

A Visuo-Haptic Guidance Interface for Mobile Collaborative Robotic Assistant (MOCA)

Edoardo Lamon^{1,2*}, Fabio Fusaro^{1,3*}, Pietro Balatti^{1,2}, Wansoo Kim¹, and Arash Ajoudani¹

Abstract—In this work, we propose a novel visuo-haptic guidance interface to enable mobile collaborative robots to follow human instructions in a way understandable by non-experts. The interface is composed of a haptic admittance module and a human visual tracking module. The haptic guidance enables an individual to guide the robot end-effector in the workspace to reach and grasp arbitrary items. The visual interface, on the other hand, uses a real-time human tracking system and enables autonomous and continuous navigation of the mobile robot towards the human, with the ability to avoid static and dynamic obstacles along its path. To ensure a safer human-robot interaction, the visual tracking goal is set outside of a certain area around the human body, entering which will switch robot behaviour to the haptic mode. The execution of the two modes is achieved by two different controllers, the mobile base admittance controller for the haptic guidance and the robot’s whole-body impedance controller, that enables physically coupled and controllable locomotion and manipulation. The proposed interface is validated experimentally, where a human-guided robot performs the loading and transportation of a heavy object in a cluttered workspace, illustrating the potential of the proposed *Follow-Me* interface in removing the external loading from the human body in this type of repetitive industrial tasks.

I. INTRODUCTION

The raising awareness of worker ergonomics and the flexibility requirements of the modern enterprises have called for a radical change in manufacturing processes. Thereupon, companies started to focus on the design of new production lines, rather than on corrective ex post interventions, which were much more expensive and not always effective. Several new tools such as collaborative robots (cobots) and wearable sensors and displays (e.g., augmented reality) were introduced, with the aim to provide better working conditions for human labour, while keeping high levels of productivity, flexibility, and cost-efficiency [1] in manufacturing processes.

Cobots, in particular, have demonstrated the potential of pushing small-medium enterprises towards highly adaptive and flexible production paradigms [1], [2], while, at the same time, improving human ergonomics [3], [4]. To address fast re-programming and adaptation requirements of cobots, kinesthetic teaching and learning from demonstration [5]–[7] techniques have been introduced. Once trained with few demonstrations, a cobot can execute the same learned task repetitively.

* Contributed equally to this work. ¹ HRI² Lab, Dept. of Advanced Robotics, Istituto Italiano di Tecnologia, Via Morego 30, 16163, Genova, Italy. ² Robotics and Automation, Dept. of Information Engineering, Università degli Studi di Pisa, Pisa, Italy. ³ Department of Electronics, Information and Bioengineering, Politecnico di Milano, Milano, Italy. edoardo.lamon@iit.it

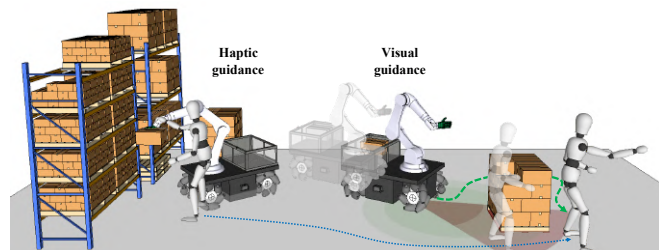


Fig. 1. Conceptual illustration of a Mobile Collaborative Assistant cooperating with a worker, by means of the haptic guidance (left) and the visual guidance while transporting loads (right), in a production environment.

Despite these advances, cobots are still far from being widely exploited because of the limitations imposed by their intrinsic complexity, which calls for specific expertise to program and operate them. Because of that, so far, cobots are exploited to perform simple and repetitive manipulation tasks in structured environments, such as the picking of small lightweight objects that are eventually assembled or placed in a location within the reachable workspace of the robot. The limitation on the workspace, caused by the length of the arm, imposes additional strict constraints on the work cell design, further restricting the usage of cobots in industry.

To address the above challenges, we recently introduced the MOBILE Collaborative robot Assistant (MOCA), a robotic coworker able to perform loco-manipulation tasks in manufacturing [8], [9] and logistic scenarios [10], [11]. The main objective was to create a modular system that can reconfigure and operate in various applications scenarios in industry, not only autonomously, but also through far-distance [12] and close-distance [11] teleoperation.

To make MOCA understandable and usable by non-experts, in this work, we propose a novel visuo-haptic guidance interface, through which a human user can intuitively guide the MOCA in the workspace to perform long-distance object picking and transportation tasks. These type of tasks are known to impose high levels of physical fatigue to human workers, contributing to the development of work-related musculoskeletal disorders and lost productivity [13].

In literature, the general problem of collaborative manipulation and objects carrying and transportation has been addressed mainly from the perspective of human-robot [14]–[17] and mobile robot-robot co-manipulation [18], [19]. In the first group, the proposed control strategies are able to identify human actions and effort, using haptic sensing [14]–[17], RGB-D visual feedback [15], [16] and learning by demonstration techniques [17]. The approaches presented in [18], [19] face the problem of the cooperative manipula-

tion of lightweight deformable and undeformable objects, respectively. All these solutions present promising results, but not suitable for the handling and transportation of large or heavy objects, since, the locomotion complexity of humanoid robots and the manipulation insufficiency of small mobile manipulators impose severe constraints on their usage in such industrial use-cases.

To go beyond the state of the art, we aim to exploit the loco-manipulation potential of MOCA, to follow human instructions for both manipulation (grasping and storage of items) and long-distance locomotion (transportation) phases. This is achieved by a novel interface called *Follow-Me*, which is composed of a haptic admittance module and a human visual tracking module. The haptic guidance enables a human worker to guide the robot end-effector in the workspace (even in long distances by using robot base mobility). With the haptic interface, it is possible to achieve accurate robot positioning to execute precise manipulation tasks, also in the case of dynamically changing environments and errors in the pose estimation of the robot. The visual interface, on the other hand, uses a human tracking module and enables autonomous and continuous navigation of the MOCA robot towards the human, with the ability to avoid static and dynamic obstacles on its path. The latter is meant to make MOCA, with the loaded items, follow human workers in long distances, by means of a visual feedback, hence removing all the external loading from the human body. To enhance human safety and agile reconfigurability, a *human interaction zone* is defined around the human body and, subsequently, the controller switches from locomotion to the haptic guidance, as soon as the distance between MOCA position and such zone get shorter, and vice-versa. The execution of the haptic guidance is achieved through an admittance controller, the visual guidance and other tasks, instead, are achieved by MOCA's whole-body impedance controller, enabling physically coupled and controllable locomotion and manipulation.

