Towards a Dual-User Haptic Training System User Feedback Setup

Angel Ricardo Licona, Guillermo Zamora de la Pena, Oscar Diaz Cruz, Arnaud Lelevé, Damien Eberard, Minh Tu Pham

Towards a Dual-user Haptic Training System User Feedback Setup

Angel Ricardo Licona[1], Guillermo Zamora de la Pena[1], Oscar Diaz Cruz[1], Arnaud Lelevé[1][0000−0001−5550−9072], Damien Ebérard[1], and Minh Tu Pham[1]

Univ Lyon, INSA Lyon, Ampère (UMR 5005), F-69621, Lyon, France,
arnaud.leleve@insa-lyon.fr,
WWW home page: http://www.ampere-lab.fr

Abstract. This paper introduces preliminary works on building an experimental end-user evaluation for dual-user haptic systems for hands-on training. Such systems bring together the advantages of haptic computer-based training systems and those of supervised training where an expert trainer actively helps in the learning process. The first results mainly permitted to highlight several technical and organizational issues to overcome in a close future. The objective of this project is to test other architectures such as those listed in the state-of-the-art section, to provide comparative conclusions about the pros and cons of each one.

Keywords: Computer Simulation · Dual-user System · Hands-On Training · Haptic interfaces · Shared Control

1 Introduction

In many gesture-based profession, dexterous manipulation is necessary. Staff then requires initial and continuing hands-on training on regularly evolving methods. In Medicine, simulators such as cadavers and animals have been a convenient way to learn by trial for decades in universities. However, due to the growing cost of providing them and ethical issues, phantoms are increasingly used. Yet, phantoms provide a limited set of common cases to practice on. Nowadays, medical trainees are still offered too few opportunities to perform hands-on training during their curriculum, due to limited access to hands-on training resources. For instance, epidural anesthesia requires at least eighty attempts to be mastered enough to practice it on a real patient [14] but only a few of them are performed during the studies. Therefore, cost-efficient solutions to be used during supervised sessions but also autonomously are necessary for a sufficient hands-on training during their curriculum.

Over the last decade, Computer-Based Simulators (CBS) have been designed to overcome the aforementioned drawbacks [15]. Virtual patients, parameterized on-demand, provide an infinite set of medical cases with various difficulty levels. Such simulators have progressively improved to provide trainees with more realistic environments, in 2D and more recently in 3D [2]). In haptic training simulators, the additional force feedback provides a realistic tool behavior, which leads to efficient training for advanced tasks such as suturing [13]. These systems feature a haptic interface (i.e. a device by which tool-environment interaction forces are transmitted back to a human operator based on their hand motion) which acts as a master and a software architecture which connects the master to the slave (a software simulating a virtual tool inside a virtual environment, or a real robot handling a real tool).

Common training simulators provide only solutions to train oneself. But, for some difficult cases, the implication of a trainer remains necessary. He/she can guide the trainees' motions to accomplish accurate and efficient gestures. Traditionally, he/she directly guides the hands of a trainee but this "four-hand fellowship" does not permit the trainee to feel and dose the correct level of force to apply on their tool. Dual-user systems are a practicable solution to this problem, as they can reproduce this important force information to both users, each one interacting with their haptic interface. These systems extend the aforementioned master-slave architecture by adding a second master, as shown in Figure 1. Users share the slave control according to a dominance factor ($\alpha \in [0,1]$). When $\alpha = 1$ (respectively 0), the trainer (respectively trainee) has full control over the slave, and the trainee's (respectively trainer's) device follows the slave motions. When $0 < \alpha < 1$, both users share the slave control with dominance (over the other user) which is a function of α. According to the architectures found in the literature, the effect of α on the force feedback provided to the users differs. These differences are highlighted in section 2. It has been stated [7] that the existing solutions did not fit all the requirements for efficient hands-on training where interaction forces have to be learned through the haptic architecture. Thus, a dual-user control architecture had been first introduced: the *Energy-based Shared Control* (ESC) [5]. It reuses the robust control approach introduced by Stramigioli [12], which ensures passivity even in presence of robot model uncertainties, limited bandwidth, or non-linearities (saturation). It thus provides robust passive and compliant interfaces. We reuse in this paper a n degree-of-freedom (dof) version, for haptic devices with the same kinematics and with an adaptive dominance mechanism [7].

