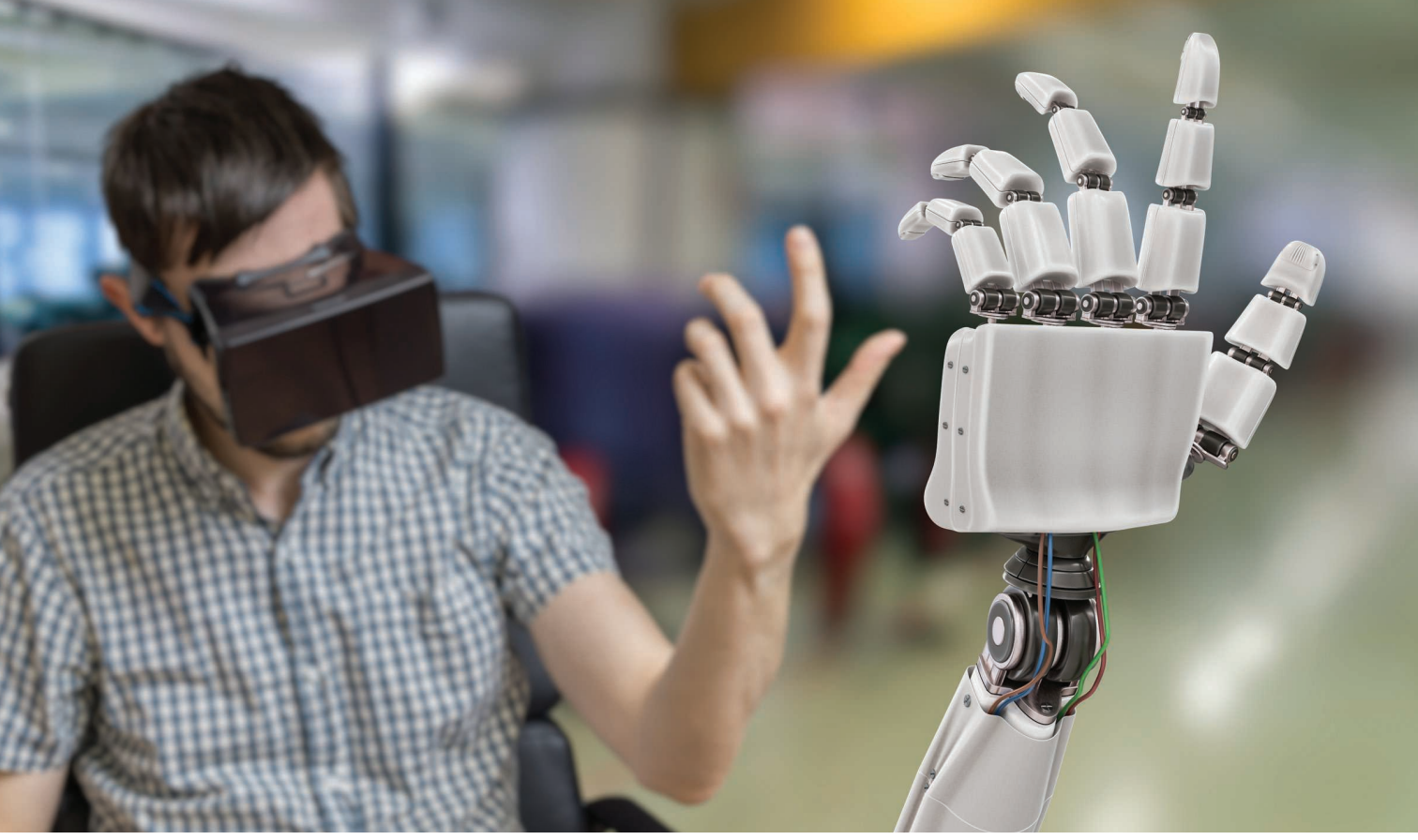


©ISTOCKPHOTO.COM/VCHAL

By Luigi Penco, Nicola Scianca, Valerio Modugno, Leonardo Lanari, Giuseppe Oriolo, and Serena Ivaldi

A Multimode Teleoperation Framework for Humanoid Loco-Manipulation

An Application for the iCub Robot

Over the years, there have been many improvements in job-related safety standards and working conditions, but there are still many situations and environments where human lives are put at risk, such as in search and rescue situations, construction sites, and chemical plants. We envision a world where robots can act as physical avatars and effectively replace humans in those hazardous scenarios through teleoperation.