The performance of the proposed interface is evaluated experimentally, where a subject guides the MOCA to grasp, load, and transport an item in a workspace. Static obstacles were placed on the path of the robot while following its human counterpart to additionally evaluate the real-time replanning capacity of the proposed *Follow-Me* interface.

II. MOCA PLATFORM AND CONTROL

MOCA is a multi-purpose research platform, designed for human-robot physical collaborative tasks. It is composed of a robotic arm, the lightweight torque-controlled 7-DoFs Franka Emika Panda, equipped with a robotic hand, the underactuated Pisa/IIT SoftHand, mounted on top of a mobile platform, the velocity-controlled 3-DoFs Robotnik SUMMIT-XL STEEL.

The motion of MOCA is achieved by the impedance controller to deal with the different causalities of the mobile base and the robotic arm. To enhance whole-body affordances for loco-manipulation of MOCA, the two main control

modes are employed in the control architecture of the interface. First, a centralised weighted whole-body impedance controller exploits to achieve the prioritised movement at manipulation, or locomotion, or loco-manipulation that the MOCA motion enables tracking the desired task (e.g., box manipulation, human following, etc.). It allows online change of parameters, to favour alternately the arm motion in close-proximity manipulation tasks and the base mobility when navigating in free spaces. Second, a decentralised strategy was developed to obtain the haptic following, where the Cartesian wrench estimated at the end-effector of the arm, controlled with a Cartesian impedance controller, is fed directly in the base admittance controller, generating, in this way, velocity commands in the same direction of the haptic input. The main difference with the approach in [12] consists in the fact that, in the latter, during far-distance locomotion tasks, the mobile base is decoupled from the arm controller, actuated in open loop with an input torque proportional to the displacement of the centre of pressure of the human operator. Hence, a crucial improvement consists in enabling the MOCA platform to execute task targeted behaviours through the same whole-body controller, regulated by the joint-level weights, with the capability to track a desired trajectory and to reject external disturbances at the end-effector.

A. Haptic Follow-Me control

For the sake of clarity, we will first explain the equations of the decoupled impedance-admittance controllers, that constitute also the foundation of the weighted whole-body impedance controller. The dynamics of a n -DoFs torque-controlled arm can be formulated as

$$\mathbf{M}_r(\mathbf{q}_r)\ddot{\mathbf{q}}_r + \mathbf{C}_r(\mathbf{q}_r, \dot{\mathbf{q}}_r) + \mathbf{g}_r(\mathbf{q}_r) = \boldsymbol{\tau}_r + \boldsymbol{\tau}_r^{ext}, \quad (1)$$

where $\mathbf{q}_r \in \mathbb{R}^n$ is the joint angles vector, $\mathbf{M}_r \in \mathbb{R}^{n \times n}$ is the symmetric and positive definite inertial matrix of the arm, $\mathbf{C}_r \in \mathbb{R}^n$ is the Coriolis and centrifugal force, $\mathbf{g}_r \in \mathbb{R}^n$ is the gravity vector, $\boldsymbol{\tau}_r \in \mathbb{R}^n$, and $\boldsymbol{\tau}_r^{ext} \in \mathbb{R}^n$ are the commanded torque vector and external torque vector, respectively. At the lower level, the torque control compensates for gravity and the mixed Coriolis/centrifugal force

$$\boldsymbol{\tau}_r = \mathbf{g}_r(\mathbf{q}_r) + \mathbf{C}_r(\mathbf{q}_r, \dot{\mathbf{q}}_r) + \boldsymbol{\tau}^{ref}, \quad (2)$$

where $\boldsymbol{\tau}^{ref} \in \mathbb{R}^n$ is the reference torque vector. In this paper, a two-level priority Cartesian torque control is exploited as

$$\boldsymbol{\tau}^{ref} = \mathbf{J}_r^T \mathbf{F}^{ext} + (\mathbf{I} - \mathbf{J}_r^T \boldsymbol{\Lambda}_r \mathbf{J}_r \mathbf{M}_r^{-1}) \boldsymbol{\tau}_0, \quad (3)$$

where $\mathbf{J}_r \in \mathbb{R}^{6 \times n}$ is the arm Jacobian matrix, $\mathbf{F}^{ext} \in \mathbb{R}^6$ is the higher priority external force to be tracked, $\boldsymbol{\Lambda}_r \in \mathbb{R}^{6 \times 6}$ is the arm inertia and $\boldsymbol{\tau}_0 \in \mathbb{R}^n$ is the second task torque, projected onto the null space of the first task. To do this, the desired dynamic behaviour in response \mathbf{F}^{ext} and the compliant joint behaviours on the null space can be computed, respectively, using

$$\mathbf{F}^{ext} = \boldsymbol{\Lambda}_d \ddot{\tilde{\mathbf{x}}} + \mathbf{D}_d \dot{\tilde{\mathbf{x}}} + \mathbf{K}_d \tilde{\mathbf{x}} \quad (4)$$