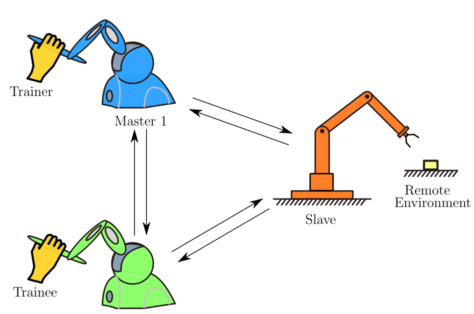

Fig. 1. Architecture of a dual-user system

The objective of this paper is to introduce the development of end-user experiments aiming at evaluating the performance of dual-user architectures for hands-on training. It is applied to the ESC aforementioned architecture. In this system, trainers and trainees manipulate their haptic interface. They share the same slave robot. They are all located in the same room.

This paper is organized as follows. Section 2 describes the dual-user concept in the literature. Section 3 introduces the method used for this experimentation, and sections 4 and 5 the preliminary results and conclusions.

2 Related Works

Various architectures have been proposed for the control of dual-user haptic systems. The "shared control" concept has been introduced by Nudehi *et al.* in [8] for Minimal Invasive Telesurgical Training purpose, i.e. to "allow experienced surgeons mentor trainee surgeons through shared control of a surgical robot". Unfortunately, each user is provided with feedback forces proportional to the difference of position of their device, according to α: this dual-user architecture can only be used to train users on motions, not on tool-interaction forces.

Khademian *et al.* introduced two distinct architectures [4]. The first one is the "Complementary Linear Combination" (CLC) architecture which provides feedback forces combining the environment and the other user forces. The desired position and force commands for each device are a complementary weighted sum of positions and forces of the other two devices. Therefore, when trying to perform a mentoring (resp. evaluation) session, the trainee (resp. trainer) has no direct force feedback from the environment, only from the other user. This is why the authors advise using this system with values of $\alpha \in \{0.2, 0.5, 0.8\}$ to get the best transparency. Yet, in these cases, both users influence the slave trajectory, which may lead to inaccurate motions, not compliant with mentoring (resp. evaluation) modes. In the second architecture, based on "Masters Correspondence with Environment Transfer" (MCET) [4], both user devices follow the motion of each other and the effort fed back to both users is half the force applied by the environment on the slave tool. Yet, according to the authors, this architecture transmits a distorted environment force to both users.

Ghorbanian *et al.* defined two dominance factors α and β [3]. α determines the balance of authority between the trainer and the trainee, while β indicates the supremacy of both trainer and trainee over the slave robot. They set a nonlinear relation between α and β to adjust the authority of the leader (the user for which $\alpha > 0.5$) over the slave. It also supports users located at distance with uncertain communication channels. However, when $\alpha = 1$, the desired Master 2 force is only linked to the position of Master 1, not to the tool-environment interaction force.

An interesting work has been introduced by Shamaei *et al.* [11] where two dominance factors are used. They experimentally evaluated, in terms of impedance matching between trainer and trainee during soft and hard interactions, an architecture which tracks the positions of all three devices. Yet, no information about force tracking error is provided. Also, the authors specify that this approach is limited in terms of control freedom as derivative operators may destabilize it.

In [10] the users' haptic feedback is a weighted (with α) sum of the virtual tool-environment interaction force and the other user interaction (with their master device) force. Experiments show that when $\alpha = 0.5$ and both users follow very close trajectories, they feel very close force feedback, in presence of time delays between each master and the slave. This is an interesting result but only limited to cooperative mode. We also need this kind of behavior in the demonstration and evaluation modes.

Furthermore, in a dual-user hands-on training system, the authority factor α should be switched anytime by the trainer. The architectures should guarantee that fast changes of α may not destabilize the system. Every LTI-based models do not inherently guarantee the stability of a variable parameter. At best, a LMI approach can ensure that the system is stable for every value of α. Only a few architectures guarantee some robustness versus α variations, employing a Lyapunov function [3, 10] or through an unconditional stability approach based on Llewellyn's criterion [11]. This short review shows that different approaches have been introduced in the literature but it remains difficult to compare them from a user point of view on their capability to train users on gestures requiring precision guiding and correct force level transmission. This motivates the work illustrated in this paper, for a generic experimental setup able to provide such comparative information.