Despite the many successful cases of teleoperating mobile robots and manipulators, even in space, the teleoperation of humanoid robots still presents major challenges. While humanoid robots are designed with the ambition of

Digital Object Identifier 10.1109/MRA.2019.2941245
Date of current version: 28 October 2019

mimicking the human body's capabilities, differences in kinematics (e.g., structure and joint limits) and dynamics (e.g., mass distribution and inertia) are still significant. Another crucial issue is the necessity of ensuring the dynamic balance of the robot while trying to imitate human motion. This is not straightforward during locomotion tasks, in which the dynamics are highly involved.

A possible solution is, therefore, to use two forms of teleoperation: a low-level one for manipulation, realized via whole-body teleoperation (Figure 1), and a high-level type for locomotion, based on the generation of reference velocities that are then tracked by the robot. We believe that this combination of different modes of teleoperation will considerably ease the burden of controlling humanoid robots, ultimately increasing their adaptability to complex situations that cannot be handled satisfactorily by fully autonomous systems.

Background and Contribution

Workplace Risks

Studies [1]–[3] indicate that hundreds of thousands of workers die on the job each year worldwide at a staggering cost (around 4% of global gross domestic product) due to time loss, worker compensation, interruption of production, and medical expenses. Not surprisingly, data show that some work activities remain inherently dangerous, even with strict regulations in place. A study [3] from 2012 identifies cancer, respiratory disease, and accidents as the major causes of job-related deaths that could be prevented through workplace automation. In particular, a survey by the U.S. Bureau of Labor Statistics [4] shows that the most common accidents in the United States are fatal falls, collisions with objects and equipment, and injuries in confined spaces. The number of these casualties could be reduced if the physical presence of the operator were avoided.

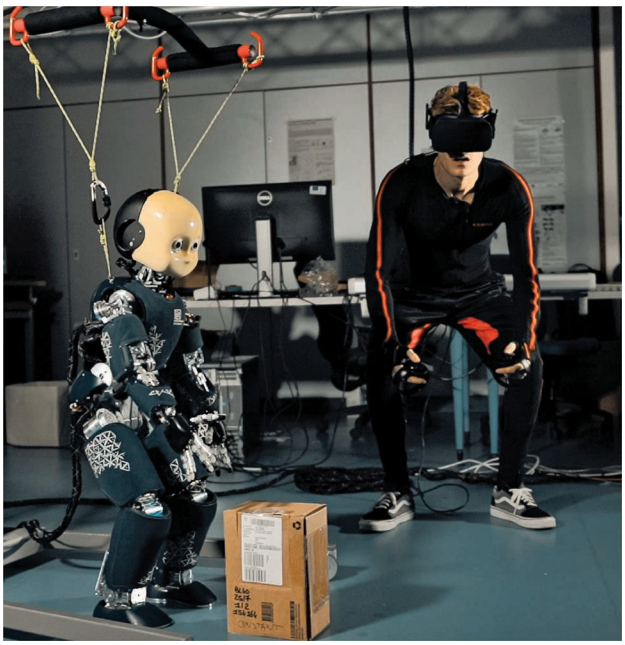


Figure 1. A human operator controlling a humanoid during a pickup task.

For example, removing asbestos roof tiles is an operation that could be performed by teleoperated robots. Currently, this task is carried out by humans in a context that is extremely risky for their health, not only because they have to move on roofs but also because they are exposed to asbestos particles [Figure 2(a)]. In the oil and gas industry, workers are often required to enter confined spaces for inspection and maintenance, exposing themselves to such hazards as toxic vapors, not to mention the difficulties of evacuating them in case of accidents [Figure 2(b)].

Humanoids at Work

In the last decade, many research projects have investigated the use of humanoids for reducing risks and human worker fatigue. In 2012, the U.S. Department of Defense launched the DARPA Robotics Challenge (DRC), a prize competition for promoting and testing humanoid applications in the context of search and rescue. Its aim was to develop semiautonomous ground robots that could perform "complex tasks in dangerous, degraded, human-engineered environments." The DRC finals also represented a midterm evaluation for WALK-MAN [5], a European Union Horizon 2020 endeavor aiming to develop a humanoid platform for autonomous or teleoperated intervention in buildings.

COMANOID [6] is another EU Horizon 2020 project focused on using humanoids to relieve human workers from tiring, dangerous, low-added-value tasks. The project revolved around the idea that humanoid robots are more suited to operate in narrow and cluttered environments typical of maintenance and manufacturing contexts, where more conventional robotic platforms like wheeled mobile manipulators would not be able to perform.