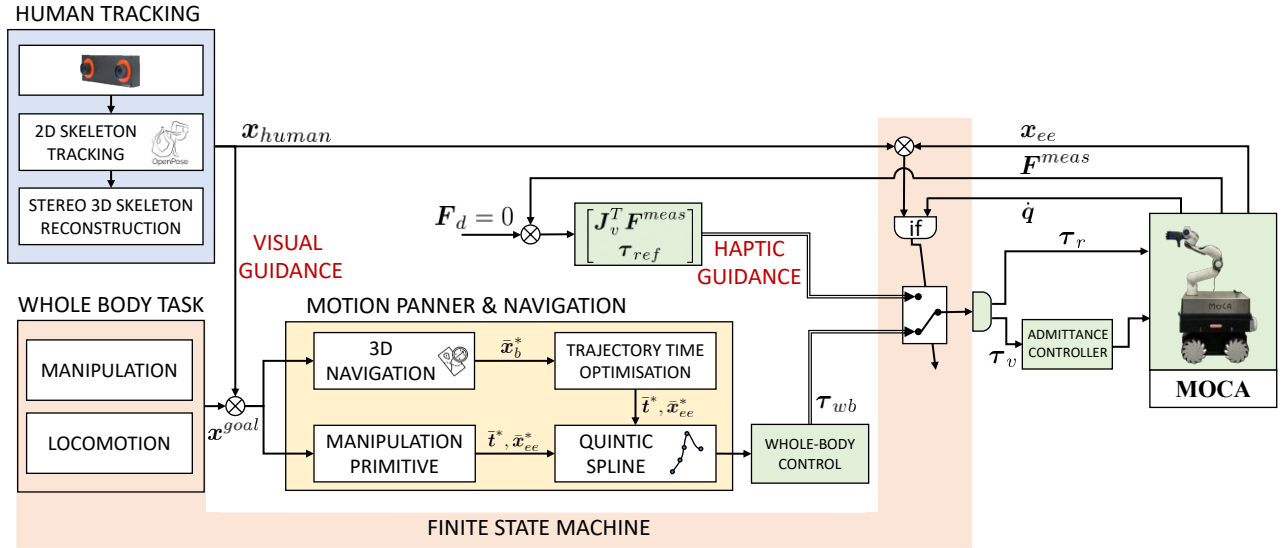


Fig. 2. Architecture of the proposed framework. High-level tasks and the Follow-Me interface are activated by a FSM. The Follow-Me interface switches on, alternately, the visual locomotion or the haptic guidance, according to the human tracked pose. The haptic interface exploits an admittance control to allow the human to move the robot. A whole-body control strategy and motion planner are in charge of computing trajectories for manipulation and locomotion tasks. A 6D path planner is added to ensure collision avoidance during locomotion tasks.

and

$$\tau_0 = -D_0 \dot{q} - K_0 (q - q_0), \quad (5)$$

where $\tilde{x} = x^{des} - x \in \mathbb{R}^6$ is the Cartesian error computed with respect to the desired Cartesian equilibrium pose x^{des} , $\Lambda_d \in \mathbb{R}^{6 \times 6}$, $D_d \in \mathbb{R}^{6 \times 6}$ and $K_d \in \mathbb{R}^{6 \times 6}$ are the desired Cartesian inertia, damping and stiffness, respectively. Moreover, q_0 is a desired joint configuration, $K_0 \in \mathbb{R}^{n \times n}$ and $D_0 \in \mathbb{R}^{n \times n}$ are the desired joint-space stiffness and damping.

Instead, the admittance model of mobile base, with virtual joints $q_v \in \mathbb{R}^m$, where the number of DoFs $m = 3$, can be described by

$$M_{adm} \ddot{q}_v^{des} + D_{adm} \dot{q}_v^{des} = \tau_v^{vir} + \tau_v^{ext}, \quad (6)$$

where $M_{adm} \in \mathbb{R}^{m \times m}$ and $D_{adm} \in \mathbb{R}^{m \times m}$ are the virtual inertial and virtual damping, $\dot{q}_v^{des} \in \mathbb{R}^m$ is the input velocity sent to the mobile platform, $\tau_v^{ext} \in \mathbb{R}^m$ and $\tau_v^{vir} \in \mathbb{R}^m$ are the external and the virtual torque, respectively. The desired velocity \dot{q}_v^{des} is then calculated by substituting $\ddot{q}_v^{des}(t) = t_s^{-1}(\dot{q}_v^{des}(t) - \dot{q}_v^{des}(t-1))$ as

$$\dot{q}_v^{des}(t) = (t_s^{-1} M_{adm} + D_{adm})^{-1} (\tau_v^{vir}(t) + \tau_v^{ext}(t) + t_s^{-1} M_{adm} \dot{q}_v^{des}(t-1)), \quad (7)$$

where t_s is the sampling time. In this control mode, we set $\tau_v^{vir} = 0$ and $\tau_v^{ext} = J_v^T F^{ext}$, where $J_v \in \mathbb{R}^{6 \times m}$ is the Jacobian matrix of mobile base and $F^{ext} \in \mathbb{R}^6$ is the estimated force at the end-effector F^{meas} . Consequently, the admittance controller will generate the desired velocity that will be regulated by the forces applied by the human on the end-effector. By changing the values of M_{adm} and D_{adm} , we can shape the relation between the force applied and the response of the system. According to (7), the virtual mass of the base represents the ratio between τ_v^{ext} and \dot{q}_v^{des} . For instance, if $\|M_{adm}\| \rightarrow 0$, for a fixed τ_v^{ext} , the admittance

controller will generate velocities $\dot{q} \rightarrow \infty$. If $\|M_{adm}\|$ is low, small forces measured at the end-effector allow the human to easily move the robot. Furthermore, when the haptic interface is active, the Cartesian stiffness K_d along the z axis is set to 0, so, in that axis, the arm can freely move just gravity compensated.

B. Weighted Whole-Body Impedance Control

The weighted whole-body impedance controller is in charge of computing the high-levels torque references for the low-level torque controller of the manipulator, that compensates for the gravity and the Coriolis and centrifugal forces, and for the admittance controller of the mobile base, that generates feasible velocity commands for its low-level velocity controller. The whole-body dynamic model of MOCA is the result of the composition of the parallel of the dynamics of the arm and the base. Under the assumptions that the motion of the mobile platform does not affect the motion of the manipulator, and neglecting the dynamics of the low-level velocity controller, i.e. $\ddot{q}_v \approx \dot{q}_v^{des}$, we can write the following whole-body dynamics¹ as

$$\begin{aligned} & \begin{pmatrix} M_{adm} & \mathbf{0} \\ \mathbf{0} & M_r \end{pmatrix} \begin{pmatrix} \ddot{q}_v \\ \ddot{q}_r \end{pmatrix} + \begin{pmatrix} D_{adm} & \mathbf{0} \\ \mathbf{0} & C_r \end{pmatrix} \begin{pmatrix} \dot{q}_v \\ \dot{q}_r \end{pmatrix} + \begin{pmatrix} \mathbf{0} \\ g_r \end{pmatrix} \\ & = \begin{pmatrix} \tau_v^{vir} \\ \tau_r \end{pmatrix} + \begin{pmatrix} \tau_v^{ext} \\ \tau_r \end{pmatrix}, \end{aligned} \quad (8)$$

that can be summarised by

$$M \ddot{q} + C \dot{q} + g = \tau + \tau^{ext}. \quad (9)$$

The robot joint torques vector $\tau \in \mathbb{R}^n$ is obtained by solving the problem of finding the torque vector τ closest to

¹ For the sake of readability, the dependencies on q and x are dropped from now on.