3 Proposed Method

3.1 Typical scenario to test

The following typical use case helps determine the main requirements of the system. Suppose, at first, that the trainer (an experienced surgeon) aims at demonstrating the right trajectories of their surgical tool to perform a task featuring free motions and some tool–environment contacts. This implies that they require realistic force feedback to dose their force, as in a bilateral teleoperation context. The trainer manually sets $\alpha = 1$ to become the **leader** (the trainee becomes the **follower**): it is a **mentoring mode**. They then get full force feedback from the slave to perform their task as if they were handling the real instruments. Meanwhile, the trainee's device follows the trajectory of the leader one. If the trainee deviates from this reference trajectory when in free motion, the compliance of the device brings them back to the right position. In case of interaction between the tool and its environment, the trainee can also feel in their hands the right level of effort to provide to the tool, through a display that guides them to set their device in the right position with the right applied force. Afterward, the functioning can be inverted by reversing α so that the trainee manipulates and the trainer follows and evaluates trainee's motions and applied forces: it is an **evaluation mode**.

3.2 Tested Dual-User Architecture

In a first approach, we reused the ESC [7] architecture, which guarantees, by means of a Time-Domain Passivity Controller (TDPC), the passivity of the system (and then its stability) even for nonlinear models and active users and the environment. We already showed in simulation [6] that ESC had equivalent performance in terms of low-level tracking functions compared to CLC and MCET architectures. For space reasons, ESC is not recalled in this paper. It is detailed in a recent paper.

In the future, other architectures such as those listed in the previous section will be used to provide comparative studies. For a comparative study between different architectures, a virtual slave should be preferred as it permits to obtain more repeatable experiments. Some virtual fixtures can help limit the trajectories and then objectively compare them.

3.3 Objectives and Requirements

The main objective of this setup is to provide end-user evaluation of such dual-user architectures for teaching (from the trainer point of view) and learning (from the trainee's point of view) experiences, on simple gestures which require haptic feedback.

It is necessary, as prerequisites, that each user context (typically their previous experience on haptic systems) be known. The setup requires recording objective user gesture performance information (feedback forces, positions, timing, and potentially undesired collisions). We also need to get comfort feedback from the users (their subjective opinion) on completion. Users must be proposed simple exercises to avoid any fatigue as these kinds of devices requires much concentration. We have to train them a minimum on using such haptic devices to avoid mixing this learning with real gesture learning. In practice, we had to take into account the limited work volume of the Geomagic 3D Touch haptic devices and over-all their limited maximum force (≈3N).

3.4 Protocol

To rapidly create a simple setup, we chose a real slave (the same haptic device as for masters). It has the advantage of being tangible, easier for us to set up and to be understood by users, and more realistic than a virtual one. The main drawback is that the repeatability is lower and variables (forces, positions, ...) are easier to record in a virtual world. Eight propositions of exercises were studied: surgical game, tying a knot, dictating dimensions, cutting something, following a path, building a Jenga, writing, and stitching. These proposals were evaluated on their capacity of being performed with the provided haptic devices, taking into account various technical, feasibility and didactic criteria. The best scores belonged to *writing* and *stitching* exercises. However, after a few trials, it was concluded that it was not feasible with this hardware. Stitching could not be done because the device did not permit it and the development of a virtual device would have been risky and time-consuming in comparison to other options. Therefore, the second wave of exercises, listed in table 1 was studied with the same criteria. These exercises were more feasible and had simpler motions. We concluded that the two most suitable exercises were "pushing an object a certain distance" and "pulling a spring a certain distance". They were respectively called PUSH and PULL. Their characteristics are summed up in table 2 and the corresponding installation is visible in Fig. 2.

