

# Portable Haptic Interface with Omni-Directional Movement and Force Capability

Carlo Alberto Avizzano, *Member, IEEE*, Massimo Satler, and Emanuele Ruffaldi, *Member, IEEE*

**Abstract**—We describe the design of a new mobile haptic interface that employs wheels for force rendering. The interface, consisting of an omni-directional Killough type platform, provides 2DOF force feedback with different control modalities. The system autonomously performs sensor fusion for localization and force rendering. This paper explains the relevant choices concerning the functional aspects, the control design, the mechanical and electronic solution. Experimental results for force feedback characterization are reported.

**Index Terms**—Haptic interfaces, force feedback, rehabilitation robotics, mobile robots

## 1 INTRODUCTION

IN recent years haptic devices have received a lot of attention. Possible applications for such systems are the field of medical training, tele-manipulation and tele-surgery, micro-manipulation, rehabilitation and virtual prototyping. These applications require a high demand for large operational workspaces while existing solutions are affected by limited workspaces and reduced or non-existent device portability.

In rehabilitation robotics, for example, devices generally consist of a cumbersome and heavy apparatus that is placed in well structured places (hospitals in general) [1], [2]. While these devices have resulted in a breakthrough in the use of haptic devices as a means of robotic aided rehabilitation, the intrinsic structure of these systems requires the presence of a firm ground linkage, which is simultaneously used to release the contact forces onto the ground, and to achieve an appropriate and accurate localization of the contact point position. The positional accuracy is essential in the haptic control loop. Alternative solutions to solve the problem of force grounding and contact localization have been provided by two different technical approaches: wearable exoskeleton devices [3] or mobile supported haptic devices [4].

Wearable devices with high kinesthetic feedback come with motor and mechanical structures whose weight, even when reduced with special materials, is far too heavy for prolonged usage such as in rehabilitation protocols.

The Mobile Haptic Interfaces (MHI) approach [4], [5] involves a decoupled system consisting of a traditional haptic interface on top of a supporting mobile platform. This solution entails high costs in terms of encumbrance, robustness and complexity. The mobile base is generally a heavy

platform with slow dynamics and tracking delays may be present when attempting to follow human motion. These conditions prevent the overall system from following the user's arm gestures reliably, and, as a consequence, the operator may feel the boundary of the haptic device workspace, which, in turn, can create spurious forces and ultimately cause the loss of transparency [6]. Finally, MHIs generally have a very high weight compared to the force feedback provided [4].

In the present work, we propose an alternative approach to haptic rendering that is realized on a new type of mobile haptic interface. The goal of this approach is the direct use of the robot locomotion interface (the wheels) for the generation of haptic stimuli. The approach poses new challenging problems in the device's design and control, among these: since the forces are generated through friction with the ground, how is it possible to deliver proper stimuli on the user's hand? How is it possible to make the force rendering consistent even in presence of relevant inertial effects and wheels slips? Since the encoders' information in the presence of wheel slips does not produce a good estimation of the device position, how is it possible to recover the robot localization with the minimum effort in terms of sensor payload and environment characterization? Therefore, the performances of the haptic response and the device localization were designed having in mind the sensitivity and the accuracy of the human hand. In addition, given the purpose of the device, we reconsidered the localization and the haptic rendering algorithms as a whole. The result is an integrated sensor fusion and motor control algorithm that simultaneously provides three features:

- high performance/sensitivity of the haptic rendering;
- accurate localization;
- drift estimation and compensation.

The internal structure of the algorithm mixes components of sensor fusion, optimal control and admittance controller. These components have been jointly designed so that the localization algorithm and the motion drift compensation run in the background to the haptic rendering without producing noticeable effects to the user.

- The authors are with the Perceptual Robotics Laboratory (PERCRO), Scuola Superiore Sant'Anna, Pisa, Italy.  
E-mail: {c.avizzano, massimo.satler, e.ruffaldi}@sssup.it.

Manuscript received 20 Feb. 2013; revised 8 Jan. 2014; accepted 2 Mar. 2014.  
Date of publication 10 Mar. 2014; date of current version 18 June 2014.

Recommended for acceptance by A. Frisoli, M. O'Malley, D. Campolo, and K. Sienko.

For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.

Digital Object Identifier no. 10.1109/TOH.2014.2310462

This paper presents the design of the portable haptic device, together with experimental tests to quantify the performance in terms of quality of force feedback and slippage. The paper describes the adopted solution and the control algorithm of such a novel link-less haptic device. How major specification issues have been devised, and how major issues to control and properly coordinate the force feedback in the space will also be described.

The remainder of this paper is organized as follows. Section 2 introduces the design objectives and explains the requirements around which the device has been developed. After that, the interface, the system architecture and how the user interacts with it are described in detail. In Section 3 the control aspects concerning the device are addressed. The employed localization method, the control architecture and the force rendering algorithm are explained. Section 4 reports a series of experiments characterizing the interface in terms of slippage issues and amount of force feedback. In addition some results of preliminary trials are also reported. The paper ends with the conclusions and future work.

## 2 INTERFACE DESIGN

In the recent past, several researchers have shown that important variables for neuroplasticity of the motor cortex and in re-learning motor skills after stroke are the quantity, duration, content, and intensity of training sessions. Unfortunately, when traditional therapy is provided in a hospital or rehabilitation center, the patient is usually seen for few hours a week. Robot-mediated therapies, that increase the amount of rehabilitation practice and the quality of medium term rehabilitation, could improve this situation [7]. However, even if interesting results have been achieved, the use of robot-mediated therapy has not become very common in clinical practice. Two relevant reasons are the cost and the complexity of these devices, which make them difficult to be purchased and used in all the clinical centers.

Nowadays most robotic rehabilitation devices employed in hospitals, require permanent or semi-permanent setups of several environmental conditions [8], [9]. In addition, such systems are not designed to be autonomously used, i.e., they require trained staff to initialize the interface, to position the machine with respect to the patient's limb and finally to start the training session. Thus, the use of such tools as instruments for long self-rehabilitation exercises is very difficult.

Recently there has been an increase of interest for portable and simple solutions that can control the therapy intensity and reduce the devices' costs [10], [11], [12]. However, most of the solutions consist of passive devices, which provide only resistance to the patient's movements. Even though passive platforms are low-cost and potentially effective in motivating patients and providing quantitative measures of patient progress, (1) they cannot help patients to complete the task, (2) they cannot provide force-feedback to correct unwanted movements, (3) and they do not allow for real-time adjustment of the therapy's intensity. With respect to the already existing devices, we propose an active system that can help patients performing rehabilitation protocol exercises.

Indeed the interface presented here, MOTORE (MOBILE roboT for upper limb neurOrtho REhabilitation) [13], aims to extend the usability of robot-mediated therapy providing a low-cost and easy to use system, that can be employed even at home. The embedded control system modulates the way the robot reacts to external perturbations such that the interface can move, guide or perturb the movement of subjects' upper limb and can record motions and physical quantities such as position, velocity, and applied forces.

### 2.1 Requirements

MOTORE introduces novel design concepts for force rendering, locomotion, localization and power supply.

The device has been developed around the key requirement of system portability. To extend the amount of rehabilitation time available for the patient, the device should be portable, able to work in domestic environments and to be used without the need for specialized personnel.

The first requirement led toward a compact lightweight device with reduced encumbrance on the working plane. Second, the interface should be simple in terms of its mechanical structure. Compact and robust mechatronic devices able to be operated in few minutes effortlessly are preferable. Third, to assure the full portability of the system, it has to be completely autonomous both for the computing aspects and for the sensing elements. The adoption of an embedded control unit and on-board sensors is required. Fourth, the requirement for portability should not result in scaling down system performance, i.e., the system size, the workspace, the amount of exerted force and velocity, should be consistent with the human use requirements, and with the typical values of force and velocity of rehabilitation tasks.