desired τ_0 ,

$$\min_{\tau} \frac{1}{2} \|\tau - \tau_0\|_{\mathbf{W}}^2 \quad \text{s.t. } \bar{\mathbf{J}}^T \tau = \mathbf{F} \quad (10)$$

where $\bar{\mathbf{J}}^T = (\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T)^{-1}\mathbf{J}\mathbf{M}^{-1}$. $\bar{\mathbf{J}}$ is the *dynamically consistent pseudo-inverse* of $\mathbf{J}(\mathbf{q})$, and the constraint $\bar{\mathbf{J}}^T \tau = \mathbf{F}$ is the general relationship between the generalised joint torques and the operational forces. By letting the positive definite weighting matrix $\mathbf{W} \in \mathbb{R}^{(n+m) \times (n+m)}$, the solution of problem (10) can be obtained by

$$\begin{aligned} \tau_{wb} = & \mathbf{W}^{-1}\mathbf{M}^{-1}\mathbf{J}^T\Lambda_{\mathbf{W}}\Lambda^{-1}\mathbf{F} + \\ & + (\mathbf{I} - \mathbf{W}^{-1}\mathbf{M}^{-1}\mathbf{J}^T\Lambda_{\mathbf{W}}\mathbf{J}\mathbf{M}^{-1})\tau_0, \end{aligned} \quad (11)$$

where

$$\Lambda_{\mathbf{W}} = \mathbf{J}^{-T}\mathbf{M}\mathbf{W}\mathbf{M}\mathbf{J}^{-1} \quad (12)$$

can be regarded as the *weighted Cartesian inertia*, analogous to the Cartesian inertia $\Lambda(\mathbf{x}) = (\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T)^{-1}$. \mathbf{F} and τ_0 can be computed according to (4) and (5), respectively. The structure of the matrix \mathbf{W} can be defined as

$$\mathbf{W}(\mathbf{q}) = \mathbf{H}^T \mathbf{M}^{-1}(\mathbf{q})\mathbf{H}, \quad (13)$$

where $\mathbf{H} \in \mathbb{R}^{(n+m) \times (n+m)}$ is the tunable positive definite weight matrix of the controller. In particular, in this paper, \mathbf{H} is diagonal and dynamically selected depending on the task. A possible choice is represented by

$$\mathbf{H} = \begin{bmatrix} \eta_B \mathbf{I}_{m \times m} & \mathbf{0}_{m \times n} \\ \mathbf{0}_{n \times m} & \eta_A \mathbf{I}_{n \times n} \end{bmatrix}, \quad (14)$$

where $\eta_B, \eta_A > 0$ are constant scalar values, a higher value of this will impede the motion. For instance, to obtain higher mobility of the arm than the base, we set $\eta_B > \eta_A$.

III. VISUO-HAPTIC FOLLOW-ME INTERFACE

A. Navigation and Trajectory Planning

In order to ensure the generation of feasible reference trajectories for the whole-body controller, we bestow MOCA of a 6D Cartesian path planner, that allows to navigate in space while avoiding fixed and moving obstacles, and a trajectory planner, for both manipulation and locomotion tasks. In general, for a locomotion task, we need to specify a robot desired pose $\mathbf{x}^{goal} \in \mathbf{R}^3 \times \mathbf{R}P^3$ defined w.r.t. a specific frame in space. Given \mathbf{x}^{goal} and the initial pose \mathbf{x}^{init} , the path planner generates a sequence of collision-free waypoints $\bar{\mathbf{x}}^*$, where $\bar{\mathbf{x}}^* = [\mathbf{x}^{*,0} \dots \mathbf{x}^{*,k} \dots \mathbf{x}^{*,K-1}]$ with $k = 0, \dots, K-1$. These waypoints are later interpolated by means of geometric paths with timing laws. In this way, we can set, for each control loop, a desired equilibrium pose \mathbf{x}^{des} of the controller.

To generate a suitable sequence of waypoints for the end-effector ${}^w\bar{\mathbf{x}}_{ee}^*$, defined in the world frame, we take advantage of an existing planner algorithm for mobile robots². Such algorithm enables to deal with unknown environments, hence, with empty maps and moving obstacles. As local planner, we exploited TEB (Timed Elastic Band)³, that computes a cost

map which fuses the data sampled by the perception systems, such as lasers and the front camera, with the odometry estimation and computes the path by minimising the overall navigation cost. The output of this planner is a sequence of 3D spatially not uniformly distributed waypoints ${}^w\bar{\mathbf{x}}_b^* \in \mathbf{R}^2 \times \mathcal{S}^1$. Then, through kinematics mapping, we hypothesise suitable desired poses of the end-effector. We opted to use a 3D Cartesian planner instead of a joint space planner like RRT mainly for computational reasons: the search space of our planner lies in a subspace of dimension 3, instead, the search space of joint space planner would lie in a subspace of dimension 10. The drawback of the proposed method consists in the fact that we do not exploit the whole-body strategy to avoid obstacles. Nevertheless, in our framework, this choice does not impose any limitation to the robot loco-manipulation capabilities.

Let's start with finding the goal for the 3D planner. The 6D goal pose⁴ ${}^w\mathbf{T}_{ee}^{goal}$ has to be projected on a subspace of the 6D space, that is the joint space of the base (translation in the x - y plane and the rotation around the z axis). In order to compute the 3D goal pose ${}^w\mathbf{T}_b^{goal}$ that will be sent to the 3D planner, the rotational and translational part are computed separately. We assume that the relative pose of the end-effector w.r.t. the base of the robot ${}^b\mathbf{T}_{ee}$ at the end of the motion is the same as the one at the beginning; in this way to compute ${}^w\mathbf{T}_b^{goal}$, we can take the yaw angle and the translational component directly from ${}^w\mathbf{T}_{ee}^{goal}$, both defined in the base frame w.r.t. the world frame. Once the 3D planner has received ${}^w\mathbf{T}_b^{goal}$, it outputs the 3D pose of each waypoint. Then, this waypoints are projected to the 6D space keeping roll, pitch, and the z axis constant as in the initial pose of the end-effector in the world frame ${}^w\mathbf{T}_{ee}$. Noteworthy, ${}^w\mathbf{T}_{ee}^{goal}$ is added in the queue of the waypoints vector to ensure that ${}^w\mathbf{T}_{ee}^{goal}$ is reached.

One crucial requirement for enabling a loco-manipulation task is to be able to smoothly interpolate the waypoints ${}^w\bar{\mathbf{x}}_{ee}^*$. A possible simple solution consists in using *quintic polynomial* functions that allow setting an initial, final velocity and acceleration, usually set to 0. However, with this method, the robot will stop and restart the motion at each waypoint. To ensure the continuity of motion, we use *quintic Bezier splines*, a piecewise polynomial parametric curve that ensures \mathcal{C}^2 continuity at the waypoint [20]. Furthermore, the desired trajectory continuously update while the previous motion is still executing that enables to take into account for the presence of moving obstacles.