Table 1. Exercise proposals and criteria, second wave

	PUSH	**PULL**
Description	Pushing a block up a slope up to a specific line	Pulling a spring in a designated direction with a designated force (1N)
Objective metrics	Distance between the final block location and the target location (mm)	Final force vector magnitude and angle
Time required per individual motion	10s	10s
Installation	Block on an inclined plane.	A spring meter fixed on an horizontal plane
Target value	X cm	1.8 N

Table 2. Description of the motions

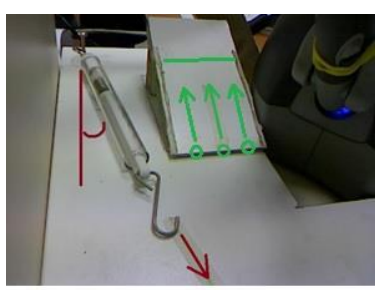

Fig. 2. Push (on the ramp, on the ramp) and Pull (with the spring meter, on the left) exercises

These exercises were chosen as they do not require rare experts.We used an "artificial trainer" who was this person previously trained on the exercises and installed in an advantageous position concerning the trainee. The trainer with a plain view of the working area with clear goal indicators had greater instantaneous knowledge and could easily perform correct and precise gestures. In the case of the PULL exercice, the indicator was the force scale indicated on the side of the spring scale. In the case of the PUSH exercise, the indicator was a drawn line that indicated the final position.

Although the participants could see these two markers on screen, the optical distortion of the webcam made these markers unreliable and ultimately less valuable than the trainer's instructions. For instance, the angle between the camera lens and the mass-spring was small acute and this distorted the view. In the case of the block, as the objected passed a certain height in the ramp the object itself blocked the view of the reference line. The participants did not have clear goal indicators and thus had to rely on haptic guidance the trainer provided they were obliged to use their kinaesthetic abilities.

3.5 Setup

The setup is composed of three Geomagic 3D Touch haptic devices (see Figures 3 and 4). The devices' kinematic and dynamic parameters are available in [9]. These devices are six d.o.f. systems but only three d.o.f. are actuated. The control software was implemented in Matlab Simulink. Concerning the software connection with the devices, the Open Haptics software library was used along with the Phantorque block [1] and extended to simultaneously work with three devices.

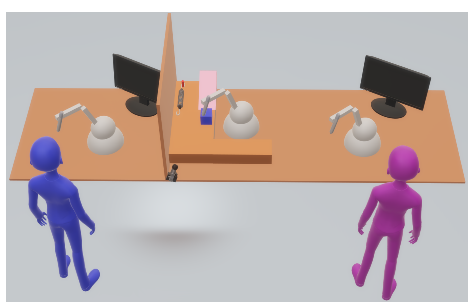

Fig. 3. The global setup organization

The slave device is located in-between the trainee's device and the trainer's device. A wall separates the participants from the slave, blocking visibility. On the other side of the wall, the teacher has a full view. A camera is located on the table 5cm away from the platform. Another one was added to surveillance the entire procedure. The camera has an overview of what was happening during the experiment (see Fig. 5) and records a separate video for the development each participant undertook during the experiment. The videos are references for outliers in the data collected.

Fig. 4. Experiment set up (in laboratory)

Fig. 5. Distorted view the participants had throughout the experiment

3.6 User Interface

As explained in the previous scenario, the *following* user needs to position his/her tool at the exact position as the leaders'. To help him/her, we developed a specific software to display an intuitive help on screen. As the control of the whole dual-user system is performed by Simulink in real-time, three alternatives were considered:

- generating C code from the Simulink control model, and enriching this code with the desired HMI functions;
- manually coding the control law in C++ accompanied by the desired HMI functions;
- developing a communication module between Simulink and a standalone HMI program.

The first option required the Simulink Coder software which we did not have. The second option required too much time to write in C++ a stable real-time controller running the ESC model. These two options also required to manage the communication with the three haptic devices, through Chaid 3D API[1] for instance. Their complexity and their high risk of incompatibility during the integration of each part convinced us to choose the third option. For this option, three ways to communicate between Simulink and a standalone HMI home-made software were considered:

– a TCP/IP local connection (both software running on the same computer);
– *SendMessage* mechanisms, which provide functions to communicate between Microsoft Windows applications;
– a Memory Mapped File, which is a fake file, located in RAM and shared by two Windows applications.