In upper limb robotic rehabilitation the MIT-MANUS [1] is currently being employed in most of the protocols and it has been proved effective for neuro-rehabilitation [14]. The specifications of MIT-MANUS, in terms of its maximum velocity and maximum force, have provided to MOTORE a relevant basis to define rehabilitation protocol requirements.

Concerning the usability of the system, it is expected that to work with the device the user should only switch-on the power button. Hence, the sensing system should be able to work with a reduced or no calibration procedures (see Section 3.2 for more details).

To assure the portability, the motion of the device should be self powered through batteries placed within its case, with an operational autonomy of at least 70 minutes, which is close to the duration of rehabilitation tasks as suggested by rehabilitation experts.

The overall weight of the device, including the batteries, should be less than 10 kg for portability reasons. The device's reflected inertia should be actively compensated using the internal motors in order to decrease patient's fatigue and to simulate different types of resistances. Such control algorithm should also be able to guarantee optimal position tracking as well as very precise haptic feedback. The absolute posture (position and orientation) of the platform is determined by the type of exercises that generally

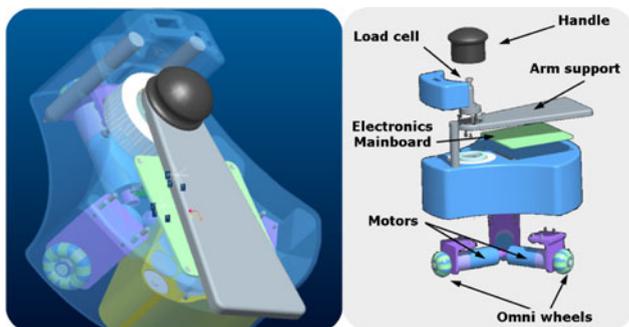


Fig. 1. MOTORE CAD design presenting the exploded representation of the system components.

are involved in the rehabilitation tasks, such as path following or pick and place operations, whereas the haptic feedback is required to provide accurate and useful guidance to the patient. In addition, the platform should have omni-directional kinematics so that the direction of motion and the feedback force direction can be changed quickly and without extensive maneuvering.

From the patient's point of view, the system should appear as user friendly as possible, i.e., it should be natural and easy to interact with. Two elements have to be considered: a proper support for the user's arm and a handle that can easily be grasped by impaired people. The system should provide force feedback only when the user grasps the handle. Moreover, a force sensor is required to measure the interaction forces between the robot's handle and the user's hand during the whole rehabilitation exercise. For a better assessment, the activities carried out at home should be monitored and recorded for future analyses and monitoring progress. The system therefore should include an internal memory to store the suitable variables describing the user's performance during rehabilitation training.

## 2.2 System Architecture

Fig. 1 shows the mechanical design of the interface. The proposed Portable Haptic Interface [15] consists in a omni-directional mobile cart actuated by three omni-wheels, which are driven by three independent DC-motors having a maximum power of 90 W and a maximum continuous torque of 110 mNm. A reduction gearbox was introduced to match the force requirements at the user's handle. A maximum operational force of 35 N has been considered sufficient for most rehabilitation exercises (in comparison the MIT-MANUS is reported to reach 45 N). Each of these three actuation groups achieves a maximum traction force of 42 N, [15]. An additional reduction ratio of  $\cos(\pi/6)$  between the wheel forces ( $[F_1, F_2, F_3]$ ) and the corresponding force on the geometrical center of the robot ( $[F_x, F_y, T]$ ) is introduced by the device's Jacobian. This results in a nominal maximum force of  $F_{x,y} = 36$  N and a nominal maximum torque of  $T = 18$  Nm. Given the actuation group design, the force/torque is limited by the weight of the device and the friction coefficient of the moving wheels. Finally, each motor is sensorized with an incremental optical encoder (512 CPR), which provides a high tracking accuracy of the wheels' movements (20  $\mu\text{m}/\text{count}$ ).

TABLE 1  
Major System Parameters and Variables

Symbol	Value	Unit	Name
M	10	kg	Mass
$I_z$	0.1086825	$\text{kg m}^2$	Inertia
B	25.4	mm	Wheel radius
L	145	mm	Robot radius
N	676/49	/	Motor reduction gear
CPR	512	/	Motor encoder resolution
${}^B Z_p = [X_p, Y_p]$	[124, -40]	mm	Optical sensor position
$F_1, F_2, F_3$	/	N	Wheel traction forces
$\omega_1, \omega_2, \omega_3$	/	rad/sec	Motor angular velocities
$u, v, r$	/	m/s	Velocities in the mobile frame
$\Sigma = [x_0, y_0, \psi]$	/	m	Device pose in fixed frame

The user handle is placed toward the front of the device in order to provide a comfortable support for the user's forearm and to achieve a better weight distribution. Fig. 3 shows the interface where the position of the arm and hand can be seen. The system is link-less with respect to both the mechanical and electrical parts. The power supply is provided by a battery pack that has been designed to provide the capacity of 3.0 Ah at 12 Volts. This pack powers the motors and all the electronics such as the DSP, the drivers, and the sensors. The operational autonomy of the device is 75 minutes in nominal continuous usage condition, see [15] for estimation details.

The overall MOTORE weight is about 10 Kg including the batteries. However, the user's forearm and hand increase the total weight of the device thereby improving the amount of force that may be transmitted by the wheels through pure friction with the surface.

On board there is a three-axis accelerometer used both for error detection and additional compensations. In particular, at the startup of the device the accelerometer is used to detect if the robot is being operated on a non-horizontal surface, whereas during the working session it is used to detect dangerous conditions. The accelerometer is also used to dynamically compensate for the robot's inertia.

A two-axis force sensor has been embedded in the robot handle to measure the planar interaction forces with the user's hand. The cell gains and the computing unit have been designed to achieve 0.01 N accuracy on the full scale of 40 N. This load cell is the control input provided to the user to drive the system.

Finally, the system embeds an optical sensor (Opt) that, together with a pattern-printed surface provides absolute localization.

The computing unit was carefully chosen in order to ensure low-power consumption but at the same time the required computational power. It consists of the DSP TMS320F28335 commercialized by Texas Instruments (Dallas, USA) running at 150 MHz, with a 32 bits floating point unit. The DSP allows handling the whole control scheme at a base rate up to 10 kHz, [15].

Concerning the communication, a Bluetooth module has been interfaced to the DSP in order to establish a bidirectional communication with remote devices.

Table 1 summarizes the major design information and relevant symbol terminology used in the paper, and Fig. 2 shows a view of the base of the device, with almost all the above mentioned system components.

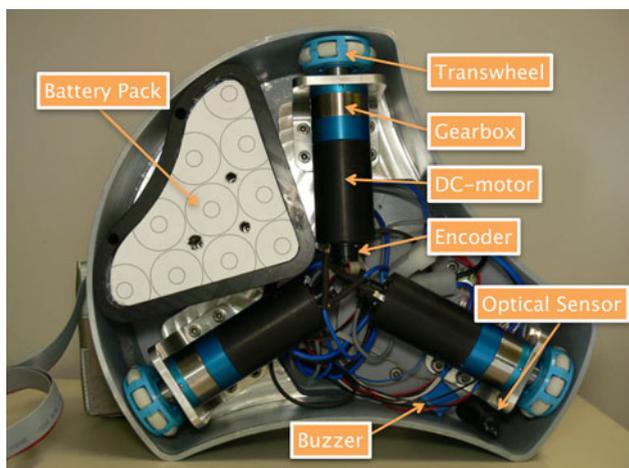


Fig. 2. Bottom view of the MOTORE system highlighting the layout of the components. With respect to the photo, the actual system configuration uses wheels composed by a double row of rollers to avoid hopping during motions.

