by the APF behaviour already explained in section 4.1.

Table II - Truth Table for Velocity Corrections

Time \ Distance	Z	AZ	M	F	VF
Z	S	SM	NM	MM	MM
AZ	S	SM	NM	MM	MM
M	SM	SM	NM	FM	MM
L	AS	AS	SM	NM	MM
VL	AS	AS	AS	SM	FM

Z = Zero, AZ = Almost Zero, M = Middle, F = Far, VF = Very Far.
L = Long, VL = Very Long.
S = Stopped, AS = Almost Stopped, SM = Slow Movement, NM = Normal Movement, FM = Fast Movement, MM = Max Movement

4.3 BACKWARD MOVEMENT

With the fuzzy system stated above, almost all situations may be covered. However, collision may also occur if the surgeon moves his or her tool rightly to the robot and forces the contact. This could happen because the surgeon needs to apart for some reason the robot tool for a moment. Thus, the natural solution consists of moving the robot instrument rightly into the obstacle velocity direction. This way, the robot would follow the obstacle as long as it is moving, and should stop when the obstacle does. Only when the obstacle frees the default APF trajectory, the robot would restart its normal auto-navigation.

More specifically, the behaviour proposed will be likely a damp between the obstacle and the robot tool. This way, a virtual force will move the robot back at the obstacle trajectory direction which expression is (5):

$$\vec{F}_3 = B\vec{v}_O \tag{5}$$

The gain B on (5) is a constant which indicates how fast the robot velocity will react to the obstacle. If this parameter is high, then the robot tool will change its velocity very fast to reach the obstacle velocity $\vec{v}_O$ fast, and similar conclusions can be taken for low values of B.

Therefore, the resulting velocity of the robot tool can be obtained by adding the three behaviours exposed (6):

$$\vec{v}_R = c_1\vec{v}_1 + c_2\vec{v}_2 + c_3\vec{v}_3 \tag{6}$$

The parameters c_1, c_2 and c_3 on expression (6) are the importance of each velocity $\vec{v}_1$, $\vec{v}_2$ and $\vec{v}_3$, and are estimated by the Fuzzy decision explained in the beginning of this section (see Figure 4).

5 IMPLEMENTATION AND EXPERIMENTS

This section describes the experiments considered to validate the proposed methodology for auto-guided movements. For this purpose, it has been used the PA-10

manipulator system from Mitsubishi Heavy Industries, Ltd., which can be viewed on Figure 6. In this picture, the manipulator has a force sensor attached to give information about the fulcrum location and to allow interaction with the surgeon as well as the patient [13]. To complete the implantation, the optical tracker Polaris Spectra from NDI gives information on the surgeon's tools location. This sensor recovers data on surgeon's position and orientation for both tools simultaneously.

Once the physical system is described, this work proposes to take gauze to the surgeon's tool location. Therefore, the experiment goal is to find a trajectory in real time between the robot initial location and its target by avoiding the obstacle tool. The robot tool is supposed to be already inside the abdomen with gauze on its tool tip. Two situations are considered: one with no movement on the obstacle tool and another one with a dynamic behaviour of the surgeon.

5.1 STATIC SURGEON

As it can be shown in Figure 7, the situation where the obstacle does not move is solved by the APF behavior. Gauze is carried by the robot at its tool tip. The robot tool trajectory represented by asterisks is linear until it feels the obstacle tool potential field. Once the robot reaches this zone, it changes the trajectory by surrounding the obstacle at a certain distance (denoted ρ_0 in section 4.1). When the robot bypasses the obstacle zone, it continues the linear trajectory until it reaches the target.

Figure 7 also shows the velocity commanded to the robot. The APF velocity appears with dashed line, whereas the commanded one is solid. It can be noted that the module of the robot velocity is the maximum possible when the tool is far from the obstacle, but it shrinks when the distance between the robot tool and the obstacle is closer. The velocity finally increases again when the robot is near enough of the target.