However, the problem of finding a trade-off between the spatial and temporal waypoint is not solved. For instance, in long-distance locomotion, the relative distance between waypoints might vary widely. If the waypoints are uniformly distributed in time, but not spatially, the computed trajectories will present high velocities and accelerations, and, consequently, the robot controller will generate undesired or infeasible motions to track such trajectories. On the contrary, if the quintic spline interpolates waypoints with a long

² http://wiki.ros.org/move_base

³ http://wiki.ros.org/teb_local_planner

⁴ For the sake of clarity, we adopt just for this section the homogeneous transformation matrices \mathbf{T} to specify a pose instead of \mathbf{x} .

waypoints duration, the path generated might present high curvature. Our solution consists in finding a desired temporal window $[0, T_{max}]$, where we can distribute all the time waypoints, proportional to the maximum desired velocity v_{max} , then generate accordingly the time samples \bar{t}^* , and finally the spline $s(t)$ from the pairs time/waypoint \bar{t}^* , \bar{x}_{ee}^* . Then the length of the curve $\ell(s(t))$ is measured. The desired spline is the one that minimises the curvature length. The pseudocode for a 1D trajectory is summarised in algorithm 1.

Algorithm 1 Trajectory Time Optimisation.

```

1: procedure  $s_{min}(t) = \text{TRAJTIMEOPT}(\bar{x}^*, v_{max})$ 
2:    $T_{min} \leftarrow (x_{final} - x_{init}) * v_{max}^{-1}$ 
3:    $T_{max} \leftarrow \alpha T_{min}$ , with  $\alpha > 1$ 
4:   for  $n < N$  do
5:     generate  $\bar{t}^*$  where
6:      $\triangleright t_i \sim \mathcal{U}(0, T_{max})$  with  $i = 1, \dots, K - 1$ 
7:      $\triangleright t_0 \leftarrow 0$ 
8:      $\text{sort}(\bar{t}^*)$ 
9:      $s(t) \leftarrow \text{SPLINE5}(\bar{t}^*, \bar{x}^*, \dot{x}_i, \dot{x}_f, \ddot{x}_i, \ddot{x}_f, )$ 
10:     $\ell \leftarrow \int_0^{t_I} \sqrt{\dot{s}^2} dt$ 
11:    if  $\ell < \ell_{min}$  then
12:       $\ell_{min} \leftarrow \ell$ 
13:       $s_{min}(t) \leftarrow s(t)$ 
14:    end if
15:  end for
16: end procedure

```

For $N \rightarrow \infty$ the algorithm ensures the spline that minimises the trajectory length is found. In practice, N can be set according to the desired computational planning time.

B. Human Tracking

In this section we will describe the visual human whole-body tracking module. In literature, different sensory systems are able to provide accurate position, velocity and acceleration information of a body, such as optical (OptiTrack⁵, Vicon⁶, etc.) and inertial (Xsens suit⁷) motion capture systems. These systems, unfortunately, present some disadvantages, that make them unsuitable for industrial scenarios. For these reasons, we opted for a solution that does not require the human worker to wear any sensor. To do this, we exploit a vision-based skeleton tracker algorithm, which adopts an OpenPose pre-trained deep learning model [21], to detect human skeletons extracted from the frames of a stereo camera⁸ at 25 Hz, calibrated with respect to the world frame of the robot. The output of the detector consists in the position of 25 skeleton keypoints in pixel coordinates $[u_i, v_i]$, with $i = 0, \dots, 24$, and the associated likelihood. To estimate the 3D coordinates of each keypoint from the 2D pixel coordinates, we exploited the stereo 3D projection: by means of the camera projection matrices P_{left} , P_{right} , we triangulate the position of a keypoint in 3D coordinates from right and left detections. It is also possible to repeat the triangulation for the $N - 1$ neighbour pixels to the detected ones and then average the results. To further strengthen the result of the procedure, we consider the dynamics of the detection by applying a median filter with a fixed-size

moving window, so the keypoint values are averaged also in time. The values of skeleton keypoints are then broadcast to the *visuo-haptic Follow-Me* interface at the frequency of 10 Hz. A more accurate description of the filters applied to strengthen the skeleton detection can be found in [3]. In case the right and left detections do not match, due to delays and wrong detections, we do not update the 3D human position. Besides, to enhance the detection robustness we remove possible false positives, by considering the likelihood associated to the keypoint provided by OpenPose. In case of an object is misdected or only few keypoints of a human are detected, the probability associated to each keypoint is low. In our scenario, a keypoint is detected with probability lower than a fixed threshold p_{th} , we considered the keypoint as not detected, and, thus, we do not broadcast the information to the visual guidance module. In this work, we experimentally set p_{th} to a conservative value of 0.65.

C. FSM & Modes switching

A high-level Finite State Machine (FSM) aims at enabling for continuous transition from one mode to another, in response to the human tracking module and to the whole-body task module. FSM is in charge of switching the different modes of operation and actions, which are autonomously executed by the MOCA, such as grasp, locomotion, etc., and to activate the *visuo-haptic Follow-Me* interface. Every action wraps different informations, for instance the action-related desired pose and the control parameters (Cartesian and joint-space stiffness K_0 , K_0 , controller weight H). Our general strategy consists in being always compliant, to minimise the eventual exerted force in unexpected contacts with the environment and with other coworkers, and stiffening only if required by precision manipulation tasks. If the *visuo-haptic Follow-Me* interface is activated, the distance between the end-effector of MOCA and the "neck" keypoint x_{human} of the tracked skeleton is measured. If this distance in the x - y plane is farther than a certain *human interaction zone* δ_{human} , the *visual guidance* is then activated, otherwise, the *haptic guidance* is executed.

