



**POLITECNICO  
MILANO 1863**

# Implementation and assessment of Assistance-as-Needed Virtual Fixtures for Surgical Training with a *da Vinci* surgical robot: an experimental study

*Politecnico di Milano - Master of Science in Biomedical Engineering*

*Academic year 2022/2023*

**Candidate: Alberto Rota**

**Supervisor: prof. Elena De Momi**

## 1 Introduction

Since the introduction of Robot-Assisted Minimally Invasive Surgery (RAMIS) in the healthcare market, patients have benefited from the diminishing of post-operative complications and an increase in the safety of procedures, while surgeons have been empowered by sophisticated robotic systems allowing the execution of complex tasks in adverse contexts.

Surgical teams have been joined by robotic systems due to the number of benefits that result from the collaborative effort of the surgeon and the robot, combining the know-how and adaptation skills of the former and the high accuracy and stability of the latter. Such a synergistic perspective highlights the role of the Human-Robot Interaction (HRI) paradigm, which involves all aspects of understanding, designing, and evaluating robotic systems for use by or with humans

[1]. Most of the surgical robotics solutions on the market consist of a teleoperation console that interfaces with the practitioner and, separate from it, the surgical robot itself, which mimics the movements of the surgeon in real-time. This setup allows for higher motion accuracy, tremor filtration and magnified viewing of the surgical area; nonetheless tactile forces, friction and texture perception are excluded from the so crucial visuo-haptic feedback loop that would guide the surgeon in a standard "non-robotic" procedure.

This work, specifically, studies and evaluates the role of haptic Virtual Fixtures in the context of surgical training. Virtual Fixtures (VFs) are high-level control strategies employed for assisting humans in man-machine collaboration tasks [2]. In the context of surgical robotics, VFs assist the surgeon by providing haptic guidance at the level of the "master" manipulator, generating mechanical forces and torques which

re-direct the motion of the surgeon’s hands. These assistance strategies may be most beneficial in the training process that aspiring surgeons undertake, which often takes place in a simulated virtual environment. Exploiting the customizability of simulated surgical tasks together with the implementation of VFs will enhance the process of learning key surgical skills, in terms of performance, retention and transfer.

## 2 State of the art

Surgical robotics companies usually commercialize a simulation framework in parallel to clinical robots. A survey of the most relevant training simulators on the market was conducted in [3], a review that also assesses the suitability of virtual environments as comparable to the one of dry-lab setups. However, since no commercially available clinical system implements a force-based assistive modality similar to the one of virtual fixtures, none of the training simulators employ VFs either. The role of such assistive strategies in a real surgical scenario is still uncertain and shall be assessed only through an extensive clinical trial, and as a matter of fact the vast majority of training protocols implementing VFs regards *ad-hoc* systems like [4], which are limited both in terms of tasks implemented and in terms of evaluation protocols. Indeed, few studies [5] have evaluated the trainee’s performance on multiple diverse tasks and over the course of multiple training days, and none has yet investigated the role of haptic assistance on skill retention and skill transfer.

This work proposes an evaluation of the role of VFs in surgical training with a multi-day experimental protocol articulated in two phases, designed in order to highlight the difference in the transfer and retention of skills between a control group and an assisted group.

## 3 Materials and methods

### 3.1 Surgical Simulator

This research was conducted on a *daVinci®* surgical robot integrated with the open-source dVRK [6]

framework. The Master Tool Manipulators (MTMs) of the *daVinci®* are in fact equipped with motors usually employed for the sake of homing and calibration; the VF forces and torques are generated by energizing these motors according to the inverse kinematics of the manipulators. A ROS [7] framework manages the communication between the teleoperation console of the dVRK and the virtual surgical scene, which is built upon the Unity® physics engine: therefore, the real Patient-Side Manipulators (PSMs) do not move during teleoperation, as the joint coordinates are communicated only to the virtual 3D objects.