### 2.3 User Interaction

This section will explain how the system is positioned and how the user interacts with it; moreover it will show the safety features the interface.

The device can simply be placed on a desk and the user sits down on a chair in front of it (see Fig. 3). At this point the user can switch-on the power button, and the robot emits a melody to indicate its state and then the device is ready to start the working session. There is no calibration procedure for the sensors, the working sheet on the desk under the device is used for setup. The sheet is made of a specially printed paper used for absolute localization (see Section 3.2 for details).

The mobile robot works by moving over the working sheet and the user can directly drive it while grasping the robot handle. The handle has been designed to facilitate the grasping by impaired people. To improve the device usability and to provide a natural posture to the user, a passive rotational support (the so called forearm support) has been introduced. The forearm support is constrained to rotate around the vertical handle axis. The bearings inside the support allow a frictionless rotation of the support, which improves user comfort while operating the device and, at the same time, it allows the best maneuverability for the robot.

Fig. 3 shows how the user interacts with the system. The user's upper limb rests upon the forearm support and Velcro straps, hold it in place while the hand exerts forces on the robot handle. Particular care has been taken to minimize the amount of forces exchanged on the motion plane between the forearm support and the user's upper limb. Thus the device feels only the interaction forces, lying in the  $xy$ -plane, between the user's hand and the robot's handle. These forces are the input signals for the admittance type controller, which provides the velocity reference that will be followed by the device by mean of the internal controller. An emergency stop button placed near the other hand provides safe operation. In the actual configuration the support surface is layered over a wooden plane with protruding borders introduced for safety reasons.

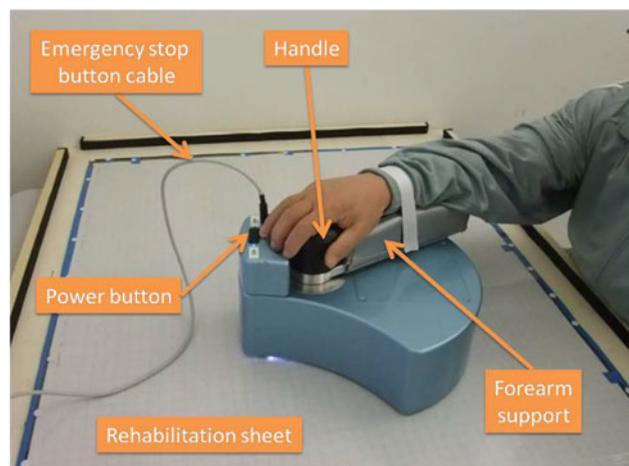


Fig. 3. User interaction with the MOTORE system. The device moves and operates on a transportable panel with given geometry and a printed pattern.

Given the system's symmetry, the user can drive the interface with the right or the left arm. Such a change can be performed at any time without any reconfiguration procedure, even during the working phase when the robot is active.

Since the interface may be used by inexperienced people, particular attention has been directed to safety aspects. An active supervising control and an easy to access emergency stop button, minimize risks to the user. To ensure safety, the device limits the range of motion in absolute coordinates. Beside those tools, other strategies have been implemented. Force feedback is limited by both software saturation and the intrinsic nature of the actuation system: due to the upper limit in the wheel friction coefficient, the robot cannot exert harmful forces to the user's upper limb even in the presence of sudden motions.

## 3 CONTROL

To date, existing MHIs decouple mobility and force generation by making the mobile platform as transparent as possible and delegating the force generation to the haptic interface component [6]. On the other hand, MOTORE adopts a different solution by directly generating forces through the mobile platform's wheels, which are used as a locomotion tool as well as a force rendering tool. This choice introduces several challenges that need to be addressed, such as slippage and localization that will be discussed in the rest of the paper. In addition a tradeoff between position accuracy and force feedback smoothness has to be considered.

This section discusses the control organization of MOTORE starting from the high-level control. Then the localization technique is discussed as a key element of system functionality. The section finishes with a description of the force feedback rendering approach.

### 3.1 Logic Architecture

The control structure is organized as a high-level control (interaction management and safety checks) and low-level control (sensor measurement and motor control) both

managed by the embedded processor. The high-level control is organized as a state machine that supervises the exercise as triggered by commands from the external communication link. The state machine contributes to the flexibility and safety of the system. In particular, after initialization, the system enters the so called "Wait Mode" in which the device waits for inputs from the higher level application. In this state the motors are enabled but no forces are exerted. When a request for command arrives the "Center Mode" state is entered moving the device to the starting point of the planned exercise. When the device reaches the starting position the system enters the so called "Run Mode." In the "Run Mode" state the robot is driven through the force sensor and it generates the force feedback according to task parameters and the chosen low-level control modality. To improve user interaction with the system, each state is clearly marked through a unique melody that is played when entering the state. Besides that, during the "Run Mode" the buzzer plays a beep periodically indicating that the system is under force feedback control.

Error conditions are managed across all the states moving from the current state to the so called "Error Mode". Examples of error conditions are: timeout for waiting commands while in "Wait Mode", thresholds in the maximum velocity of the device and emergency stop button activation. In Error Mode the power electronics disables the motors and the buzzer emits the alarm sound. "Error Mode" is exited back to "Init Mode" when an external reset command is received. Additional validations are performed in the "Init Mode" that is a short timed procedure that verifies the system and environment conditions such as battery level, sensor health status and the horizontal alignment of the working plane.

### 3.2 Localization Technique

The control algorithm requires an absolute and precise localization of the robot posture in the environment, in order to develop position-based exercises. Such a feature is not so straightforward to be achieved with the additional constraint of the absence of calibration. Actually, precise system localization cannot be solved by means of the proprioception method only, i.e., several measuring systems as well as methods for sensory data fusion have to be applied. This section presents the measuring system adopted in MOTORE and how this information is fused with the odometry through a custom designed Extended Kalman Filter that takes into the account the peculiarity of the force rendering requirements and the nature of the sensors.

The use of Kalman filtering is well known to solve the localization problem of mobile robots. Standard and multi-rate, extended and unscented Kalman filters have been proven to achieve increasing levels of accuracy [16], [17]. However, the user interaction, required in haptic force generation, poses additional constraints to the position reconstruction provided by Kalman filtering. Achieving the best interaction here means being able to mediate between the position accuracy reconstruction, achieved by Kalman filtering, and the quality of force feedback.

MOTORE is provided with two position information sources: an absolute localization system based on optical

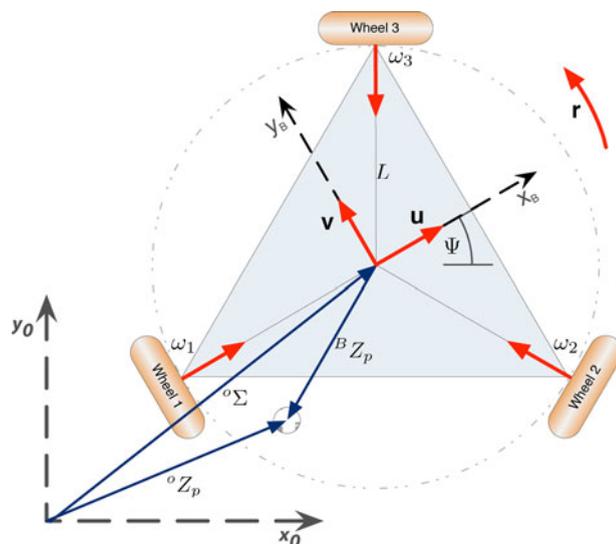


Fig. 4. Kinematic diagram of the system depicting the three wheels, the absolute position tracking device, and the reference systems.

sensors and odometry from motor encoders. The absolute localization has been achieved using a CCD sensor, which operates with a proprietary technology from Anoto (Lund, Sweden) that ensures accurate localization (0.03 mm resolution) over a very large displacement [18]. These sensors are primarily employed for creating digital pens that precisely track and record the writing patterns detecting the position of the optical sensor over a pattern-printed sheet. This special pattern encodes the absolute localization of the sensor and it is completely different from the optical flow approaches employed in modern optical mice.