In the *visual guidance* mode, the projection of the human pose x_{human} on a circular bounding box of the human, along the axis represented by the distance between the robot and human, is passed to the 6D navigation stack. The size of the bounding box depends on a *safety threshold* δ_{safety} , that, in practice, scales the human pose. In this way, the robot goal pose is set on the boundary of the safety zone to account for inaccurate human tracking. To determine more accurately the size of safety zones such as δ_{safety} , in compliance with the requirements of *ISO/TS 15066*, one might refer to [22]. Moreover, the desired orientation is independent of human orientation. Thanks to this trick, the human can freely turn around without producing unnecessary motions of the robot. To avoid the continuous update of the robot goal, and, hence, to prevent unnecessary motions, the goal pose is updated when the distance between the position of the human is greater than a fixed distance δ_{goal} . The new path is continuously computed at 10 Hz but it is updated at 0.3

⁵ optitrack.com ⁶ vicon.com ⁷ xsens.com
⁸ roboception.com/product/rc_visard-160-color/

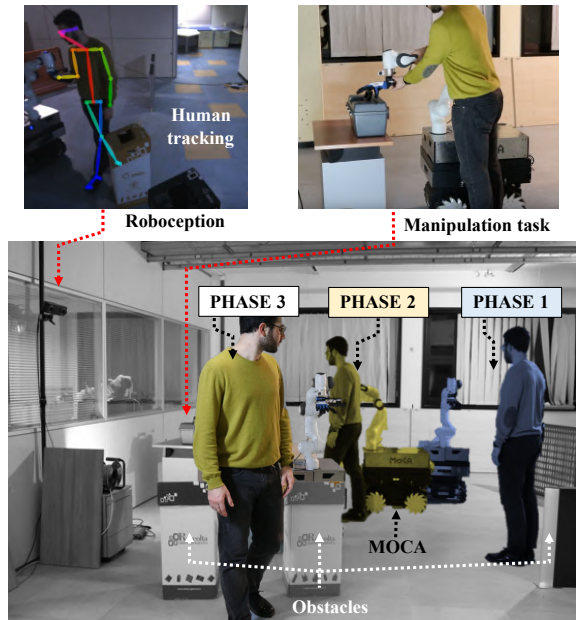


Fig. 3. Experimental setup for the visuo-haptic Follow-Me interface. The three main phases of the experiment (bottom) consist of a visual locomotion to reach the human partner (PHASE 1), a manipulation task (PHASE 2), i.e. load carrying (top right), and human following (PHASE 3). The execution of the phases is triggered by the Follow-Me interface according to the human tracked pose (top left).

H_z. The navigation stack computes collision-free pairs of time-position waypoints that are later smoothly interpolated by quintic splines, as explained in III-A. Once the spline is computed, a new equilibrium pose is passed to the whole-body controller.

In the *haptic guidance* mode, i.e. if the distance is less than *human interaction zone* δ_{human} , the robot will switch from whole-body control to the decoupled control, with an arm desired robot pose equal to the actual one. In this way, the robot will stop any motion performed and wait for the haptic interaction. In the base admittance controller, we assume that the external forces applied at the end-effector, as explained in II-A, are defined in the end-effector frame with the same orientation of the world frame. Hence, the human worker can move manually the robot to a precise desired pose. Since the relationship between the force estimated at the end-effector and the actuation velocity depends mainly on the value of the virtual mass M_{adm} of the admittance controller, to provide higher velocity with a small estimated force we set $M_{adm} = \text{diag}\{20, 20, 4\}$ (see Tab. I). On the contrary, if the distance is higher than δ_{human} the guidance mode will switch back to the visual guidance and restart following the human with whole-body control as soon as the human move far from MOCA. To ensure the safe behaviour and avoid the tracking error during the *haptic guidance mode*, the FSM keep checking the joint velocity. For instance, if the joint velocity is different from zero, due mainly to the admittance controller, the mode will stay with the *haptic guidance mode*, even in case of the wrong detection of the human. It is a crucial aspect that the human intention or action is assigned to higher priorities during the collaboration.

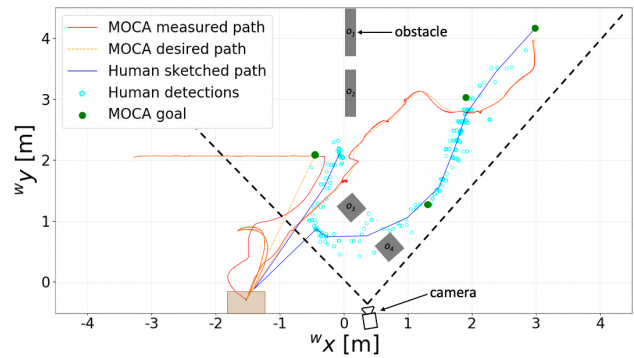


Fig. 4. Plots of human tracked position (blue circles) and robot desired/executed plan (orange dashed/red continuous) in the x - y plane. The collision-free plan is continuously updated according to the human position, that constitutes the locomotion goals (green dots) in the visual guidance.

TABLE I

VALUE OF PARAMETERS USED IN THE EXPERIMENTAL SETUP

Parameter	Value	Description
δ_{human}	0.5	Human interaction zone [m]
δ_{safety}	0.7	Safety distance threshold [m]
δ_{goal}	1.5	Update goal distance [m]
f_{th}	7	Contact force threshold [N]
M_{adm}^{haptic}	$\text{diag}\{20, 20, 4\}$	Mass (haptic Follow-Me) [Kg · m]
M_{adm}^{wb}	$\text{diag}\{32.5, 32.5, 6.5\}$	Mass (wb) [Kg · m]
η_B^{manip}	$\sqrt{8}$	Weight base (wb manipulation)
η_B^{loc}	1	Weight base (wb locomotion)
η_A	1	Weight arm (wb)

IV. EXPERIMENTS

A. Experimental setup

The proposed interface was validated with a proof-of-concept experiment in an unstructured scenario, typical of fast-reconfigurable and flexible factory setup, where the robot will have to achieve sequentially either a manipulation task and different navigation ones when triggered by the visuo-haptic interface. The task-related desired positions were not pre-planned but online assigned by means of the visuo-haptic interface, according to the human actions. Some objects were placed in the environment to prove that with the proposed planner algorithm, the MOCA was not just following the human partner, but also was able to avoid obstacles.

The experimental setup and the main experiment phases are depicted in Fig. 3. In PHASE 1, a human worker was located in the trackable area of the camera, while MOCA approached him/her through the *visual guidance* mode. During PHASE 2, the *haptic guidance* was activated (subsequent to the MOCA approaching human), during which the subject could manually guide MOCA end-effector in space, by exploiting arm and mobile base, to perform a manipulation task. Once a target object was reached using this mode, the user pushed a button, which is located on the robot end-effector, to trigger a set of pre-planned actions: first, the SoftHand closed the fingers to grasp the object, then a spline was generated to place the object on the carriage space of the MOCA robot. To obtain such a motion, the weight of the whole-body controller was set to favour arm motions rather than base motions (see Tab. I for the values of the parameters)