The simulator comprises eight surgical tasks, four of which (*Path*, *Rings*, *Pillars* and *Exchange*) are simplistic training tasks built with objects of simple geometry, while the remaining four (*Liver Resection*, *Nephrectomy*, *Thymectomy* and *Suturing*) emulate *in-vivo* surgical procedures and are therefore more realistic. Fig.?? collects snapshots of the tasks. All of these are constructed and set-up in order to be as challenging as possible in relation to a specific surgical skill. A set of fundamental pre-operative and intra-operative skills that any robotic surgeon should acquire during training was proposed in [8]. Specifically:

- *Path* and *Liver Resection* require articulate wrist motion and stability
- *Rings* and *Nephrectomy* survey the depth perception skills
- *Pillars* and *Thymectomy* are hand-eye coordination tasks
- *Exchange* and *Suturing*, both bi-manual tasks, challenge the capabilities in terms of instrument exchange

Each training task having a corresponding emulated surgical procedure will allow, during the experimental phase, to evaluate the transferability of skills.

The simulator also exploits 3D viewing capability of the High-Resolution Stereo Viewer (HRSV) installed on the teleoperation surgical console: two virtual

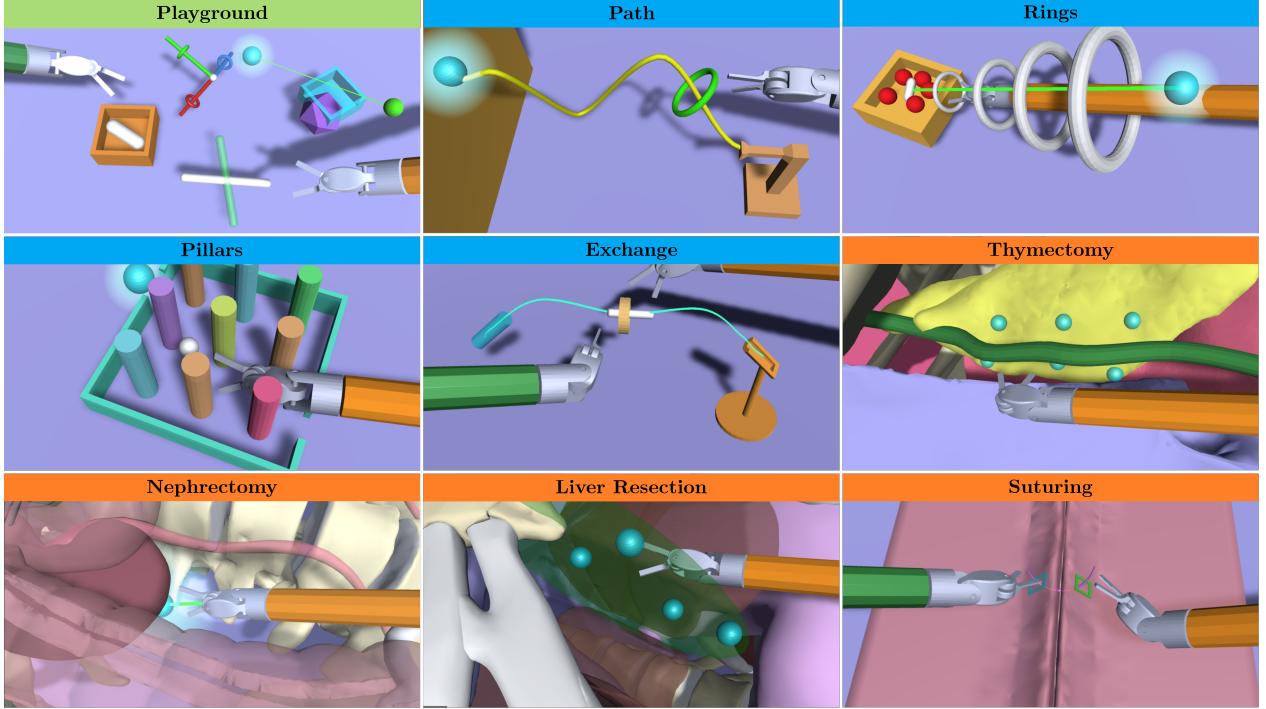


Figure 1: *Snapshot of the simulated surgical tasks, with the respective denomination. Training tasks have blue headlines, while realistic evaluation tasks have orange headlines. Playground is a propaedeutic task and isn't featured in the*

cameras are positioned in the Unity<sup>®</sup> scene at a distance of 5.3mm, as their feed is sent separately on the left and right oculars at the console achieving a three-dimensional perception of the virtual environment.