The optical sensor is placed in a given position inside the MOTORE case,  ${}^B Z_p$  see Fig. 4 and Table 1, and it is oriented along the vertical axis in order to observe, perpendicularly, the sliding sheet. The CCD sensor delivers at 75 Hz the absolute  $xy$ -position measurement relative to the workspace ( ${}^o Z_p$ ). The communication with the embedded control unit is realized by means of the Bluetooth channel. This solution has advantages with respect to camera based or acoustic tracking since it is insensitive to the environmental light conditions, it does not require calibrations and, finally, it is completely embedded into the device.

The proposed fusion technique is almost independent of the absolute tracking technology. Fig. 4 summarizes the conventions used for the mobile device and introduces the adopted reference frames. For the definition of the symbols please refer to Table 1. A fixed reference system "o" is used to model the motion on the sheet of paper, and a local reference system "B" is placed on the device. The fixed reference frame has been defined originating in the center of the working sheet and the  $xy$ -axes are oriented in the transverse plane of the user:  $x$ -axis laying in the intersection with the coronal plane (positive direction toward right), while the  $y$ -axis is at the intersection with the sagittal plane (positive direction toward front).

The system dynamics have been represented by a non-linear discrete-time equation referred to as the process model:

$$x(k) = f(x(k-1), u(k-1)) + w(k-1), \quad (1)$$

where  $x(k)$  is the state at time  $k$ ,  $u(k)$  is the control input, and  $w(k)$  is a zero-mean, white, Gaussian process noise with covariance matrix  $Q(k)$ .  $f(\cdot)$  represents the relation between variables in  $(k-1)$ th step and  $x(k)$ , i.e., it is the state prediction based on the actual state and inputs. The state prediction equation is derived in a discrete form based on the controller sampling time (1 kHz):

$$\begin{bmatrix} x_{o,k} \\ y_{o,k} \\ \psi_k \end{bmatrix} = \begin{bmatrix} x_{o,k-1} \\ y_{o,k-1} \\ \psi_{k-1} \end{bmatrix} + J_{k-1} \left( \begin{bmatrix} \Delta\theta_{1,k-1} \\ \Delta\theta_{2,k-1} \\ \Delta\theta_{3,k-1} \end{bmatrix} + \begin{bmatrix} w_{1,k-1} \\ w_{2,k-1} \\ w_{3,k-1} \end{bmatrix} \right), \quad (2)$$

where according to Table 1,  $[x_0, y_0, \psi]$  is the pose of the mobile frame with respect to the fixed frame and  $J_k$  is the motion Jacobian matrix defined as:

$$J_k = \frac{2B}{3N} \begin{bmatrix} \sin(\psi_k) & \sin(\psi_k + \frac{2\pi}{3}) & \sin(\psi_k - \frac{2\pi}{3}) \\ -\cos(\psi_k) & -\cos(\psi_k + \frac{2\pi}{3}) & -\cos(\psi_k - \frac{2\pi}{3}) \\ \frac{1}{2L} & \frac{1}{2L} & \frac{1}{2L} \end{bmatrix}. \quad (3)$$

The measurement equation is,

$$z(k) = g(x(k), u(k)) + v(k), \quad (4)$$

where  $z(k)$  is the measured value by means of the optical sensor,  $w(k)$  is a zero-mean, white, Gaussian measurement noise with covariance matrix  $R(k)$ , and  $g(\cdot)$  is the relation between the state and the measured value, which is,

$$\begin{bmatrix} z_{1,k} \\ z_{2,k} \end{bmatrix} = \begin{bmatrix} \cos(\psi_k) & -\sin(\psi_k) & x_{o,k} \\ \sin(\psi_k) & \cos(\psi_k) & y_{o,k} \end{bmatrix} \begin{bmatrix} X_p \\ Y_p \\ 1 \end{bmatrix} + \begin{bmatrix} v_{1,k} \\ v_{2,k} \end{bmatrix}. \quad (5)$$

According to Table 1,  $[X_p, Y_p]$  represent the position of the optical sensor in the mobile frame.

The prediction phase is executed at 1 kHz whereas the correction happens at 50 Hz due to the limitations in the communication channel rate that reduces the optical sensor operation frequency (75 Hz). Every time a correction step is applied, the estimated position shows a step behavior, which, due to force feedback generation, like for example a classical impedance control, results in a force impulse on the robot handle. Since the quality of haptic feedback has to be maximized, the standard EKF has been modified in order to apply the estimate correction in a smoother way than applying it as a single pulse. This innovation is applied gradually between two consecutive update steps of the Kalman algorithm.

### 3.3 Force Rendering

The low-level control algorithm has to provide force and position control and it has to satisfy the fail safe criteria. In the following an overview on the low-level control architecture is presented, while a detailed treatment can be found in our previous work [19].

The low-level control algorithm is mainly composed by three closed loops and by two open loops that take into the account relevant compensations: inertia compensator and

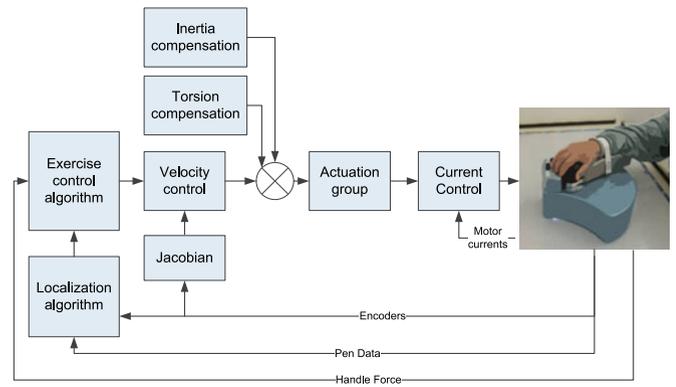


Fig. 5. Block diagram of the control algorithm. The architecture is based on several nested loops and two open loop compensations. Pole separation among these feedback components ensures overall stability. In addition, the localization algorithm updates have been optimized to minimize the force changes when the drift is recovered.

torsion compensator. Fig. 5 shows the organization of the low-level control algorithm and it highlights the relation between all the components involved in the generation of wheel torques. At the inner level a PI current control determines the PWM duty cycle to control the current flowing within the motors. The current regulation is closed on the velocity controller that enables the platform to follow the desired velocities in the local coordinate system. The velocity controller generates the wheel torques to track the desired velocity provided by the “Exercise control algorithm” sub-system. According to the control structure, the device velocities can be accurately estimated using only the proprioception system and, in particular, without requiring the Kalman filter. The choice of this configuration allows us to implement safe behavior in the absence of absolute information due, for example, to large Bluetooth delays. The position update sub-system, introduced in Section 3.2, is the outer loops running at the slowest frequency (50 Hz).