Yet another Horizon 2020 initiative, AnDy [7], addressed human–robot collaboration in industry. Among its objectives was the design of collaborative policies for humanoids and cobots to anticipate and assist the human worker. In this project, teleoperation was successfully demonstrated to be an intuitive way to convey collaborative policies to humanoids.

Proposed Approach

In all of these projects, humanoids are preferred over more conventional robotic platforms because their structure is a better fit for environments and tasks that are designed for and performed by human workers. This operational versatility makes humanoids suitable for work activities that require a variety of complex movements, such as inspection, maintenance, and interaction with human operators.

Unfortunately, flexibility and adaptability come at the cost of increased complexity. Planning and controlling tasks while maintaining balance can be a challenging endeavor. Teleoperation can ease the control complexity and facilitate the interaction with the environment. Indeed, in spite of recent progress in robot cognition based on machine-learning techniques, fully

autonomous solutions are not yet viable. Our view is that the intuition and intelligence of human operators can be leveraged to make humanoids perform complex tasks, provided that suitable control interfaces and teleoperation modes are designed.

In this article, we present a teleoperation framework for executing loco-manipulation tasks with a humanoid. The proposed control architecture provides two different modes of teleoperation:

- a high-level teleoperation setting in which the operator uses a joystick to send reference commands to the robot, such as direction and velocity of motion, without dealing with their actual execution
- a low-level teleoperation scenario in which the operator generates whole-body movements for the robot by means of a motion capture suit (motion retargeting).

In both cases, the human operator receives visual feedback through a virtual reality (VR) headset connected to the robot avatar's cameras. In this article, we demonstrate the framework for the teleoperation of the humanoid robot iCub.

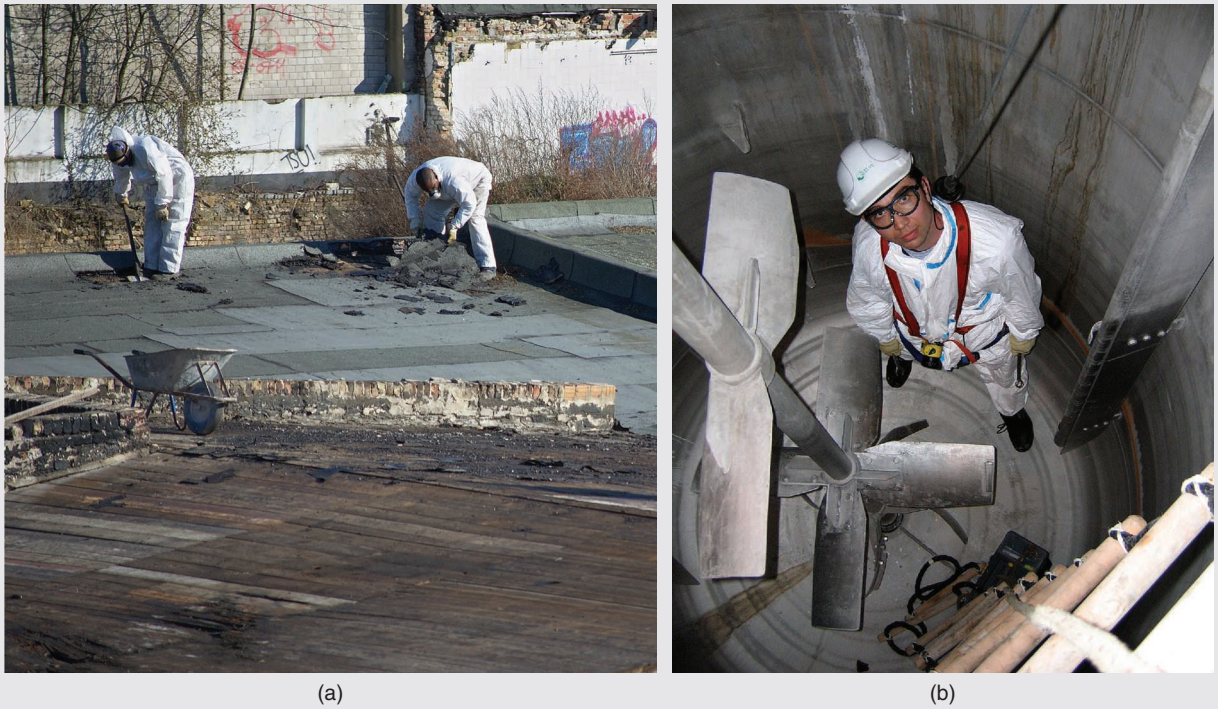
Related Work

The idea of teleoperating robots with VR was first proposed by Tachi [8], [9]. The retargeting of the upper-body joints (important for manipulation) has often been performed independently of the motion generation of the legs, which is crucial for balancing and locomotion. In [10], for example, the authors employ the mobile manipulator Justin to retarget upper-body motions with haptic feedback at the hands, without considering leg motions [Figure 3(b)].