### 3.2 Virtual Fixtures

During teleoperation the surgeon is de-coupled from the patient: this is necessary in order to exploit the benefits of a robotic solution, which requires the system to be "in between" the practitioner and the patient in order to enhance the surgical experience. However, this de-coupling removes the haptic component from the motion control feedback loop of the surgeon, who is used to relying on the sense of touch when operating with a conventional approach. For this reason, this work analyses how the introduction of haptics in a virtual surgical environment may enhance the training phase, acting as a guidance and error-correcting medium.

In the context Virtual Fixtures, also known as Active

Constraints, a haptic force is applied to the manipulators at the surgical console, which re-directs the motion of the surgeon's hands in case of improper or unsafe maneuvers. The magnitude and direction of the VFs force are computed from the PSM position and orientation relative to the surgical space and the position of objects in the scene, in a true feedback fashion. A transformation matrix  $T_{PSM}^{MTM}$  mapping the PSM operatory space to the MTM console space is necessary in order to sensibly use the feedback force and torque as corrective medias.

$$\mathbf{f}_{MTM} = T_{PSM}^{MTM} \cdot \mathbf{f}_{PSM} \quad (1)$$

$$\mathbf{T}_{MTM} = T_{PSM}^{MTM} \cdot \mathbf{T}_{PSM} \quad (2)$$

Most of the assistance strategies implemented here will use the distance from the PSM to the target or obstacle as the primary metric for determining the intensity of the feedback force or torque. However,

different surgical tasks and situations require a level of control over how the distance is taken into account, and for this reason a sigmoidal mapping function is employed for the normalization of the linear or angular error into a suitable interval. Specifically, such mapping is formulated as:

$$f_{map}(x) = \frac{1}{1 + e^{5\delta w(x-t-h)}} \quad (3)$$

with  $\delta = +1$  for guidance VFs and  $\delta = -1$  for avoidance VFs. Here:

- $t$  is the fixture *threshold*, hence the value at which the sigmoid starts to significantly increase from zero
- $h$  is the distance from the threshold at which *half* of the maximum force is provided
- $w$  controls the *width* of the linear region, hence the steepness of the curve

For example, if  $t = 2\text{mm}$  and  $h = 3\text{mm}$  the surgeon will start to feel a force for errors higher than  $2\text{mm}$ , and at  $5\text{mm}$  he/she will half of the maximum force that can be delivered.

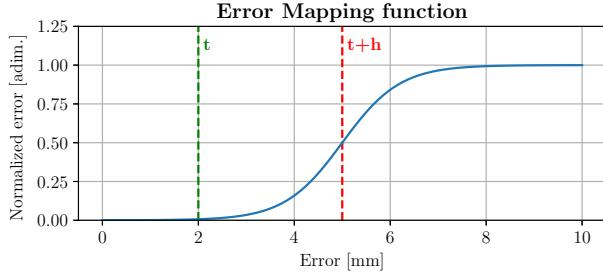


Figure 2: Plot of the Error Mapping function. The position of  $t$  and  $t+h$  can be set manually to achieve a suitable behavior of the assistance strategy

For the purpose of stability and, therefore, safety in the assistive feedback loop, all the VFs implemented have an elastic component, proportional to the error mapped with the sigmoidal function, and a viscous component proportional to its rate of change, damping the possible oscillatory instabilities. The optimal visco-elastic balance is heavily task-dependent and surgeon-dependent, and for this reason it is tuned by manually setting the values of  $K_f$ ,  $K_T$ ,  $\eta_f$  and  $\eta_T$

in the equations

$$\mathbf{f} = K_f \cdot \mathbf{f}_{elastic} + \eta_f \cdot \mathbf{f}_{viscous} \quad (4)$$

$$\mathbf{T} = K_T \cdot \mathbf{T}_{elastic} + \eta_T \cdot \mathbf{T}_{viscous} \quad (5)$$

which will ultimately result in the haptic outputs provided to the actuators.

The four types of VFs featured in the surgical simulator are described in the following paragraphs:

**Trajectory Guidance** Given a reference trajectory - planned in the pre-operative phase - this VF steers the surgeon's hands in order to align the robotic End-Effectuator (EE) with the closest point on the trajectory itself. The feedback force attracts the EE towards the reference, while the torque rotates it so that it's aligned with the tangent vector at the closest point.