The core of the system is the “Exercise control algorithm,” which decides how to propagate the user input force, measured on the force sensor, to the velocity controller. The module operates in different way according to both the state of the system and the phases of the rehabilitation application. In particular, it provides zero velocity during “Init Mode” and “Error Mode,” free motion during “Wait Mode,” point attraction/potential fields (the device is attracted to a point in space) during “Center Mode,” and constrained admittance tracking during the “Run Mode.” In the latter case the device behaves like a viscous virtual mass and it allows the user to move along the reference path while at the same time it corrects for errors in the direction orthogonal to the path itself. This behavior is achieved by defining the local unit vectors and implementing an admittance control along the tangential direction and an impedance control along the orthogonal one. The desired velocity ( $\vec{V}_d$ ) resulting from the two contributions is given as follow:

$$\vec{V}_d = \vec{V}_{\parallel} + \vec{V}_{\perp} = Y(\vec{F}_{\parallel} + (\vec{F}_{\perp} - \vec{d}_{\perp}K)), \quad (6)$$

where  $\vec{V}_{\parallel}$  and  $\vec{V}_{\perp}$  are the desired velocity components parallel and orthogonal to the path, respectively.  $Y$  represents

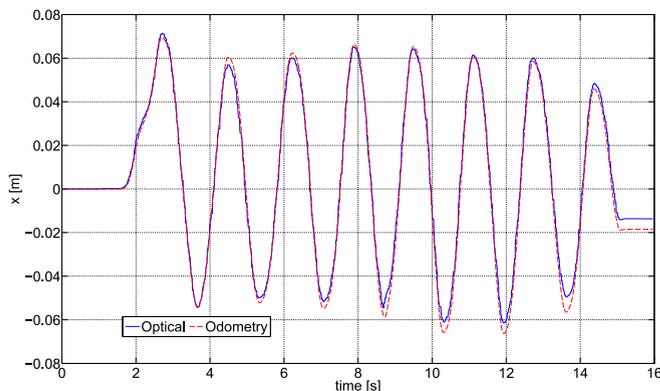


Fig. 6. Position estimation comparison along the  $x$ -axis between odometry and the optical sensor. Slips integrate over time and produce a drift, which is compensated by the sensor fusion algorithm.

the admittance gain filter,  $K$  the stiffness of the impedance filter,  $\vec{d}_\perp$  is the distant vector from the path,  $\vec{F}_\parallel$  and  $\vec{F}_\perp$  are the external force decomposition according to the parallel and orthogonal direction, respectively. This control law allows us to achieve complementary device behavior along the orthogonal and the parallel directions of the path. The device behaves like a pure mass along the direction parallel to the reference path and like a constrained object along the orthogonal one. To better understand the control law, consider the following example where the user pushes on the robot's handle until an equilibrium point is reached. Moreover consider that the user is pushing only along the orthogonal direction of the programmed constraint (a straight line for the sake of simplicity). Thus, with respect to eq. (6):  $\vec{F}_\parallel = 0$  and  $\vec{F}_\perp = \vec{d}_\perp K \Rightarrow \vec{V}_d = 0$ . The interaction force with the user is exactly equal to the impedance effort provided by the control law.

Given the modularity of the designed control algorithm, the system is easily re-configurable for applications in other fields. The base control components (current control, velocity control and Kalman Filter) guarantee the system stability whereas changing the "Exercise control algorithm" allows device customization.

## 4 EXPERIMENTAL SETUP

A series of experiments have been carried out in order to characterize the proprioceptive noise and error in the position estimation, the planar forces that the system can provide to the user and to evaluate the general usability of the system in a rehabilitation like exercise.

### 4.1 Slip Characterization

#### 4.1.1 Design

The test aims to estimate the amount of proprioceptive inaccuracy in the estimation of device posture. The odometry of the system is inaccurate due to both the encoder quantization and the slip of the wheels. Several tests have been carried out, each one with different device conditions. In the first subset of experiments a person moved the device through its case while the motors were powered off. After that, in the second subset of experiments, the device moved autonomously in order to follow the

desired path. In the last subset of experiments the device was loaded with 5 kg in order to simulate the presence of the user's arm. Here the device moved autonomously according to the programmed path. Each test explored different speeds of motion along the  $x$ -axis first, and the  $y$ -axis later. The movements were directed forward and backward from the axis itself (see Fig. 6). Thus, the path consisted of linear movements, which were followed with increasing velocities. During these tests, the device applied no force feedback and the inertia compensation was switched off. It has to be pointed out that, given the device kinematics (see Fig. 4) the chosen directions of movement are enough to represent all the situations that could happen in a more complex path. In fact, the movement along the  $x$ -axis involves only two wheels (wheel numbers 2 and 3 in particular) while the last wheel is completely out of gear, whereas the movements along the  $y$ -axis involves all the wheels.

The recorded variables consisted of the encoders' signals and in the absolute position signal. With such variables the integral of absolute movements can be calculated:

$$D_y = \sum |\Delta pos_y|. \quad (7)$$

Here  $D_y$  is the estimation of the position variation between two consecutive sample times as estimated by means of the measuring entity, such as  $D_e$  for the encoder and  $D_p$  for the optical sensor. The slip is estimated by comparing the integral of the absolute movements resulting from the odometry with the one obtained from the absolute position measured from the optical sensor.

#### 4.1.2 Results

Fig. 6 shows how the trajectories look like in the tests. In particular, the figure plots a test where the interface moves along the  $x$ -axis. The dashed path refers to the estimated trajectory from the encoder signal whereas the solid line represents the measured trajectory. The estimated position is very close to the measured one in the initial phase, but after a while the estimation diverges due to the wheel slippage. At the end of the test, the displacement estimated with the odometry is 1.815 m, while measured with the optical sensor is 1.754 m. The relative error results roughly equal to 3.5 percent.

This analysis has been repeated for each kind of experiment previously outlined, both for the  $x$ -axis and the  $y$ -axis. Fig. 7 reports the boxplot of the relative error between the odometry and the optical sensor. The expression of the error  $E_s$ , that is the relative slip, is given as follow:

$$E_s = 100 \frac{|D_e - D_p|}{D_p}. \quad (8)$$

The figure shows the results obtained in the three kinds of experiment. In the first group, called "Passive," the device had no desired current for the motors while the user manually moved it. The same happened in the second condition, called "Open," where the motor's electrical power circuits were physically opened to avoid electrical brake due to current dissipation into the device circuit. In the last group, called "Active," the device control loops were closed

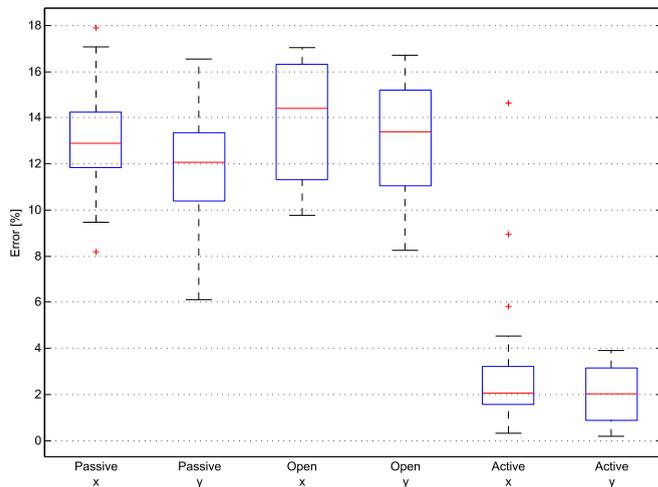


Fig. 7. Relative position estimation error  $E_s$  (eq. (8)) of the odometry against the optical sensor, compared across different actuation group modalities.