The "Memory Mapped File" (MMF) solution was chosen, as it provided the best performance with a low implementation complexity. As Simulink does not provide Memory Mapped File blocks, a MEX Function was programmed in C++ to send data vectors out of the Simulink model in real-time. This function retrieves these data and stores them in the MMF utilizing the Boost library[2]. The HMI software, written with Visual Basic .Net, retrieve these data from the MMF and display them on screen. It featured a 3D and a camera view to help users interact and perform their tasks. Two spherical objects were displayed in the view, corresponding each one to the tip of the tools of the trainer and the trainee. Initially, we also displayed the position of the slave robot but it disturbed the users. Note that the 3D world displayed on screen is oriented so that the motions of the tip of both user devices coincided with the directions on screen. It was written using HelixToolKit[3].

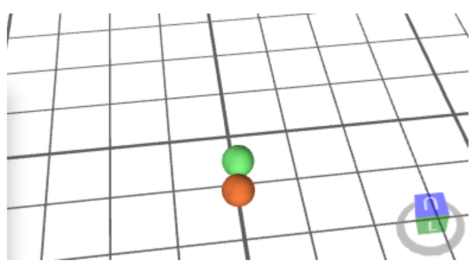

Fig. 6. The HMI software 3D view

4 Preliminary Results

18 participants, divided into two groups, were invited:

[1] See https://www.chai3d.org/
[2] See https://www.boost.org/
[3] See http://helix-toolkit.github.io/

- Group 1: 8 participants successively used the system in the demonstration, **guidance** (use of AAA) and evaluation modes.
  For each exercise, the participants had 3 demonstrations, 3 attempts with guidance and 3 evaluations.
- Group 2: 8 participants tested demonstration and evaluation modes.
  For each exercise, the participants had 3 demonstrations and 6 evaluations. The first 3 evaluations were not analyzed and were only there to remove any advantage group-1 participants might have had due to the higher number of attempts (the three guided attempts).

Finally, since the two groups had the same exercises, practiced the same number of times and had the same haptic interfaces, the only difference between the two groups comes down to the haptic guidance (guidance mode). Thus, their improvement (or lack thereof) can be attributed to the help brought by the training system.

The evaluation took approximately 15 minutes per participant. Each motion took approximately 1 minute. In total, the number of attempted motions was that of 9 minutes. The survey took 5 minutes, which amounts to a total of 14 minutes. We recorded the evolution of the positions and the actuation forces for the Master 1, the master 2 and the slave, and α. For the PUSH exercise, the angle between the effective force vector and the reference force vector was calculated. Participants 10 and 16 were eliminated (inconsistent angle values), only the angle between the mean force vector of Group 1 and the reference vector outperformed the Group 2.

5 Discussion

We faced a few issues. Initially, the pens of the Phantom devices were not fixed sufficiently solidly and precisely, so that the position and force measures provided by the devices were not accurate enough. To counter this, a band was later placed around each pen so that all pens were fixed in the same position. Also, whenever any of the masters overpassed the 3N limit, the overshoot affected the calibration between the slave and the masters. The guidance mode could be interpreted as a disagreement between two masters and this might have caused calibration problems and may account for Group 1's poor performance.

In terms of objective measures, the precision in the performance of the exercises obtained by the participants of group 1 was nearly the same as for group 2. With the error tolerance, it was not possible to differentiate the users of group 1 who were guided during their training, compared to the others. These results lead to several conclusions: the system is not efficient in hands-on training, the exercises are not sufficiently probative, the power range (3N) of the haptic device is not sufficient to get valuable data (and generates disturbing calibration issues), and, in any case, the number of participants was too low. Concerning subjective feedback, it was noted that some participants were shy to grab the haptic device. These participants were nervous and might have performed below their abilities. On the other hand, some participants with previous experience (participant 14 for instance) may prove to have significantly higher skills than other participants. To go around this, we propose a warm-up exercise.

For future experiments, we recommend the trainer subjectively assessing the gestures performed by the trainees on each exercise completion, through participants' performance grades. This should permit to tune objective assessment according to a subjective one, performed by an expert on a first panel of participants. We then test it on another panel of users which level is known in advance to determine with which level of precision, the system can situate trainees in their learning curve.