Kim et al. [11] were among the first to extend robot teleoperation to walking motions. In [11], upper-body motions and walking are separately retargeted onto the humanoid robot Mahru by using a wearable motion capture system. Whenever the operator walks, a human-independent walking by the robot is triggered. For the retargeting of upper-body motions, only the arms are involved. Conversely, in [12], Hu et al. focus exclusively on teleoperating the walking for the humanoid TORO but also consider the human footsteps and configuration of the leg joints in the retargeting.

In the last decade, many research projects have investigated the use of humanoids for reducing risks and human worker fatigue.

In [13], the authors teleoperate the iCub robot in an immersive scenario using a VR headset and a walking platform [Figure 3(c)]. The robot starts and stops walking whenever the operator does, but the retargetable double-support motions are only limited arm movements. Motion retargeting can also be performed at the whole-body level. Ishiguro et al. [14] conducted some experiments retargeting highly dynamic upper-body and leg motions onto the humanoid robot JAXON [Figure 3(d)]. Although suitable for executing



(a) (b)

FLICKR

Figure 2. Examples of hazardous working environments where robotic avatars could replace humans: (a) an asbestos tile removal operation and (b) a maintenance operation in a confined space (in the oil and gas industry).

challenging movements, such as kicking or hitting a tennis ball with a racket, their technique cannot be used for motions like jumping, running, or walking on rough terrain.

The main challenge of teleoperating highly dynamical motions is to ensure smooth and stable motions in real time while guaranteeing the robot's balance. Inverse dynamics approaches would be ideal for handling the changing dynamics of the robot during teleoperation, but they are computationally expensive and prone to numerical ill conditioning. For this reason, the classic approaches, including the works cited previously, rely on inverse kinematics.

Similar issues must be addressed in robotic walking, where a widespread approach to generate robust dynamic motions exploits model predictive control (MPC) on reduced models of the robotic system. For gait generation, the most common strategy relies on the concept of zero moment point (ZMP), i.e., the point with respect to which the horizontal momenta of the ground reaction forces are zero. Dynamic equilibrium is guaranteed by keeping the ZMP at all times within the robot support polygon, i.e., the convex hull of the contact points.

Many successful techniques for generating stable gaits are based on a simplified linear dynamic model [15] relating the ZMP to the center of mass (CoM), derived by neglecting any rotational contribution around the CoM, which is also assumed to be at a constant height. This is called the *linear inverted pendulum* (*LIP*) or *cart-table* (*CT*) model, depending on whether the ZMP is treated as an input or an output.

In the fundamental work in [16], the CT model was used to design a linear quadratic controller with a preview. Constraints were added in [17], leading to an MPC formulation and also allowing the automatic choice of the footsteps [18]. To cope with the unstable nature of the LIP, an explicit stability constraint ensuring that the CoM trajectory is bounded with respect to the ZMP was introduced in the MPC design in [19]. Extensions for walk-to locomotion [20] or uneven ground [21] have also been proposed.

Multimode Teleoperation Framework

Our framework is illustrated in Figure 4. The human operator can choose between two different teleoperation modes using the buttons of the VR controller: 1) a low-level approach for full real-time control of the robot via motion retargeting or 2) a high-level method to command walking or preoptimized task trajectories. Both modalities share the same whole-body controller for computing in real time the commands to be sent to the robot.

In the first operational mode, the human posture is tracked by a motion tracking system—in our case, the Xsens motion capture suit. The data are then mapped to feasible corresponding joint values for the robot. To achieve dynamical balance, the references are corrected by a stabilizer and then fed to the whole-body robot controller. The latter is formalized as a multitask quadratic programming (QP) controller, where the task references are the desired robot posture and its stabilized CoM.

The second operational mode is characterized by a higher degree of robot autonomy and triggered by the operator using the joystick. For walking, the operator imparts a direction and velocity references for the humanoid gait through the analog sticks. These are translated into a timed sequence of footsteps, along with swinging foot and CoM trajectories, through an MPC-based control scheme. Alternatively, the operator uses the joystick buttons to select one of several predefined task trajectories. The corresponding trajectories, which have been preoptimized offline, are sent as references to the robot controller and then simply reproduced.