5.2 DYNAMIC SURGEON

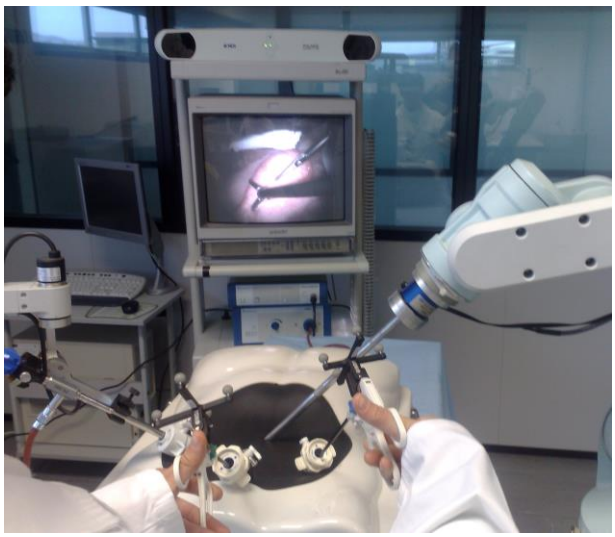

Figure 6 Experimental setup with the auto-guided algorithm proposed on this work.

Figure 8 shows the resulting trajectory of the robot tool with an asterisk line when the obstacle has freedom of movement. The obstacle trajectory, drawn as a series of dots over the obstacle tool, has been obtained by optical tracking measurements. It can be noticed that the robot not only displaces due to its proximity to the obstacle, but also changes its direction and speed. This way, the resulting trajectory has some zones where the robot has to adapt its trajectory because the obstacle tool is very close to the robot instrument.

The velocity behaviour of the robot tool tip is pretty similar to the static experiment, as shown in Figure 8. This time, the velocity of the obstacle is also represented by a dotted line. The main difference with the static obstacle appears when the robot tool is very close to the obstacle. It can be noted that the velocity in this situation is higher than the one planned by the APF behaviour, because the robot must follow the obstacle velocity.

6 CONCLUSION

This work is a first step on auto-guided movements for autonomous assistance to the surgeon. Developers have found a valid strategy for the robot assistant to find free-obstacle paths to a target location. In order to validate the methodology, a robotic arm system has been used for implementation purposes. An experiment where the robot should take gauze to the surgeon's tool has been developed with success.

However, as future works the author believes that a deeper interaction between the camera movements and the robot tool would improve auto-guiding tasks. Also, there is another important issue not covered in this paper: the robot-patient interaction. It is important to generate safe trajectories for the robot avoiding tissue collisions. There may be also tasks where the robot must directly interact with the patient.

REFERENCES

[1] Kumar R., Jensen P., Taylor R.H., Experiments with a Steady Hand Robot in Constrained Compliant Motion and Path Following. *IEEE International Workshop on Robot and Human Interaction*, 1999.

[2] Hynes P., Dodds G.I., Wilkinson A.J., Uncalibrated visual-servoing of a dual-arm robot for surgical tasks. *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 2005.

[3] Krupa A., Gangloff J., Doignon C., Mathelin M.F., Morel G., Leroy J., Soler, L., Marescaux J., Autonomous 3-D positioning of surgical instruments in robotized laparoscopic surgery using visual servoing. *IEEE Transactions on Robotics and Automation*, 2003.

[4] Gangloff J., Ginhoux R., Mathelin M., Soler L., Marescaux J., Model predictive control for compensation of cyclic organ motions in teleoperated

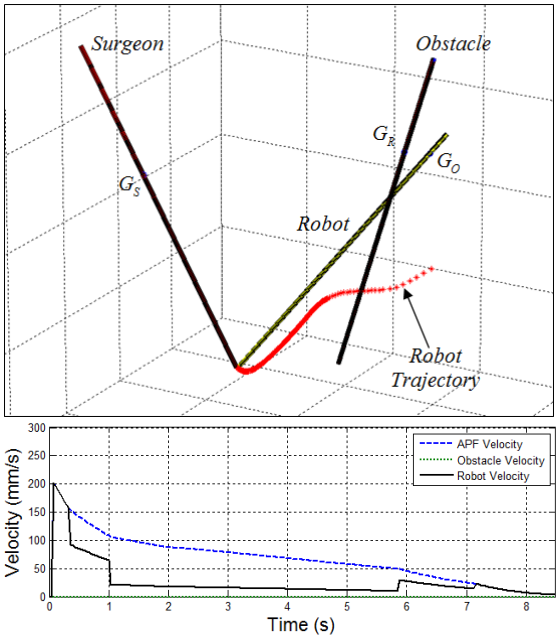

Figure 7 Robot trajectory when the surgeon's obstacle tool remains static (above) and resulting velocity modules (below).