**Insertion Guidance** This VF aids the surgical insertion of the EE towards a target point, maintaining the path of the tooltip stable inside an insertion cone. Here the feedback force has the same direction of the radial conical coordinate and a magnitude which is proportional both to the distance to the cone centerline and the distance to the target point. With this configuration, the PSM's tooltip will be kept inside the reference insertion cone.

**Surface Guidance** Similarly to the *Trajectory Guidance* VF, the distance vector to the closest point belonging to a reference surface (planar or non-planar) determines the force feedback, while the torque aligns the EE to the tangent plane at the closest point.

**Obstacle Avoidance** Having identified a 3D mesh as an obstacle, the distance to the robotic tooltip and its rate of change are used for computing the force feedback. By setting  $\delta = -1$  in Eq.3, the sigmoidal mapping function is flipped and higher forces will be generated from small distance errors.

### 3.3 Clinical Validation

Two resident surgeons from the *Istituto Europeo di Oncologia*, both regularly performing RAMIS procedures with the *daVinci®* robot, kindly dedicated their time in testing the surgical simulator in all its aspects, from the motion truthfulness to the complexity of the wrist articulation to the invasiveness and visco-elastic balance of the virtual fixtures. Their opinion and expertise were precious and insightful tools that guided the development towards a clinically validated robotic surgical simulator.

Moreover, the most expert resident surgeon allowed to have its performance recorded when practicing with the simulator, which will be considered "peak performance" in the experimental analysis.

### 3.4 Experimental Protocol

The effectiveness of the haptic virtual fixture paradigm has been assessed with an experimental study where the performance of un-assisted subjects in a control group was compared to the one recorded from subject to whom was provided haptic assistance. Since the aim of this work is to establish the role of VFs in the training context, eight novice subjects with little to no experience with surgical robots were recruited for the study. Subjects were 25% females and 75% males, between 23 and 27 years of age, all right-handed and either had never teleoperated a surgical robot or did it less than 5 times. Assignment to the control or assisted group was random.

The subjects underwent a week-long training phase:

- On *Day 1*, they were given a concise explanation about the *daVinci®* surgical system, the simulator, haptics, and the training protocol. Later, each subject experienced 5 minutes in a *Playground* environment (Fig.?? ), purposely designed to understand the elementary mechanics of teleoperation, clutching and object grasping. No performance metric was recorded at this time
- From *Day 2* to *Day 5* they were asked to execute the four training tasks of the simulator

(*Path, Rings, Pillars and Exchange*), each one for a total of 3 repetitions. Tasks appeared sequentially in a random order, contributing to un-bias the results. At the beginning of each daily session, subjects could interact with the *Playground* task for 1 minute at maximum.

- On *Day 6* and *Day 7* subjects were given a break period and no training was performed
- On *Day 8* they were asked to execute the four validation tasks of the simulator (*Thymectomy, Nephrectomy, Liver Resection and Suturing*), each one for a total of 3 repetitions. Again, tasks appeared sequentially in a random order, contributing to un-bias the results.

### 3.5 Metrics

A quantitative estimation of surgical performance is obtained by combining metrics recorded in real time during the execution of the task. The simulator logs these metrics autonomously detecting when the user initiates the execution and when the task is completed, at a framerate of 30 Hz. The metrics are:

- $D$ : Distance error, from the EE to the closest point of the target (trajectory, surface or object)
- $A$ : Angular error, from the EE to tangent vector in the closest point of the reference trajectory or to the tangent plane in the closest point of a surface
- $F$ : Feedback force, computed from the VF. The force is computed also for the subjects in the non-assisted control group, but it's not delivered during the operation
- $T$ : Feedback torque, computed from the VF. The torque is computed also for the subjects in the non-assisted control group, but it's not delivered during the operation
- $M$ : Missed exchanges. When, in the bi-manual tasks, an object is dropped when passing it from the left PSM to the right, or viceversa

- $C$ : Clutch time, the fraction of the total task time during which the clutch pedal was pressed for repositioning

These values are logged at each frame of the task execution, and are then averaged once the task is complete. Indeed, a statistical analysis will be conducted on these averaged values, where  $X_{u,r}$  is the averaged generic metric  $X$  for the  $r$ -th repetition performed by the  $u$ -th user.