The reason for the two distinct modes is that retargeting is essentially kinematic and so not effective for online teleoperation of dynamic motions like walking or stepping. In our experience, retargeting of walking is not a viable solution, for three reasons:

1) The stride of the robot is typically shorter than the operator's.
2) The robot foot trajectory is often optimized for balance and impacts, while the trajectory retargeted from the human is not compliant with these requirements.

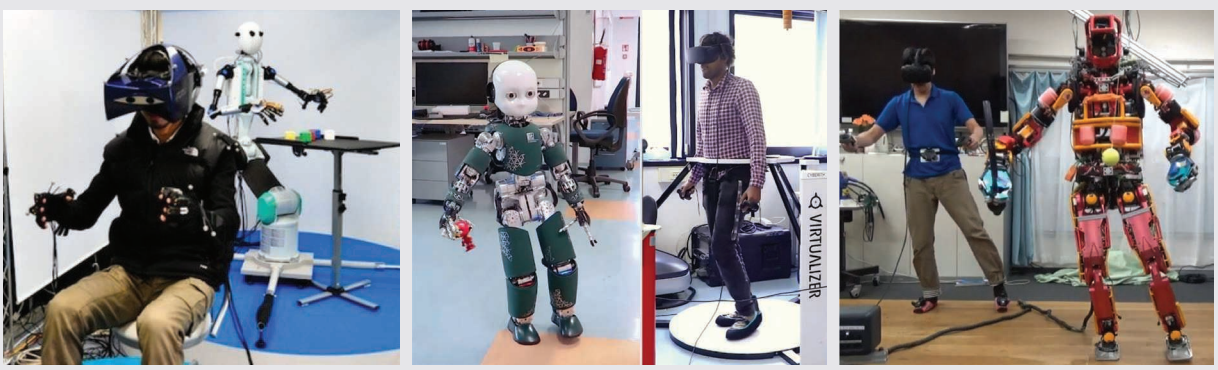


Figure 3. The state of the art in humanoid teleoperation: (a) the TELESAR II [9], a mutual telexistence system; (b) the telexistence of the iCub robot with a VR walking platform [13]; and (c) the dynamical whole-body teleoperation of the JAXON robot [14].

3) Since humanoids cannot walk as fast as humans, in a retargeting context, the operator would be forced to walk in an unnatural way, ultimately leading to inefficient robot locomotion.

For these reasons, it is better to rely on MPC-based gait generation and avoid retargeting altogether in this phase. Similar considerations can be made for many motion primitives or task trajectories that impact or leverage the robot dynamics, such as stepping or serving in tennis. In our view, these are the kind of motions that should be preoptimized offline, as they are specific to the robot dynamics. The following sections describe in detail the two teleoperation modes.

Low-Level Teleoperation Mode

The motion retargeting in the low-level teleoperation mode is built on our previous work [22]. Joint positions are measured and grouped into subcategories: head, torso, left arm, right arm, left leg, and right leg. In addition, the ground projection of the CoM, the height of the waist, the orientation of the head, and the position of the feet are controlled.

In the joint retargeting module, the Xsens skeleton's degrees of freedom are assigned to the corresponding ones of the iCub robot, as shown in Figure 5. Then, we consider the joint angle variations of the human with respect to the starting posture to compute the corresponding instantaneous values of the robot joint angles:

$$\boldsymbol{q}_R^k = \boldsymbol{q}_R^0 + (\boldsymbol{q}_H^k - \boldsymbol{q}_H^0),$$

where $\boldsymbol{q}$ is the joint positions vector, the superscripts 0 and k refer to measurements at the initial time and time k, and the subscripts H and R indicate measurements on the human and robot, respectively. The same approach is used to retarget the relative Cartesian position $\boldsymbol{p}_{BS}$ of a body segment with respect to a base link, with the difference that the variation of the human positions has to be properly scaled by the human–robot limb length ratio, as explained in [22].

To track the human CoM, we use normalized offsets, from which we then reconstruct the robot CoM ground position. We consider the ground projection of the human CoM $\boldsymbol{p}_{\text{CoM}}^g$. Its position with respect to an arbitrary foot (let us say, the left) is projected onto the line connecting the two feet. The result is then normalized to obtain an offset $o \in [0, 1]$:

$$o = \frac{(\boldsymbol{p}_{\text{CoM}}^g - \boldsymbol{p}_{l\text{Foot}}^g) \cdot (\boldsymbol{p}_{r\text{Foot}}^g - \boldsymbol{p}_{l\text{Foot}}^g)}{\|\boldsymbol{p}_{r\text{Foot}}^g - \boldsymbol{p}_{l\text{Foot}}^g\|^2},$$

where $\boldsymbol{p}_{l\text{Foot}}^g$ and $\boldsymbol{p}_{r\text{Foot}}^g$ are the ground projections of the left and right foot, respectively. When the human is in a symmetric pose, the offset o has a value around 0.5; when the human stands on a single foot, it is either zero (left foot) or one (right foot). The robot CoM ground projection is then reconstructed on the line connecting the humanoid's feet by means of this offset value. To also retarget the CoM changes that are not on the