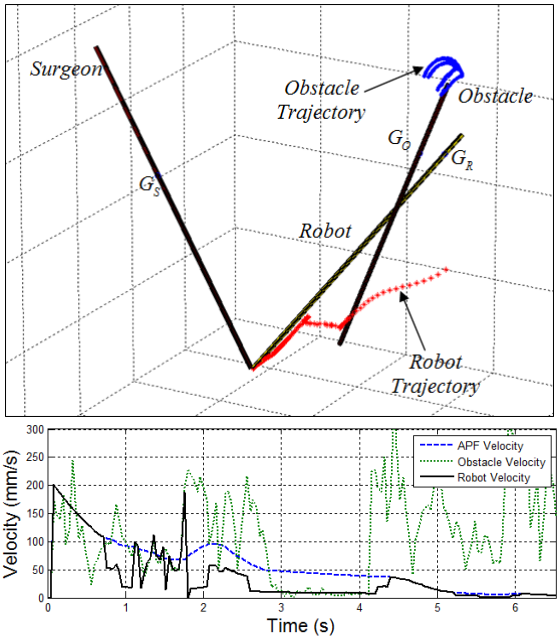

Figure 8 Robot trajectory when the surgeon's obstacle tool moves (above) and resulting velocity modules (below).

laparoscopic surgery. *IEEE Transactions on Control Systems Technology*, 2006.

[5] Ortmaier T., Gröger M., Boehm D.H., Falk V., Hirzinger G., Motion Estimation in Beating Heart Surgery. *IEEE Transactions on Biomedical Engineering*, 2005.

[6] Kang H., Wen J.T., Robotic assistants aid surgeons during minimally invasive procedures. *IEEE Engineering in Medicine and Biology Magazine, vol. 20, no. 1*, 2001.

[7] Chen G., Pham M.T., Redarce T., A semi-autonomous micro-robotic system for Colonoscopy. *IEEE International Conference on Robotics and Biomimetics*, 2008.

[8] Ho-seok S.; Ki-young K.; Jung-ju L., Development of the dexterous manipulator and the force sensor for Minimally Invasive Surgery. *International Conference on Autonomous Robots and Agents*, 2009.

[9] Dumpert J., Lehman A.C., Wood N.A., Oleynikov D., Farritor S.M., Semi-autonomous surgical tasks using a miniature in vivo surgical robot. *IEEE International Conference on Engineering in Medicine and Biology Society*, 2009.

[10] Mayer H., Nagy I., Knoll A., Schirmbeck E.U., Bauernschmitt R., The Endo[PA]R system for minimally invasive robotic surgery. *IEEE International Conference on Intelligent Robots and Systems*, 2004.

[11] Mayer H., Nagy I., Burschka D., Knoll A., Braun E.U., Lange R., Bauernschmitt R., Automation of Manual Tasks for Minimally Invasive Surgery. *International Conference on Autonomic and Autonomous Systems*, 2008.

[12] Brett P.N., Taylor R.P., Proops D., Coulson C., Reid A., Griffiths M.V., A surgical robot for cochleostomy. *IEEE International Conference of the Engineering in Medicine and Biology Society*, 2007.

[13] Bauzano, E., Munoz V.F., Garcia-Morales I., Estebanez B., Three-layer control for active wrists in robotized laparoscopic surgery. *IEEE International Conference on Intelligent Robots and Systems*, 2009.

[14] Iraji R., Manzuri-Shalmanit M. T., A New Fuzzy-Based Spatial Model for Robot Navigation among Dynamic Obstacles. *IEEE International Conference on Control and Automation*, 2007.

[15] Enxiu S., Tao C., Changlin H., Enxiu S., Junjie G., Study of the New Method for Improving Artifical Potential Field in Mobile Robot Obstacle Avoidance. *IEEE International Conference on Automation and Logistics*, 2007.

[16] Fernandez R., Mandow A., Munoz V.F., Garcia-Cerezo A., Real-Time Motion Control for Safe Navigation. *IFAC Symposium on Intelligent Autonomous Vehicles*, 1998.