Comparison of Control Modes of a Hand-Held Robot for Laparoscopic Surgery⋆

Oliver Tonet[1,2]⋆⋆, Francesco Focacci[2,1], Marco Piccigallo[2,3], Filippo Cavallo[1,2], Miyuki Uematsu[4], Giuseppe Megali[1,2], and Paolo Dario[1,2]

[1] CRIM Lab, Scuola Superiore Sant'Anna, Pisa, Italy
[2] EndoCAS Center for Computer-Assisted Surgery, University of Pisa, Italy
[3] IMT Institute for Advanced Studies, Lucca, Italy
[4] Major in Bioscience and Biomedical Engineering, Waseda University, Tokyo, Japan

Abstract. Teleoperated robots for minimally invasive surgery make surgeons loose direct contact with the patient. We are developing a hand-held, dexterous surgical robot that can be controlled with one hand only, while standing at the operating table. The instrument is composed of a master part (the handle) and a slave part (the tip). This work compares the performance of different control modes, i.e. different ways to map the degrees of freedom of the handle to those of the tip. We ask users to drive the tip along complex trajectories in a virtual environment, using the real master to drive a simulated slave, and assess their performance. Results show that, concerning time, users with no training in laparoscopy prefer a direct mapping of position and orientation, like in free hand motion. However, users trained in laparoscopy perform equally fast with our hand-held robot and, concerning precision, make a smaller number of errors.

1 Introduction

Minimally invasive surgery procedures bring numerous benefits to the patient, but severely hamper surgeons' perception and motor skills. Technology can provide methods and devices aimed at restoring – and possibly augmenting – visual, haptic and tactile feedback, motor coordination and dexterity: stereo endoscopes and displays, teleoperated master/slave systems, robotic interfaces and surgical instruments with many degrees of freedom (DoFs). Teleoperated systems, like the da Vinci Surgical System [1] by Intuitive Surgical Inc., bring some advantages at the cost of longer setup times and of cluttering the already crowded operating table, pushing the surgeon away from the patient: the surgeon operates at a console and only the robot is in direct contact with the patient. Experienced

⋆ This work has been supported in part by the FIRB-2001 Project *ApprEndo* (no. RBNE013TYM) and by *EndoCAS*, the Center of Excellence for Computer-Assisted Surgery (COFINLAB-2001 no. CLAB01PALK), both funded by MIUR, the Italian Ministry of Education, University and Research.

⋆⋆ Correspondence to: `oly@sssup.it`

surgeons tend to agree that in many procedures the benefits provided by teleoperated systems are not really needed during the whole surgical procedure, and they tend to prefer the traditional hands-on approach for routine tasks.

Led by these considerations, we are developing a hand-held surgical robot that can be operated by a surgeon with one hand only, while the other hand is free to use a traditional surgical instrument. The robot consists of a master part (the handle) and a 6-DoF slave part (the tip), connected together [2]. Selecting the best way to control the tip by means of the handle is one of the major issues in the design. A *control mode* is defined as a way to map the DoFs of the handle to the DoFs of the tip [3]. In this paper we present an experiment aimed at selecting the control mode that provides the most intuitive and efficient way to steer the instrument. We have developed a teleoperated system composed of custom handles mounted on a haptic interface that also serves as 6-DoF digitizer. On this system, we implemented an exercise mimicking a surgical gesture requiring high dexterity, namely knot tying, and asked subjects to perform the exercise with four different control modes in order to select the most efficient. Two control modes are feasible on a hand-held instrument, while two require kinematic separation between master and slave. The latter have been included to compare the performance of the hand-held configuration to master/slave teleoperation systems.

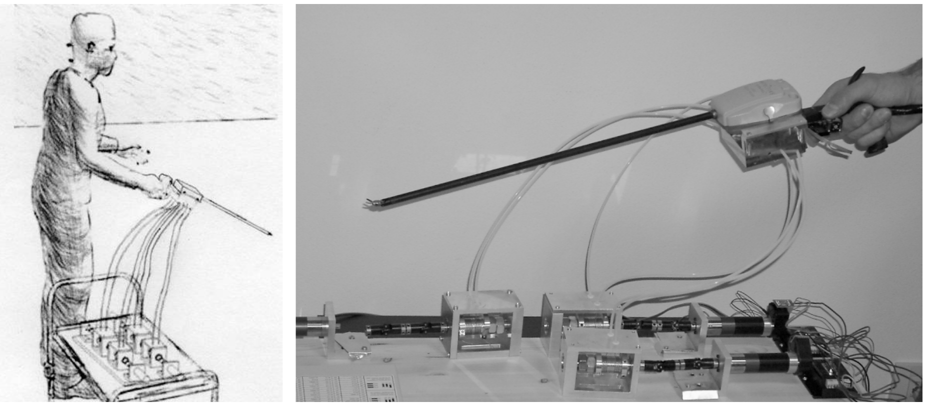

Fig. 1. Concept drawing of the lightweight hand-held laparoscopic robot (left) and the first prototype (right), using the EndoWrist of the da Vinci Surgical System as end-effector.