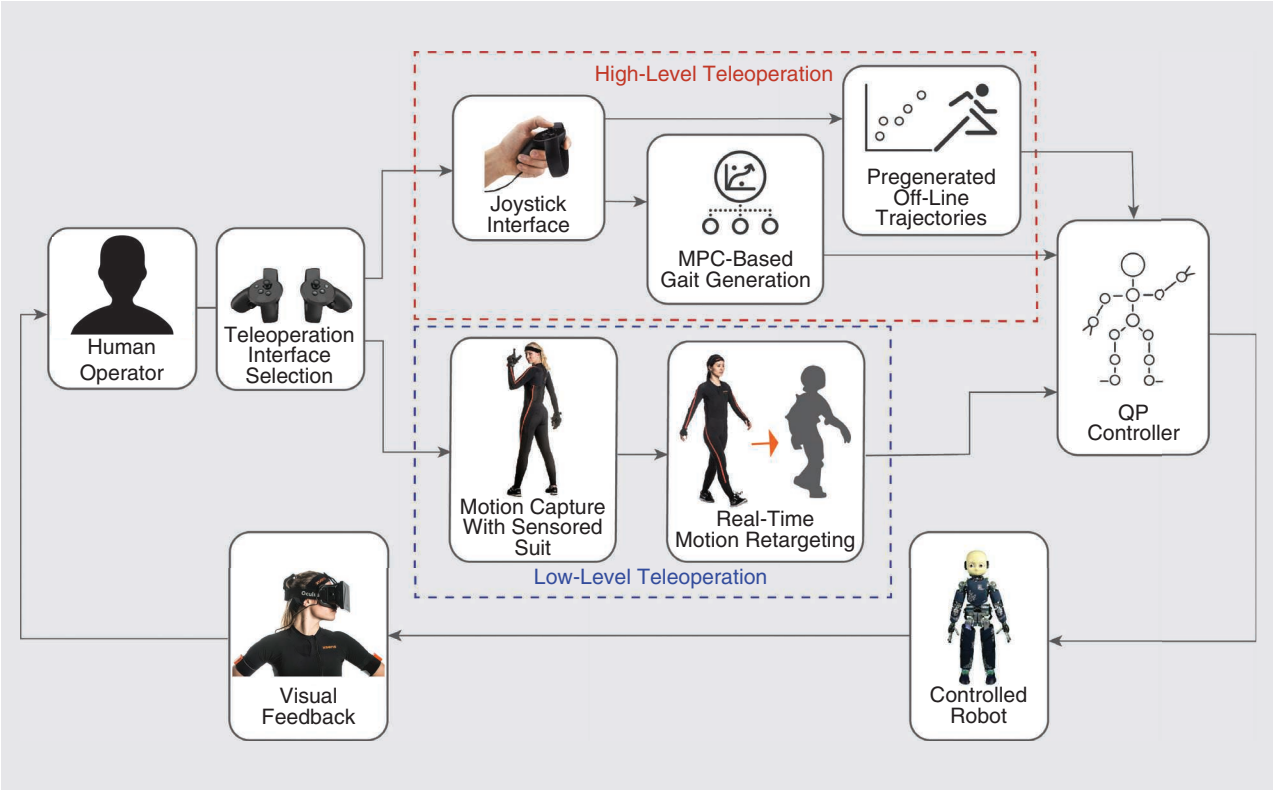


Figure 4. An overview of the proposed robot teleoperation framework. The user can choose between two modes, i.e., high- or low-level operation.

line connecting the feet, we can apply the same concept while considering the maximum backward and forward CoM displacement in the orthogonal direction of the line connecting the feet, as done in [22].

The resultant retargeted motion is not guaranteed to be dynamically balanced, and different stabilizers can be used to correct it (Figure 5). In our teleoperation approach, we want a dynamically balanced CoM trajectory, and we adopt the LIP model to properly modify the reference trajectory.