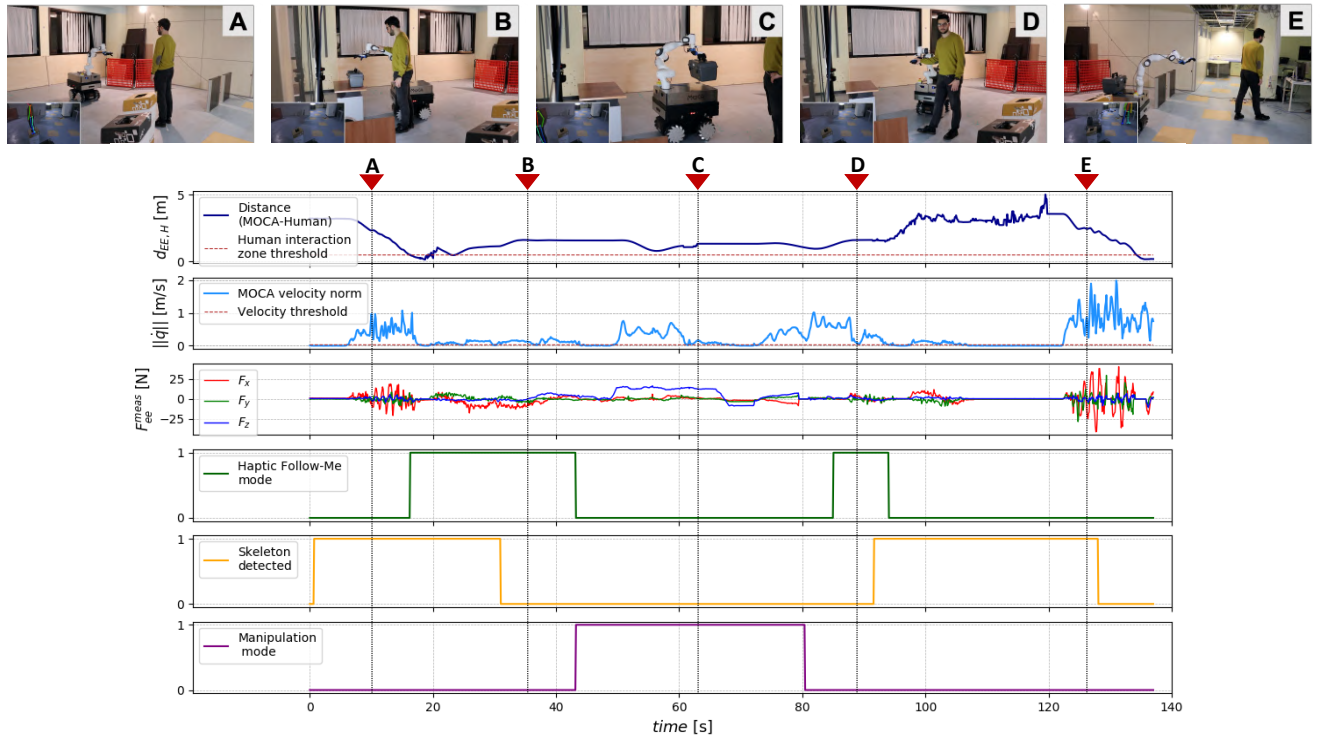


Fig. 5. Snapshots of the experiment. MOCA joins the human through the visual guidance (A), then the location of the manipulation task is reached by means of the haptic guidance (B), and MOCA performs a manipulation task (C). Once finished, the previous haptic guidance mode is activated (D). Finally, the robot dynamically follows the human through the visual guidance (E). Lower plots show the values monitored to obtain the correct modes switching, that are the distance between the MOCA and the partner, MOCA joint velocities, end-effector measured forces, if the human is tracked, and the manipulation mode. The combination of these variable triggers the activation of the haptic Follow-Me.

and the joint-space stiffness \mathbf{K}_0 was set to 0. After releasing the toolbox, the arm moved back to the initial position. For safety considerations, during the toolbox placement, the arm stopped its motion if the estimated force at the end-effector was higher than a *force threshold* \mathbf{f}^{th} . In PHASE 3, the MOCA carried the box to a desired position, which was defined by the human via the *visuo-haptic interface*. The human worker was asked to pass through a narrow way on his/her path, with a 0.5 m distance between two obstacles (o_3 and o_4), causing a potential collision for the MOCA platform (approx. length 0.6 m) (See Fig. 4). In this situation, to follow the human via the *visuo-haptic interface*, the robot has to recognise and avoid the obstacles by generating a new obstacle-free path.

B. Results

The full experiments paths on the x - y plane of the human, tracked by the vision system, and MOCA, both planned and executed, are depicted in Fig. 4. The figure shows that the desired MOCA trajectory is correctly tracked by the controller (red and orange lines); the goals sent to the robot (green dots) correspond to human detected pose (cyan circles), according to the update rule imposed by δ_{goal} and the frequency. Moreover, even if the human subject passes through the two obstacles, the proposed planner algorithm generates collision-free paths that reach eventually the human. In Fig. 5 we show the different phases of the experiment and the logged data from each phase of the experiment, used to trigger the different *haptic* and *visual guidance*. In the

initial phase (A) the human is detected and its distance is higher than the *human interaction zone* δ_{human} . For this reason, the visual guidance is activated and MOCA moves towards the human. When the human enters the *interaction zone*, the haptic guidance is activated and the human subject can move MOCA in the desired pose of the manipulation action (B) and then activate the primitive by pressing the button (C). In that area the person cannot be tracked, so for safety issues, we prevent any automatic change of mode. For this reason, the haptic phase lasts until the button is pressed and the manipulation primitive until it finishes. When the manipulation finishes, the haptic following is reactivated (D), even if the person is not tracked. We want to point out that in this phase, even if the human is detected, and his distance is higher than the *interaction zone*, the mode does not change because of the robot velocity generated by the admittance controller. Whenever the subject exits the *interaction zone*, the visual guidance triggers the whole-body locomotion and the planner generates online collision-free paths to follow the moving subject (E).

To further evaluate the effectiveness of the proposed interface in removing the external loading from the human body, we estimated the human effort in our experiment and compared it with the effort that the same worker would allocate to achieve the same action manually. In order to measure this effort, similarly to [23], [24], we calculated impulse, i.e. the norm of the aggregate of the forces in time exerted by the human, in both tasks. In the manual

transportation of the load, we considered just the gravity force that the human exerts when he is carrying the toolbox. Since the toolbox weights approx. 1.8 Kg, the force applied by the human is approx. 17.64 N, for an average duration of the transportation task of 25 s, from the toolbox grasping to the final transportation location. In this case, the estimated impulse is 441 Ns. On the other hand, in our experiment, the worker exerts a force on the robot only during the *haptic guidance*. Hence, the human impulse was calculated from the measured wrench F_{ee}^{meas} that corresponds to the *haptic phase*. From the plot in Fig. 5, it is clear that such forces are quite small, bounded to ± 10 N. The resulting impulse is 292.7 Ns, that is lower than the impulse required to achieve the same task manually. Furthermore, we would like to highlight that, in the case of heavier boxes, or longer transportation distance (and, therefore, longer duration), the effort in the manual execution would increase, while using our interface it will remain constant. The reason is that the forces applied by the human are needed only in haptic phase, to obtain an accurate positioning of the robot for the manipulation task, instead, during the locomotion task, the load is entirely carried by MOCA.