2 Methods

2.1 Control modes

For the comparative evaluation, we selected four control modes, obtained by switching two control parameters independently from each other. The first parameter is defined *PCM* (position control mode) and its values are *Tip* or *Handle*; the second parameter is called *ACM* (angle control mode) and its values are *Absolute* or *Relative*. PCM denotes how the position of the master is mapped to the position of the slave, while ACM specifies how the angular DoFs at the instrument tip are controlled. PCM=Tip means that the position of the master is directly mapped to the position of the tip. In other words, moving the master in one direction causes the tip to move in the same direction. By setting PCM=Handle we introduce a fulcrum constraint at the trocar position, hence moving the master in one direction causes the tip to move in the opposite direction (as in laparoscopic surgery). By setting ACM=Absolute, the orientation of the tool tip in 3-D space will be exactly the same as the orientation of the handle. By setting ACM=Relative, the angles at the tip joints are controlled incrementally through a joystick (moving the joystick in one direction makes the corresponding tip angle increase in the same direction). By combining the two PCMs with the two ACMs we obtain four different control modes that can be evaluated on our platform: *H_A* that means (PCM=Handle, ACM=Absolute), *T_A* (PCM=Tip, ACM=Absolute), *H_R* (PCM=Handle, ACM=Relative), and *T_R* (PCM=Tip, ACM=Relative). E. g. teleoperated master/slave systems like the da Vinci Surgical System have *T_A* control mode, whereas our hand-held robot has *H_R*. Of course, a hand-held instrument will have PCM=Handle, while PCM=Tip requires separate kinematics for the master and slave parts.

2.2 Teleoperated system with simulated robot

Figure 2 shows the architecture of the system used for the experiments. For the master system, we fabricated two types of handle, one for each ACM. To control the slave angles of the instrument in the ACM=Relative configurations, a joystick, driven by the user's thumb, has been mounted on the foil grip used as ergonomic handle. The joystick is a two-axis potentiometer and communicates with the host PC through an electronic board connected to the serial port. A switch has also been mounted in the foil grip in order to implement the stem roll movement in the virtual environment. For the ACM=Absolute configuration we fabricated a tweezers-like handle. The handles were connected to a Phantom Premium 1.0 by Sensable Inc., which is a 6 DOF localizer and 3 DOF force feedback interface. We used the Phantom to measure the master position; we only used the orientation encoders when ACM=Absolute.

To simulate PCM=Tip the handle is free to move in space and the position of the Phantom end-effector is mapped to the instrument tip position directly. To replicate PCM=Handle, the handle (foil grip or tweezers) is attached to the Phantom end-effector and the Phantom stylus, extended with an aluminum cylinder, is inserted through a trocar on an abdomen replica. Since the length of the instrument and the position of the trocar are known, we can compute the tip position by measuring the position of the Phantom stylus. The four configurations are depicted in Fig. 3. It is worth noting that in *H_A*, an additional gimbal ring was needed, allowing to control the handle orientation independently from the aluminum cylinder, constrained by the trocar.

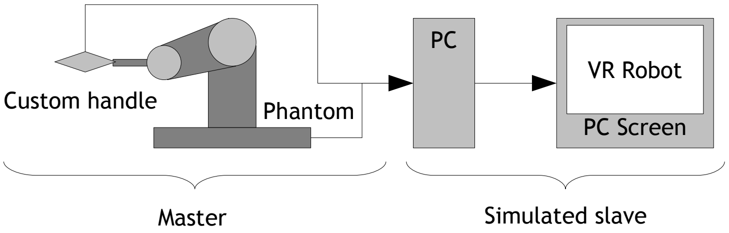

Fig. 2. The architecture of the system, with sensorized master and simulated slave.