Considering "optimal execution" the one achieved by the resident surgeon, the quantitative performance index  $P$  is the ratio of the task metrics gathered from the optimal execution from the surgeons and the one recorded from the training subjects.

## 4 Results

Fig.?? shows the performance trends for the 4 training tasks, where the performance at each repetition is the average among the subjects in the assisted or control group. Apart from the *Pillars* task, the trends are increasing both for the assisted group and the unassisted group. Most significantly, the performance in the assisted group is consistently higher than the one in the control group in all training tasks.

## 5 Discussion

## 6 Conclusions

## References

- [1] Michael A. Goodrich and Alan C. Schultz. Human-robot interaction: A survey, 2007.
- [2] Stuart A. Bowyer, Brian L. Davies, and Ferdinando Rodriguez Y Baena. Active constraints/virtual fixtures: A survey. *IEEE Transactions on Robotics*, 30:138–157, 2014.
- [3] Justin D. Bric, Derek C. Lumbard, Matthew J. Frelich, and Jon C. Gould. Current state of virtual reality simulation in robotic surgery training: a review, 6 2016.
- [4] Yanping Lin, Xudong Wang, Fule Wu, Xiaojun Chen, Chengtao Wang, and Guofang Shen. Development and validation of a surgical training simulator with haptic feedback for learning bone-sawing skill. *Journal of Biomedical Informatics*, 48:122–129, 2014.
- [5] Nima Enayati, Allison M. Okamura, Andrea Mariani, Edoardo Pellegrini, Margaret M. Coad, Giancarlo Ferrigno, and Elena De Momi. Robotic assistance-as-needed for enhanced visuomotor learning in surgical robotics training: An experimental study. pages 6631–6636. Institute of Electrical and Electronics Engineers Inc., 9 2018.
- [6] Peter Kazanzidesf, Zihan Chen, Anton Deguet, Gregory S. Fischer, Russell H. Taylor, and Simon P. Dimaio. An open-source research kit for the da vinci® surgical system. pages 6434–6439. Institute of Electrical and Electronics Engineers Inc., 9 2014.
- [7] Stanford Artificial Intelligence Laboratory et al. Robotic operating system.
- [8] Roger Smith, Vipul Patel, and Richard Satava. Fundamentals of robotic surgery: A course of basic robotic surgery skills based upon a 14-society consensus template of outcomes measures and curriculum development. *International Journal of Medical Robotics and Computer Assisted Surgery*, 10:379–384, 9 2014.