We previously recalled that balancing moments leads to the definition of the ZMP. Dynamic balance is enforced by keeping this point at all times within the robot support polygon. By neglecting rotational terms and assuming a constant

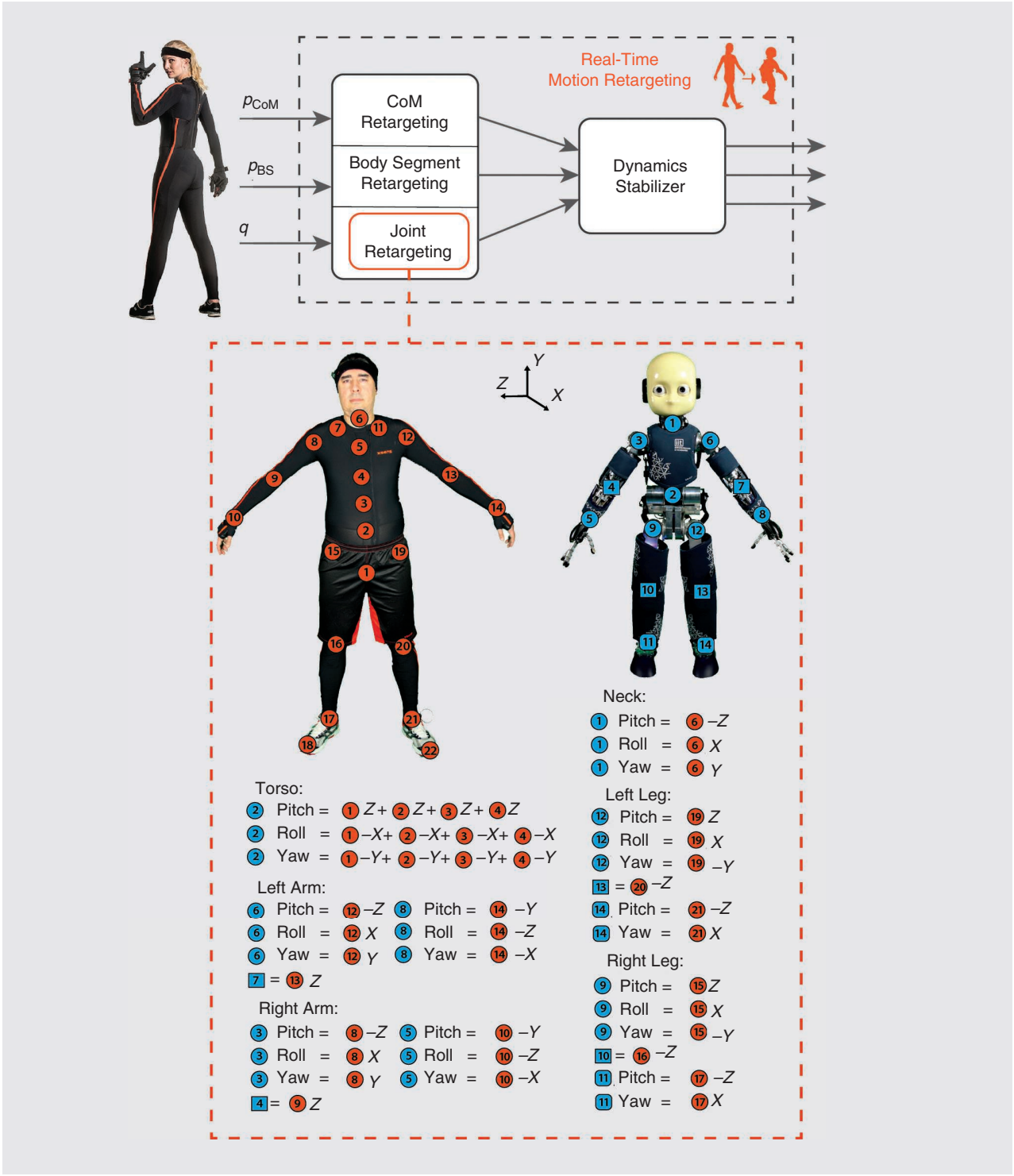


Figure 5. The pipeline of the whole-body retargeting approach.

height h_{CoM} for the CoM, the moment balance equation of the robot leads to the LIP,

$$x_{\mathrm{ZMP}} = x_{\mathrm{CoM}} - \frac{1}{\eta^2}\ddot{x}_{\mathrm{CoM}}, \tag{1}$$

where $\eta = \sqrt{g/h_{\mathrm{CoM}}}$, with g the gravitational constant while x_{CoM} and x_{ZMP} are the CoM and ZMP positions, respectively, along x (similarly for y).