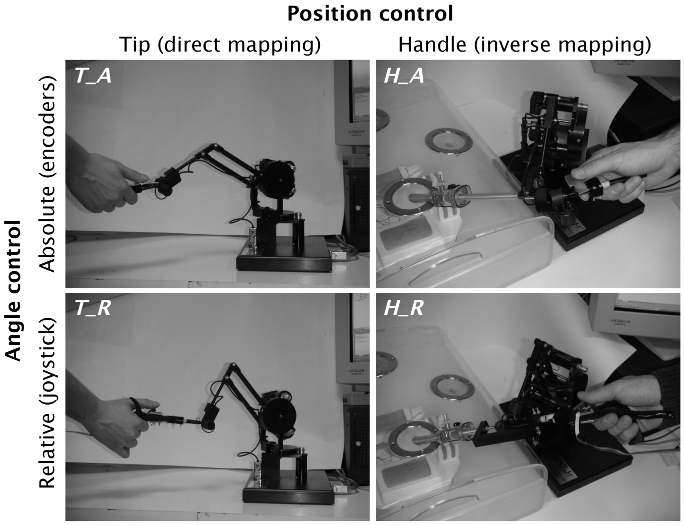

Fig. 3. The four control modes and the experimental setup.

The slave robot kinematics has been simulated in a virtual reality environment (see Fig. 4.right). We derived our robot design from the end-effector of the da Vinci, called EndoWrist. We exported 3D objects from the CAD models of the mechanical components and assembled them in a 3D scene graph using the OpenGL Optimizer library. We calculated the inverse kinematics of the EndoWrist in order to map the end-effector position and orientation correctly and reproduce realistic animation in the virtual environment.

2.3 Performance Assessment

To compare the four control modes we asked test subjects to reproduce, by means of the different interfaces, a gesture that follows a complex path in 3-D space, thus requiring high dexterity and use of all 6 DoFs of the instrument tip. The trajectory to reproduce was created by sampling, at 60 Hz, the coordinates of the instrument tip while a surgeon was mimicking a knot-tying gesture on the platform itself. The data were smoothed with a low-pass filter, interpolated by means of a cubic spline, and downsampled to obtain a final trajectory composed of 100 points.

In the virtual environment the trajectory is shown as composed of a sequence of spheres that must be pierced by means of a needle placed in the slave forceps. A ring, perpendicular to the needle, is displayed as a visual aid to identify the needle orientation, as depicted in Fig. 4. Only a few of the spheres composing the trajectories are shown at the same time, to keep the display clear. One sphere is marked in bright yellow: this is the active sphere. To cross the active sphere successfully with the ring, the user must slide the needle past the sphere center, without touching the sphere with the ring, and keeping the ring normal to the direction tangent to the trajectory. Actually, a tolerance angle of 60 deg has been used. When the active sphere has been crossed, the following sphere is marked as active and so on, until the whole trajectory has been correctly followed. As additional requirement, to cross the actual active sphere successfully, the two previous spheres must also have been crossed successfully. These spheres are highlighted when they have been crossed. Moving away from the trajectory resets the task and requires the user to cross the two previous spheres again. In the following, we will refer to this latter condition as *off* state; conversely, we define the *on* state when the user is crossing the spheres correctly. The position and orientation of the instrument tip, together with a time stamp and all interactions with the spheres are logged on file for offline analysis.

A group of five subjects with only some experience on a laparoscopic trainer was asked to complete two trials with each of the four control modes. The trials were performed during two days, with no predefined order, allowing the subject at least 10 minutes of rest between the trials. We decided not to involve expert surgeons at this stage in order to avoid biasing the results with dexterous movement schemes expert surgeons have acquired during prolonged practice with laparoscopic instruments.

2.4 Data Analysis

The period of the simulation cycle is between 15 and 16 ms. The position and orientation of the tool was sampled from the Phantom interface at every cycle, sufficient to describe human gesture [4], especially fine movements. performed during surgery [5]. The acquired data was interpolated and resampled at a constant rate of 70 Hz. Only low-frequency components are present in surgeon's movements. However due to artifacts and some error of instruments, 3D position data is also contaminated by high-frequency noise. Since our data analysis involves first, second and third derivatives, the acquired data are off-line filtered using a numerical fourth-order low-pass Butterworth filter, with cut-off frequency of 15 Hz.

For the evaluation of the exercises, we have computed the following parameters: total time spent (t), total time spent in *on* state (t_{on}), total time spent in *off* state (t_{off}), percentage of time spent in off state ($t := t_{\text{off}}/(t_{\text{on}} + t_{\text{off}})$), number of errors, mean velocity, mean acceleration, and normalized jerk [6].