6 Conclusion

This paper introduced preliminary works on building an experimental end-user evaluation for dual-user haptic systems for hands-on training. Such systems bring together the advantages of haptic computer-based training systems and those of supervised training where an expert trainer actively helps in the learning process. It was tested with the ESC architecture. The first results were not convincing but they mainly permitted to highlight several technical and organizational issues to overcome in a close future. The objective of this project is to test other architectures such as those listed in the state-of-the-art section, to provide comparative conclusions about the pros and cons of each one.

Acknowledgments

The authors acknowledge the financial support of the China Scholarship Council (CSC) and the Consejo Nacional de Ciencia y Tecnologia (CONACyT) in Mexico.

References

[1] C. I. Aldana, E. Nuño, L. Basañez, and E. Romero. Operational space consensus of multiple heterogeneous robots without velocity measurements. *Journal of the Franklin Institute*, 351(3):1517–1539, 2014.

[2] S. Delorme, D. Laroche, R. DiRaddo, and R. F. Del Maestro. Neurotouch: A physics-based virtual simulator for cranial microneurosurgery training. *Neurosurgery*, 71:32–42, Sep. 2012.

[3] A. Ghorbanian, S. Rezaei, A. Khoogar, M. Zareinejad, and K. Baghestan. A novel control framework for nonlinear time-delayed dual-master/single-slave teleoperation. *ISA Transactions*, 52(2):268–277, Mar. 2013.

[4] B. Khademian and K. Hashtrudi-Zaad. Shared control architectures for haptic training: performance and coupled stability analysis. *The International Journal of Robotics Research*, 30(13):1627–1642, Mar. 2011.

[5] F. Liu, A. Lelevé, D. Eberard, and T. Redarce. A dual-user teleoperation system with online authority adjustment for haptic training. In *Proceedings of the 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC15)*, Milano, Italy, Aug. 2015.

[6] F. Liu, A. Lelevé, D. Eberard, and T. Redarce. An energy based approach for passive dual-user haptic training systems. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS 2016)*, Daejeon, South Corea, Oct. 2016.

[7] F. Liu, A. R. Licona, A. Lelevé, D. Ebérard, M. T. Pham, and T. Redarce. An energy-based approach for n-d.o.f. passive dual-user haptic training systems. *Robotica*, pages 1–21, Revision majeur 2019.

[8] S. Nudehi, R. Mukherjee, and M. Ghodoussi. A shared-control approach to haptic interface design for minimally invasive telesurgical training. *IEEE Transactions on Control Systems Technology*, 13(4):588–592, Jul. 2005.

[9] T. Sansanayuth, I. Nilkhamhang, and K. Tungpimolrat. Teleoperation with inverse dynamics control for phantom omni haptic device. In *Proceedings of SICE Annual Conference (SICE'12)*, pages 2121–2126, Aug. 2012.

[10] M. Shahbazi, H. Talebi, S. Atashzar, F. Towhidkhah, R. Patel, and S. Shojaei. A new set of desired objectives for dual-user systems in the presence of unknown communication delay. In *Advanced Intelligent Mechatronics (AIM), 2011 IEEE/ASME International Conference on*, pages 146–151, July 2011.

[11] K. Shamaei, L. Kim, and A. Okamura. Design and evaluation of a trilateral shared-control architecture for teleoperated training robots. In *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*, pages 4887–4893, Aug 2015.

[12] S. Stramigioli. Modeling and IPC control of interactive mechanical systems: A coordinate-free approach. *Springer*, 2001.

[13] A. Talasaz, A. L. Trejos, and R. V. Patel. The role of direct and visual force feedback in suturing using a 7-dof dual-arm teleoperated system. *IEEE Transactions on Haptics*, 10(2):276–287, April 2017.

[14] N. Vaughan, V. N. Dubey, M. Y. Wee, and R. Isaacs. A review of epidural simulators: Where are we today? *Medical Engineering & Physics*, 35(9):1235–1250, Sept 2013.

[15] E. Yiannakopoulou, N. Nikiteas, D. Perrea, and C. Tsigris. Virtual reality simulators and training in laparoscopic surgery. *International Journal of Surgery*, 13(9):60–64, Feb. 2014.