and thus the interface was driven through the force sensor using the admittance filter. In these experiments the speed of movement, which depends only on the user, ranges between 10 and 20 cm/s. This figure shows the results for both the  $x$ -axis and the  $y$ -axis. A one-way ANOVA model was applied to compare the three groups (Passive, Open, Active) taking the two coordinates together. The test shows an overall discrimination between the groups' means ( $F = 395.47$ ,  $df = 3$ ,  $p < 0.01$ ). In particular pairwise tests, with Bonferroni correction, show discrimination between the three groups: passive and open are not different ( $F = 5.49$ ,  $p = 0.02$ ,  $95\% CI [-1.9886, 0.0512]$ ) while open and active are distinct ( $F = 373.43$ ,  $p < 0.01$ ,  $95\% CI [9.7612, 12.1952]$ ). The results show that there is no a clear difference between the Passive and Open meaning that the electric brake has no strong influence in the wheel dynamics. In both cases the high value of relative slip  $E_s$ , around a mean value of 13 percent, is mainly given by the rotor inertia and the gearbox, which, in turn, try to keep the steady state of the wheels. On the other hand, when the motors are used to move the device, the slip is drastically reduced. The TOST equivalence test [20] has been applied, for each group, testing the equivalence between  $x$  and  $y$  motion inside the groups. The two axes result equivalent for all groups. In particular we computed the separation following the eq. (7) of [20] obtaining 3.7 for Open/Passive and 3.3 for Active. For the way TOST is formulated it is also possible to find the minimal separation to achieve equivalence: 1.65 for Passive, 2.9 for the Open, and 0.95 for the Active group.

The amount of relative slip  $E_s$  as function of the speed of motion and load is shown in Fig. 8. There are represented four velocity-load conditions where the device moved autonomously at different velocities along the  $x$ - and  $y$ -axis. The labels "v0.2 m0," "v0.3 m0" and "v0.4 m0" represent the cases where the device moved at 20, 30 and 40 cm/s respectively without load on the vertical axis, whereas "v0.2 m5" represents the experiment where the robot moved at 20 cm/s with a weight of 5 kg on its case in order to simulate the presence of the user's arm. As before, the figure shows the results for both the  $x$ -axis and the  $y$ -axis.

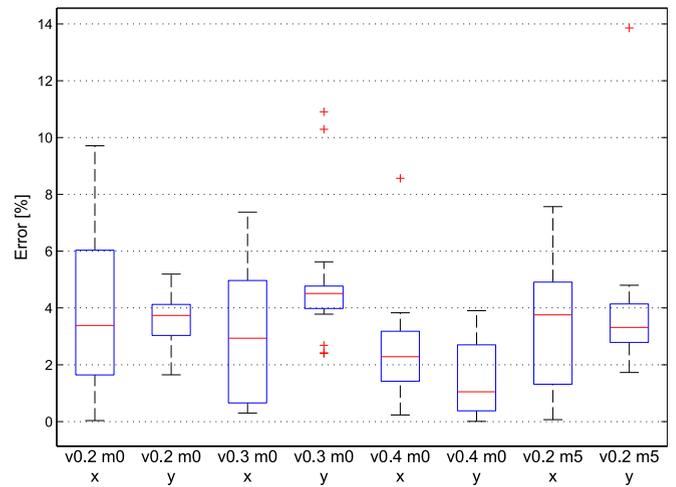


Fig. 8. Relative position estimation error  $E_s$  (eq. (8)) of the odometry against the optical sensor at different speeds of motion and loaded weights.

There is no a significant difference in relative slip  $E_s$  when measured across all the trials ( $F = 3.07$ ,  $df = 3$ ,  $p = 0.0291$ ). Also, there is no significant difference when comparing  $x$  and  $y$  cases inside each velocity-load condition "v. xx mx," except for the group "v0.3 m0."

## 4.2 Force Feedback Characterization

### 4.2.1 Design

This test characterizes the amount of force feedback that the device can provide to the user before slip occurs. The interface is set under velocity control with desired velocity equal to zero, while the user pushes on the robot handle along a specific radial direction in the  $xy$  plane. The external force has to be carefully exerted in order to do not load the robot along its vertical ( $z$ ) axis since this affects the wheel traction forces. Thus, the user pushes with his/her finger the handle from the side instead of grasping it as in the typical working conditions. Moreover, to simulate the presence of the user's arm, a 5 kg mass weight is placed over the device case.

The recorded variables consist in the absolute position signal and in the force sensor signal. The estimation of the static force that the interface can provide is computed by looking for the first force value that triggers the slip event. Thus, it is required that the user gradually increases the exerted force along a specific radial direction until the interface starts to drift. This amount of force is the maximum static force that the robot can provide in that direction. This process is then repeated at regular angular steps of around  $\pi/4$  in order to cover the entire circle.

### 4.2.2 Results

Fig. 9 shows the experimental results in a polar coordinate plot overlapped on the device diagram. Each arrow represents the force that triggers the drift of the device. The length of the arrow represents the amount of force whereas the spatial direction points to the direction where the user "mainly"<sup>1</sup> exerted the forces. The forces are expressed in

1. The direction of the applied external force has been calculated through the mean of unit vectors sampled each sample time.

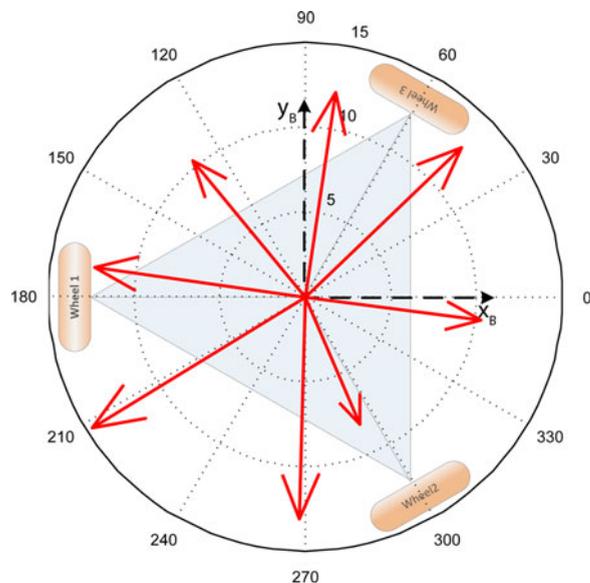


Fig. 9. Maximum force before slipping when the device is set in static condition. The force is expressed in mobile frame.

Newton with respect to the mobile frame of the device. For instance, the arrow oriented toward 270 degree represents the external force exerted along the negative  $y_B$ -axis of the mobile frame. Experimentally it has been found that the maximum amount of static force that the device can provide is approximately equal to 15 N. Beyond such force level the device is still able to provide high value of force feedback but its wheels begin to slip. The position threshold after which the device has been considered losing the static condition was set to 5 mm.

The result is quite different from the theoretical expected value of 36 N (see Section 2.2) due to two main reasons: low static friction coefficient and displacement of the center of mass from the device geometrical center. The plastic cover, employed to protect the pattern-printed sheet from being early damaged, significantly reduces the friction coefficient between the wheels and the working plane. Moreover, the center of mass is placed around 15 mm on the  $x_B$ -axis and due to this displacement two wheels (“wheel2” and “wheel3” in Fig. 9) are more loaded than the last one (“wheel1” in Fig. 9). Reducing the vertical load on “wheel1” decrease also the traction force that it can provide. Furthermore, the effect of the unbalanced wheels conditions depends on the external force direction. In Fig. 9 it can be seen that better results have been obtained exerting forces that increase the load on the “wheel1” (i.e., along the negative direction of  $x_B$ -axis) and coherently worst results in the opposite situation.