V. CONCLUSION

In this work, we presented a novel visuo-haptic interface for collaborative mobile assistants to simplify the physical interaction between human workers and robotic assistants. The proposed framework enabled MOCA to simultaneously adapt to the co-worker's behaviour by means of the FSM and different control modules to achieve haptic and visual guidance. The results showed the potential of the interface in offloading human workers from heavy tasks, providing, at the same time, natural and simplified interactions between coworkers, demonstrating the potential of collaborative robot in improving labourer ergonomics conditions and boosting, at the same time, also flexibility. The proposed interface in this work is suitable not only for flexible logistic of medium-small industrial enterprises, but also for commercial scenarios, like department stores, where the employees are required to bring goods from the storage to shelves or exhibitors. Future works will focus on the ergonomics assessment of the proposed interface and on the improvement of the vision module to enable multi-human tracking, enabling MOCA to distinguish the human partner from other coworkers.

ACKNOWLEDGEMENT

This work was supported by the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 871237 (SOPHIA).

REFERENCES

- [1] A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kose, and O. Khatib, "Progress and prospects of the human-robot collaboration," *Autonomous Robots*, vol. 42, pp. 957–975, 2018.
- [2] A. Cherubini, R. Passama, A. Crosnier, A. Lasnier, and P. Fraitse, "Collaborative manufacturing with physical human-robot interaction," *Robotics and Computer-Integrated Manufacturing*, vol. 40, 2016.
- [3] W. Kim *et al.*, "Adaptable workstations for human-robot collaboration: A reconfigurable framework for improving worker ergonomics and productivity," *IEEE Robotics Automation Magazine*, Sep. 2019.

- [4] M. Lorenzini, W. Kim, E. D. Momi, and A. Ajoudani, "A new overloading fatigue model for ergonomic risk assessment with application to human-robot collaboration," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 1962–1968.
- [5] P. Kormushev, S. Calinon, and D. G. Caldwell, "Imitation learning of positional and force skills demonstrated via kinesthetic teaching and haptic input," *Advanced Robotics*, vol. 25, no. 5, pp. 581–603, 2011.
- [6] A. Muxfeldt, J.-H. Kluth, and D. Kubus, "Kinesthetic teaching in assembly operations – a user study," in *Simulation, Modeling, and Programming for Autonomous Robots*. Springer, 2014.
- [7] C. Schou, J. S. Damgaard, S. Bøgh, and O. Madsen, "Human-robot interface for instructing industrial tasks using kinesthetic teaching," in *IEEE ISR 2013*, Oct 2013, pp. 1–6.
- [8] W. Kim, M. Lorenzini, P. Balatti, Y. Wu, and A. Ajoudani, "Towards ergonomic control of collaborative effort in multi-human mobile-robot teams," *The IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.
- [9] M. Lorenzini, F. Fusaro, P. Balatti, E. De Momi, F. Mastrogiovanni, W. Kim, and A. Ajoudani, "Toward a synergistic framework for human-robot coexistence and collaboration (hrc2)," *Italian Conference on Robotics and Intelligent Machines (I-RIM)*, 2019.
- [10] E. Lamon, A. Leonori, W. Kim, and A. Ajoudani, "Towards an intelligent collaborative robotic system for mixed case palletizing," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2020.
- [11] W. Kim, P. Balatti, E. Lamon, and A. Ajoudani, "Moca-man: A mobile and reconfigurable collaborative robot assistant for conjoined human-robot actions," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2020.
- [12] Y. Wu, P. Balatti, M. Lorenzini, F. Zhao, W. Kim, and A. Ajoudani, "A teleoperation interface for loco-manipulation control of mobile collaborative robotic assistant," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3593–3600, Oct 2019.
- [13] S. Bevan *et al.*, "Fit for work," *Musculoskeletal disorders in the European workforce. London: The work foundation*, 2009.
- [14] A. Mörtl *et al.*, "The role of roles: Physical cooperation between humans and robots," *The International Journal of Robotics Research*, vol. 31, no. 13, pp. 1656–1674, 2012.
- [15] D. J. Agravante, A. Cherubini, A. Bussy, P. Gergondet, and A. Kheddar, "Collaborative human-humanoid carrying using vision and haptic sensing," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 607–612.
- [16] J. Stückler and S. Behnke, "Following human guidance to cooperatively carry a large object," in *2011 11th IEEE-RAS International Conference on Humanoid Robots*, Oct 2011, pp. 218–223.
- [17] L. Rozo, D. Bruno, S. Calinon, and D. G. Caldwell, "Learning optimal controllers in human-robot cooperative transportation tasks with position and force constraints," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2015.
- [18] J. Alonso-Mora, R. Knepper, R. Siegwart, and D. Rus, "Local motion planning for collaborative multi-robot manipulation of deformable objects," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 5495–5502.
- [19] E. Simetti, G. Casalino, F. Wanderlingh, and M. Aicardi, "A task priority approach to cooperative mobile manipulation: Theory and experiments," *Robotics and Autonomous Systems*, vol. 122, 2019.
- [20] H. Prautzsch, W. Boehm, and M. Paluszny, *Bézier and B-spline techniques*. Springer Science & Business Media, 2013.
- [21] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields," in *arXiv preprint arXiv:1812.08008*, 2018.
- [22] J. Saenz, C. Vogel, F. Penzlin, and N. Elkmann, "Safeguarding collaborative mobile manipulators - evaluation of the valeri workspace monitoring system," *Procedia Manufacturing*, vol. 11, pp. 47 – 54, 2017, 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM 2017.
- [23] K. Alexopoulos, D. Mavrikios, and G. Chrysolouris, "Ergotoolkit: an ergonomic analysis tool in a virtual manufacturing environment," *International Journal of Computer Integrated Manufacturing*, 2013.
- [24] P. Moschonas, N. Kaklanis, and D. Tzovaras, "Novel human factors for ergonomics evaluation in virtual environments using virtual user models," in *Proceedings of the 10th International Conference on Virtual Reality Continuum and Its Applications in Industry*, 2011.