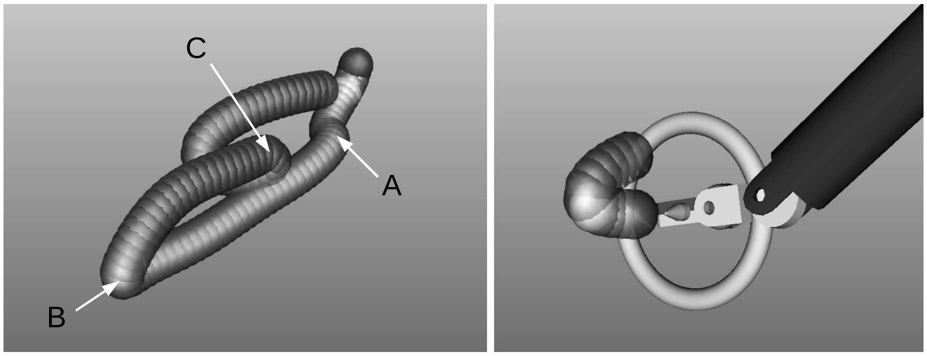

Fig. 4. The trajectory mimicking a knot-tying task as displayed in the virtual environment (left) and the simulated slave robot approaching the active sphere (right).

3 Results

Figure 5 shows the progression of the users during the exercises, separately for each control mode. The five users are called U1-U5. We have chosen to plot only the second trial in order not to overload the figure; the first trial has roughly similar shape. The horizontal axis shows which of the 100 spheres that build the trajectory the user was interacting with. The vertical axis shows the time, in s. The nearly horizontal segments correspond to parts of the trajectory where the user was progressing, crossing the spheres rapidly. The sloped segments correspond to difficult points in the trajectory, where the users spent much time in advancing to the next sphere. Comparing performance of users we can see that there are mainly three difficult points, roughly corresponding to spheres number 15, 45, and 70. These points correspond to passages with higher curvature, as highlighted in Fig. 4.left with the letters A, B, and C. We can also see that passage B is more difficult with ACM=Absolute (encoders), and passage C is more difficult with ACM=Relative (joystick).

Figure 6 analyzes time and number of errors of each user, using the four control modes. U1 and U2 were faster in performing the exercises, and also made a smaller number of errors. This result was expected, since they are more familiar with laparoscopy and also had some previous experience with laparoscopy simulators. From the histograms, it looks like there is no single control mode preferred by all users, however we can still highlight some basic trends.

Concerning time, it looks like the T_A control mode allows all users to perform the exercise in the shortest time. The reason is that T_A is the most intuitive way to control the needle, since there is a direct mapping of both position and angles, and is equivalent to moving one's hand in free space. Note, however, that U1 spends most of the time in t_{off} state, trying to get back to where he made a mistake. If we compare times of the users U1 and U2, who are a little trained in laparoscopy, it can be seen that, apart from T_R, they score quite similar with the other control modes. In particular U1 (the more trained in the usage of our

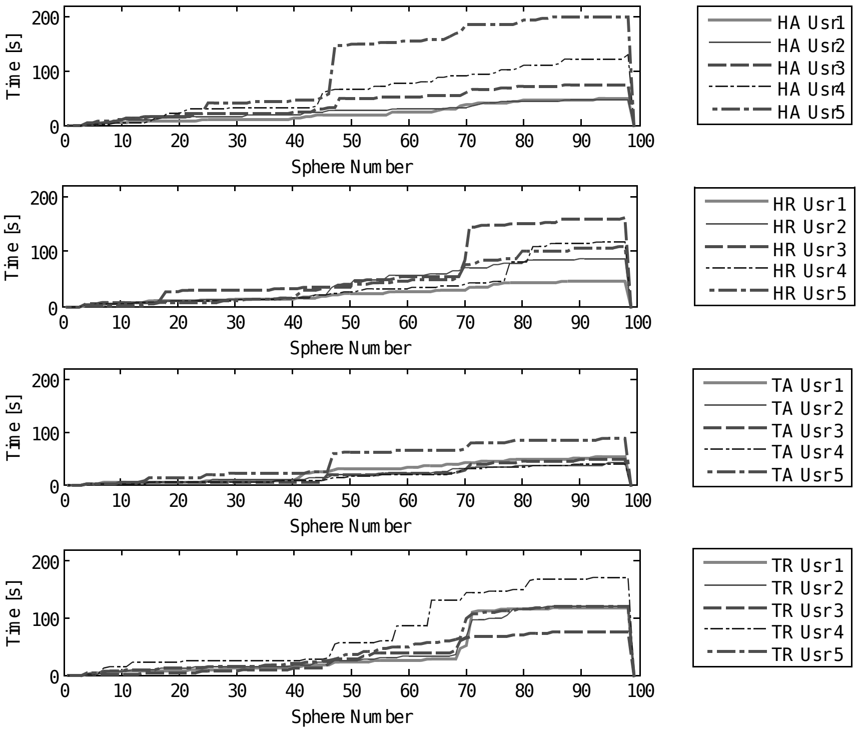

Fig. 5. Plots showing the time spent to complete the exercise by the users with the four control modes. Steep parts highlight difficult points.

hand-held robot), performs as well with H_R than with T_A, but spending much less time in t_{off} state. U2 has similar performance with H_R than with T_A.