### 4.3 Preliminary Evaluation

#### 4.3.1 Design

The objective of this test is to verify the interface usability and the whole system effectiveness as possible rehabilitation tool. It is also important to test the stability of the control system and the correct integration of all the algorithms that compose the whole system. For example, the force rendering algorithm has to cope with both the position estimation provided by the Kalman filter and

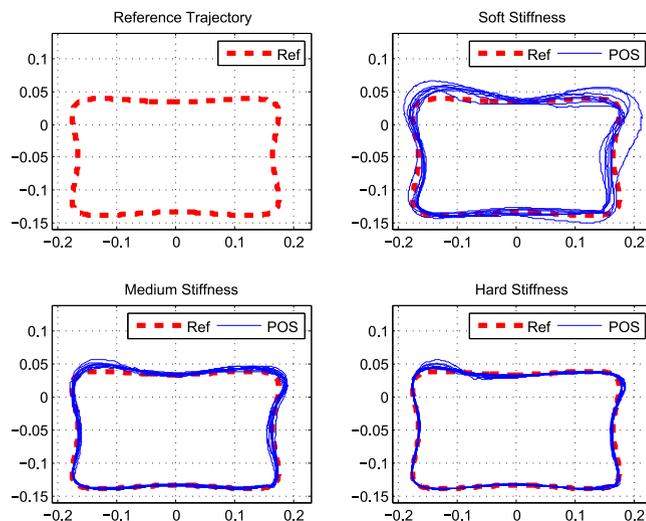


Fig. 10. Trajectory tracking performed by the healthy user.

the velocity controller. Again, the efficacy of the logic control piloting the interface during the rehabilitation session and/or in the presence of possible alarms has to be accurately verified.

In this basic evaluation test to the users were asked to follow the trajectories shown on the screen in front of them. MOTORE is placed on the working plane between the user and the monitor, which, among other statistics, displays in real-time the device position as well as the reference path through a custom GUI. The graphical interface can also be used to manage the rehabilitation session and to store the results. The working plane where the robot can move has been fixed to  $1,080 \times 720$  mm.

The instructions given to the users were to follow the reference path shown on the screen trying to be as accurate as possible. In these preliminary evaluation trails there were no specific constrains concerning the session duration, the number of turns around the path or the speed at which the exercise has to be performed. The main aim was to tests the system effectiveness driving the subject’s arm along the reference path. The system driving capability has been investigated under different driving stiffness levels. The evaluation has been performed comparing the results, in terms of tracking error, obtained from a healthy user with the ones obtained from a patient who signed a consent form. The recorded variables are the device position, the position error from the reference path, the contact force between the robot and the user, the exercise duration, the exercise parameters like for instance the level of stiffness and other signals useful for debugging purpose and/or graphical representation.

#### 4.3.2 Results

Figs. 10 and 11 show the device’s position recorded during the path following exercise of a rectangular path. Fig. 10 displays the data from a healthy subject (right handed, 29 years old) whereas Fig. 11 are the measurements from a patient (hemiparetic, chronic, aged 59 years old) that accepted to participate to initial testing of the device. In both figures the reference path (dashed red line) is shown

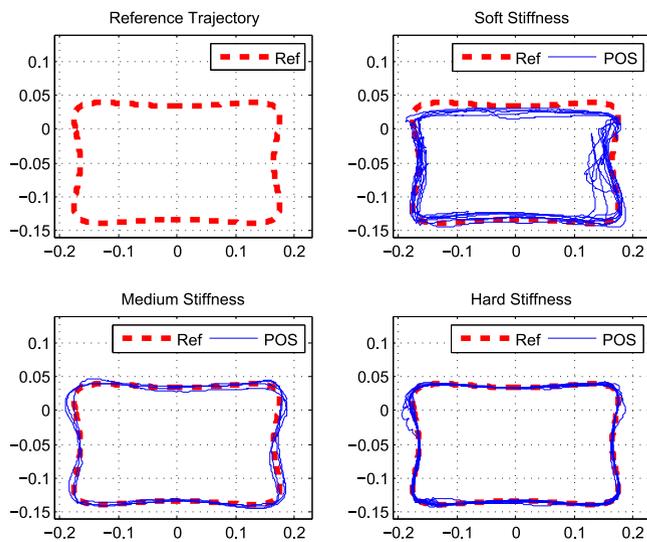


Fig. 11. Trajectory tracking performed by the patient.

in the top left corner, while the other plots show the performed trajectories (solid blue lines) overlapped on the reference path. The position is expressed in meters with reference to the absolute frame. Starting from the top right corner and going clockwise the trajectories displayed were performed with a “soft” device operation (stiffness value equal to 100 N/m), a “hard” device intervention (stiffness value equal to 1,000 N/m) and a “medium” one (stiffness value equal to 500 N/m), respectively. The results show the excellent repeatability of the user’s path (especially with increasing stiffness), even by the patient that did not have complete dexterity over all the task workspace. The gap between the actual path and the reference one increases with decreasing robotic assistance. It can be seen that the path does not trace perfectly the reference. In particular, in the “soft” case, both the users had worse performance compared to the other cases. The patient had difficulties reaching the most distant points from his/her body (the most distant points along the  $y$ -axis) and without a considerable help from the robot he/she moved far away from the reference path. On the other hand, the healthy user, having less amount of constraint, was not able to follow the reference with the same accuracy he/she had in the other conditions. Increasing the amount of robot aid, the paths became more regular and they closely resemble each other. The recorded trajectories also indicate the good position estimation of the robot.

The comparison, in terms of position error from the reference path, between the healthy user and the patient, is shown in Fig. 12. The evaluation has been performed for each stiffness level of Figs. 10 and 11. The error has been evaluated as the absolute value of the minimal distance from the reference path of each point in the space.

It can be seen that with increasing driving stiffness the error decreases for both users and the most important result coming from the analysis is the comparable performance between them. It can be seen that the median value is similar in each case and the variation is reduced with increasing stiffness. These results suggest the MOTORE system is a promising rehabilitation tool.

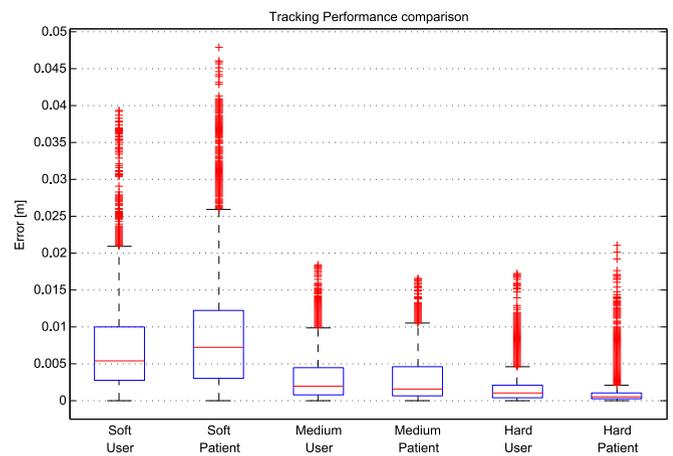


Fig. 12. Tracking error comparison for each stiffness condition between the healthy user and the patient.

## 5 CONCLUSIONS

The work deals with the concepts and the design of the prototype called MOTORE. The device paves the way for new type of interfaces that combine the principles of mobile robotics with the requirements of haptic interaction. The work describes the series of design decisions required to create a portable and effective solution.

The device is a general haptic interface that has been oriented toward robotics rehabilitation in three ways. First, during the design phase particular attention has been paid to adopt standard electronic and mechanical components in order to reduce final cost. Second, the type of device and its control have been aimed at improving patient safety during use. Finally simplified setup procedure were established by removing calibration requirements. With respect to the state of the art, the system represents a breaking point not only in rehabilitation robotics but also in those applications where portable, lightweight and user-friendly device are the primary requirements. Moreover, with respect to the usual devices employed in the rehabilitation therapy, MOTORE performance is not affected by the handle position and orientation. This to deploy easily large workspace span exercise, differently from grounded haptic platforms.