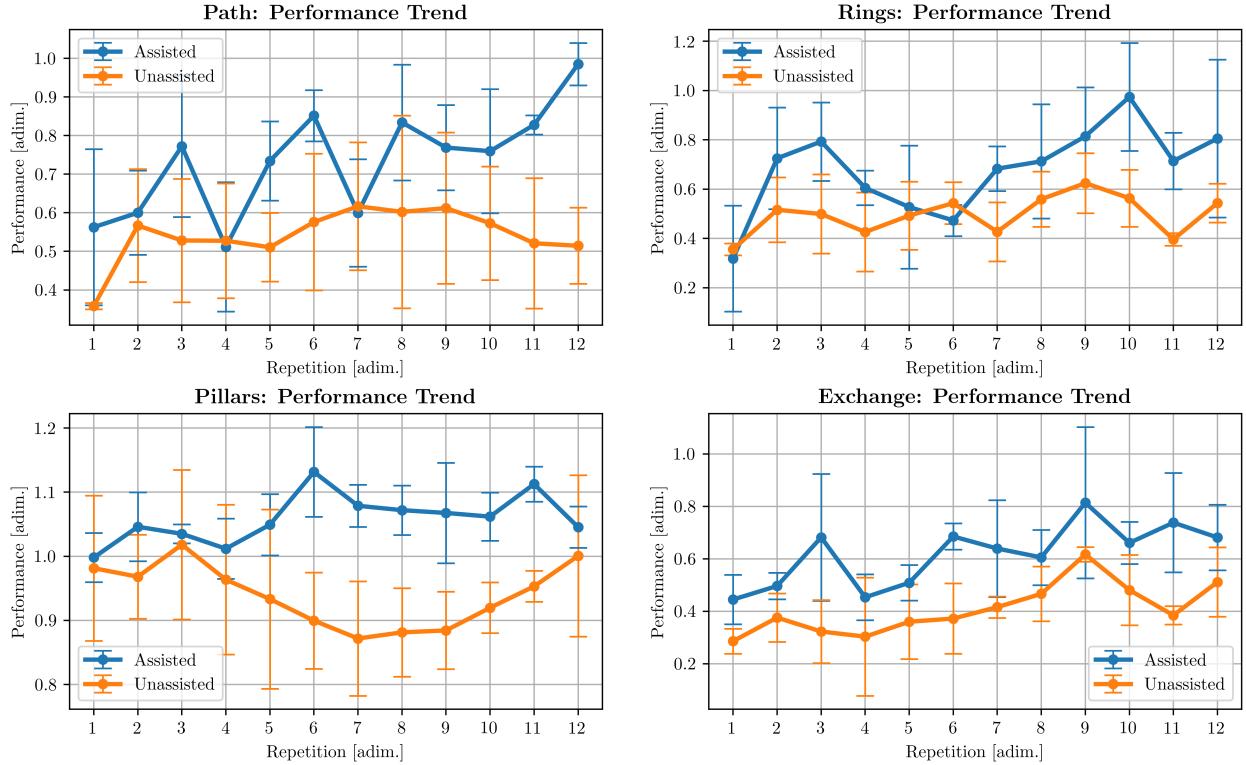


Figure 3: *Performance Trend for the four Training Tasks. The value at each repetition is averaged across the subjects belonging to the same group. Errorbars show the standard variation at each repetition*

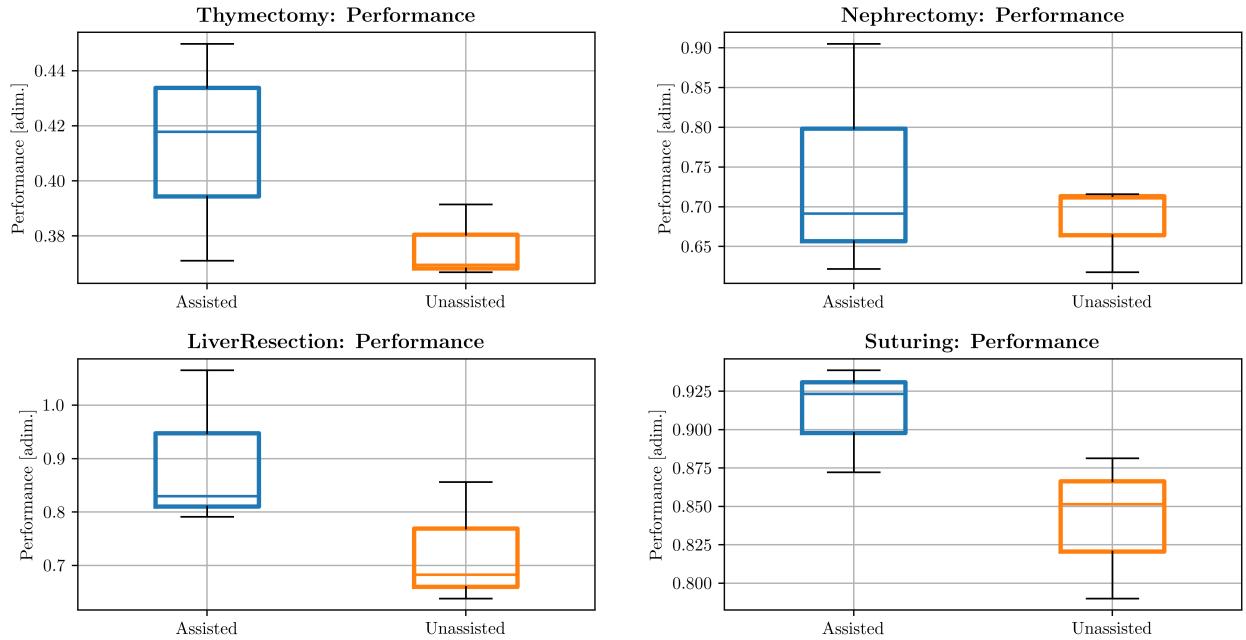


Figure 4: *Boxplots of the average performance of assisted subjects (blue) and unassisted subjects (orange) on the four validation tasks*