Concerning number of errors, H_R always is the 1st or 2nd choice of 4 out of 5 users. In particular, U1 and U2 make less errors with H_R than with T_A.

Concerning mean velocity $\bar{v}$, mean acceleration $\bar{a}$ and normalized jerk, trends are less evident. It appears as $\bar{v}_{\text{on}}$ and $\bar{a}_{\text{on}}$ are less when PCM=Handle, suggesting that the trocar contributes to slow down, but also to control, the movements. Normalized jerk is higher when ACM=Relative, but this is easily explained since the movements of the joystick are, exactly, *in jerks*.

4 Conclusions

We presented a teleoperation platform to assess the performance of different control modes in laparoscopic surgery. The platform is based on a real master, with custom handles mounted on a Phantom haptic interface, and a simulated 6-DoFs slave. This platform has been used to measure performance of users, in order to select the most suitable control mode for a lightweight hand-held robot for laparoscopy that we are developing.

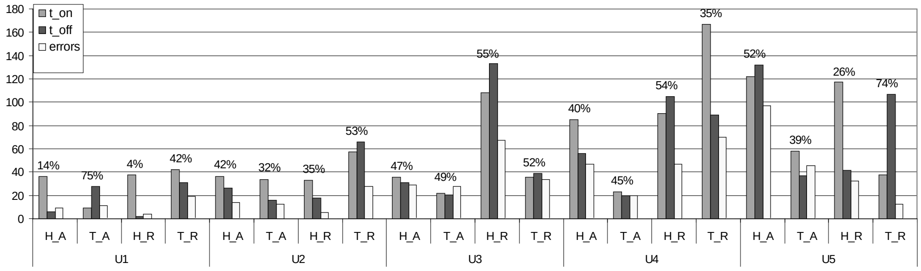

Fig. 6. Histogram showing, for each user and control mode, t_{on}, t_{off} and the number of errors (see text for the description). The percentage shown is the ratio $t_{\text{off}}/t_{\text{total}}$.

Results show that a direct mapping of position and angles (PCM=Tip, ACM=Absolute), like in the da Vinci system, is the most intuitive for users with no experience in laparoscopy and allows them to complete the exercises faster. However, users which have some training in laparoscopy, performed as fast with our prototype hand-held robot (PCM=Handle, ACM=Relative) and also made a smaller number of errors. This result encourages us to continue the development of the hand-held robot.

Future work will concern the assessment of the best control modes involving also expert surgeons. We expect them to perform well with the hand-held robot, since users U1 and U2 with only a little training in laparoscopy, scored very well. Moreover, we will implement additional exercises, requiring to perform dexterous movements in more challenging conditions.

References

1. Guthart, G.S., Salisbury, J.K.: The Intuitive^{TM} telesurgery system: overview and application. In: Proc. IEEE International Conference on Robotics and Automation – ICRA. (2000) 618–621
2. Focacci, F., Piccigallo, M., Tonet, O., Megali, G., Dario, P.: Design of a hand-held robotic instrument for dexterous laparoscopic surgery. Minim Invas Ther **14**(4-5) (2005) 308–9
3. Lai, F., Howe, R.D.: Evaluating control modes for constrained robotic surgery. In: Proc. IEEE International Conference on Robotics and Automation – ICRA. (2000) 603–609
4. Burdea, G.: Force and touch feedback for virtual reality. John Wiley & Sons, Inc. (1996)
5. Hannaford, B.: Experimental measurements for specification of surgical mechanisms and understanding of surgical skill. Lecture notes, Lecture notes of the European summer school on surgical robotics, Montpellier, France (2003)
6. Cotin, S., Stylopoulos, N., Ottensmeyer, M., Neumann, P., Rattner, D., Dawson, S.: Metrics for laparoscopic skills trainers: the weakest link! In Dohi, T., Kikinis, R., eds.: Proc. Medical Image Computing and Computer-Assisted Intervention – MICCAI. Volume 2488 of Lecture Notes in Computer Science., Berlin, Springer (2002) 35–43