The compact and lightweight characteristics of MOTORE allow it to be used in whichever horizontal plane. Equally noteworthy is the new opportunity for patients to take the device with them and exercise on their own time. Since all exercise’s data are recorded, a therapist could simply download them and analyzes remotely the effectiveness and the therapy progress, without requiring the patient to revisit the clinical center every time. Such a situation could improve the domestic rehabilitation since generally it has not a direct medical control. Finally, new rehabilitation modality like tele-rehabilitation or therapist tele-supervised rehabilitation begins to be attractive.

## ACKNOWLEDGMENTS

This work has been funded by Regione Toscana in the context of the POR-CREO project “MOTORE.” The authors

wish to thank Humanware s.r.l. for the mechanical design and Robotech s.r.l. for the electronics.

## REFERENCES

- [1] N. Hogan, H. Krebs, J. Charnnarong, P. Srikrishna, and A. Sharon, "MIT-MANUS: A workstation for manual therapy and training. I," in *Proc. Robot and Human Commun.*, 1992, pp. 161–165.
- [2] D. Reinkensmeyer, C. Takahashi, W. Timoszyk, A. Reinkensmeyer, and L. Kahn, "Design of robot assistance for arm movement therapy following stroke," *Adv. Robot.*, vol. 14, no. 7, pp. 625–637, 2001.
- [3] M. Bergamasco, A. Frisoli, and C. Avizzano, "Exoskeletons as man-machine interface systems for teleoperation and interaction in virtual environments," in *Advances in Telerobotics*, Springer, 2007, pp. 61–76.
- [4] N. Nitzsche, U. Hanebeck, and G. Schmidt, "Design issues of mobile haptic interfaces," *J. Robot. Syst.*, vol. 20, no. 9, pp. 549–556, 2003.
- [5] N. Nitzsche, U. D. Hanebeck, and G. Schmidt, "Mobile haptic interaction with extended real or virtual environments," in *Proc. IEEE Robot Human Interactive Commun.*, 2001, pp. 313–318.
- [6] A. Formaglio, D. Prattichizzo, F. Barbagli, and A. Giannitrapani, "Dynamic performance of mobile haptic interfaces," *IEEE Trans. Robot.*, vol. 24, no. 3, pp. 559–575, June 2008.
- [7] A. Frisoli, M. Bergamasco, M. C. Carboncini, and B. Rossi, "Robotic assisted rehabilitation in virtual reality with the L-EXOS," *Stud. Health Technol. Informat.*, vol. 145, pp. 40–54, 2009.
- [8] D. Reinkensmeyer, L. Kahn, M. Averbuch, A. McKenna-Cole, B. Schmit, and W. Rymer, "Understanding and treating arm movement impairment after chronic brain injury: Progress with the ARM guide," *J. Rehabil. Res. Develop.*, vol. 37, no. 6, pp. 653–662, 2000.
- [9] A. Frisoli, F. Salsedo, M. Bergamasco, B. Rossi, and M. Carboncini, "A force-feedback exoskeleton for upper-limb rehabilitation in virtual reality," *Appl. Bionics Biomech.*, vol. 6, no. 2, pp. 115–126, 2009.
- [10] A. Peattie, A. Korevaar, J. Wilson, B. Sandilands, X. Chen, and M. King, "Automated variable resistance system for upper limb rehabilitation," in *Proc. Australasian Conf. Robot. Autom.*, 2009, pp. 2–4.
- [11] G. Burdea, D. Cioi, J. Martin, D. Fensterheim, and M. Holenski, "The rutgers arm II rehabilitation system—A feasibility study," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 18, no. 5, pp. 505–514, Oct. 2010.
- [12] J. Perry, H. Zabaleta, A. Belloso, C. Rodriguez-de Pablo, F. Cavallaro, and T. Keller, "ArmAssist: Development of a functional prototype for at-home telerehabilitation of post-stroke arm impairment," in *Proc. IEEE Biomed. Robot. Biomechatron.*, 2012, pp. 1561–1566.
- [13] A. Scoglio, G. Cappiello, Z. Curto, B. Rossi, C. Avizzano, M. Satler, and E. Ruffaldi, "Haptic system and device for man-machine interaction," Patent WO/2012/114,274, Aug. 2012.
- [14] S. E. Fasoli, H. I. Krebs, J. Stein, W. R. Frontera, and N. Hogan, "Effects of robotic therapy on motor impairment and recovery in chronic stroke," *Archives Phys. Med. Rehabil.*, vol. 84, no. 4, pp. 477–482, 2003.
- [15] C. Avizzano, M. Satler, G. Cappiello, A. Scoglio, E. Ruffaldi, and M. Bergamasco, "MOTORE: A mobile haptic interface for neuro-rehabilitation," in *Proc. IEEE Robots Human Interactive Commun.*, 2011, pp. 383–388.
- [16] L. Armesto and J. Tornero, "SLAM based on kalman filter for multi-rate fusion of laser and encoder measurements," in *Proc. IEEE Intell. Robots Syst.*, 2004, vol. 2, pp. 1860–1865.
- [17] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," in *Proc. IEEE*, Mar. 2005, vol. 92, no. 3, pp. 401–422.
- [18] J. Lee, P. Dietz, D. Leigh, W. Yerazunis, and S. Hudson, "Haptic pen: A tactile feedback stylus for touch screens," in *Proc. User Interface Softw. Technol.*, 2004, pp. 291–294.
- [19] M. Satler, C. Avizzano, and E. Ruffaldi, "Control of a desktop mobile haptic interface," in *Proc. IEEE World Haptics Conf.*, 2011, pp. 415–420.
- [20] G. B. Limentani, M. C. Ringo, F. Ye, M. L. Bergquist, and E. O. MCSorley, "Beyond the t-test: Statistical equivalence testing," *Analytical Chemistry*, vol. 77, no. 11, pp. 221–226, 2005.



He is a member of the IEEE.

**Carlo Alberto Avizzano** is the coordinator of PERCRO. He is also an external professor with the University of Pisa for the course of mechatronics. His research interests include intelligent robot controllers including autonomous robots as well as human interfaces. Relevant application fields include: reactive robots, robot controllers, mobile robotics, PhD, and design of virtual behaviours. He has collaborated/coordinated several EU and National projects. He is an associated editor of several international boards and journals.



research interests include the field of teleoperation, sensor fusion, and control of mobile interfaces.

**Massimo Satler** received the MSc degree in automation engineering from the University of Pisa in 2008 and the PhD degree in perceptual robotics from the Scuola Superiore Sant'Anna in 2012. He was a visiting PhD Student at the University of Twente, The Netherlands, in 2012, where he worked at the Robotics and Mechatronics Group (RAM Lab.) on estimation of the environment in a bilateral tele-manipulation chain. He is currently within the Intelligent Automation Systems division at PERCRO and his



is a member of the IEEE. His research interests include the field of haptics and machine learning.

**Emanuele Ruffaldi** received the MSc degree in computer engineering and the PhD degree with a thesis on perceptually inspired Haptic Algorithms in 2006 from Scuola Superiore Sant'Anna. He is an assistant professor at Scuola Superiore Sant'Anna in applied mechanics. He is a member of the research Intelligent Automation Systems division at PERCRO leading the group "Sensing, Modelling and Learning for Humans." He currently participates in the EU projects PELARS (PI), REMEDI, and VERE. He

▷ For more information on this or any other computing topic, please visit our Digital Library at [www.computer.org/publications/dlib](http://www.computer.org/publications/dlib).