By employing the dynamic equation of the LIP, it is possible to set up a QP optimization problem that provides a correction of the desired CoM at each control iteration that satisfies the humanoid's balance condition:

$$\min_{x_{\mathrm{ZMP}}} (\dot{x}^{\mathrm{ref}}_{\mathrm{CoM}} - \dot{x}_{\mathrm{CoM}})^2,$$
$$\text{subject to:} \quad \dot{x}_{\mathrm{CoM}} = \dot{x}^{m}_{\mathrm{CoM}} + \frac{\delta g}{h^{m}_{\mathrm{CoM}}}(x_{\mathrm{CoM}} - x_{\mathrm{ZMP}}),$$
$$x^{\min}_{\mathrm{ZMP}} < x_{\mathrm{ZMP}} < x^{\max}_{\mathrm{ZMP}},$$

where $\dot{x}^{\mathrm{ref}}_{\mathrm{CoM}}$ is the reference human retargeted CoM velocity; δ is the sampling time; $\dot{x}^{m}_{\mathrm{CoM}}$ and h^{m}_{CoM} are the last CoM and the last CoM height, respectively, measured from the robot; $x^{\min}_{\mathrm{ZMP}}$ and $x^{\max}_{\mathrm{ZMP}}$ are the lower and upper bounds, respectively, of the support polygon of the robot; and the first constraint is derived from (1) using the Euler approximation.

The stabilized CoM reference, Cartesian tasks, and postural tasks are set as reference tasks in the multitask QP controller. At each time step, a linearly constrained QP optimization problem is solved to minimize the given cost function characterizing the motion tracking, subject to system constraints such as the joint and torque limits. More detail about the QP controller can be found in [23].

High-Level Teleoperation Mode

As illustrated in Figure 4, the desired humanoid motion is either defined offline (pregenerated task trajectories or action primitives [24]) or online (an MPC-based gait generator). Pregenerated trajectories are triggered by the operator, depending on the situation. For example, in the final phase of the reported experiment, the robot moves its feet apart autonomously, independently of the human's lower limbs motion, to pick up the box more effectively.

Other tasks or repetitive movements like standing up can be recorded offline and replicated for a quick and precise execution. We also recommend the use of this option to make the robot perform motions that are not ergonomic for the operator or can be uncomfortable to perform. Note also that the execution of predefined motions may be more convenient in the presence of signal degradation or delay.

The online gait generation is based on the MPC scheme proposed in [19] and [25] and is summarized in Figure 6. The joystick provides reference velocities (v_x, v_y) and ω for, respectively, the sagittal, coronal, and angular motions. Footstep orientations are computed in a separate stage and used as known parameters in the next module. This is useful to guarantee the linearity of the constraints in the following MPC formulation.

To generate the footstep orientations, a first QP problem,

$$\text{(QP1):} \qquad \min_{\Theta^k_f} \sum_{j=1}^{F} (\theta^j_f - \theta^{j-1}_f - \omega T_s)^2,$$
$$\text{subject to} \quad |\theta^j_f - \theta^{j-1}_f| \le \theta_{\max},$$

is solved for all of the F predicted footsteps that fall within the prediction horizon of the MPC problem. These are denoted as $\Theta^k_f = (\theta^1_f, \ldots, \theta^F_f)$, while T_s is the step duration and $\theta_{\max}$ the maximum allowed rotation between two consecutive steps.

A second module based on MPC generates the CoM trajectory and the positions of the footsteps, while their orientations are inherited from the first module. At each time instant t_k, the MPC solves a QP problem on a prediction horizon $[t_k, t_{k+C}]$ based on a prediction model and constraints. The sampling interval has duration δ. We use as the prediction model the LIP (1) with an additional integrator on the input (dynamic extension), so that the ZMP velocities $(\dot{x}_{\mathrm{ZMP}}, \dot{y}_{\mathrm{ZMP}})$ are used as the control input. The decision variables of the QP problem are, therefore, $\dot{x}^k_{\mathrm{ZMP}}, \ldots \dot{x}^{k+C-1}_{\mathrm{ZMP}}$, $\dot{y}^k_{\mathrm{ZMP}}, \ldots \dot{y}^{k+C-1}_{\mathrm{ZMP}}$ and the footstep positions $\dot{x}^1_f, \ldots \dot{x}^F_f$, $\dot{y}^1_f, \ldots \dot{y}^F_f$, which are collected in the vector U^k of decision variables. The ZMP velocities are assumed to be constant in each sampling interval $[t_i, t_{i+1}]$, resulting in a piecewise linear ZMP.

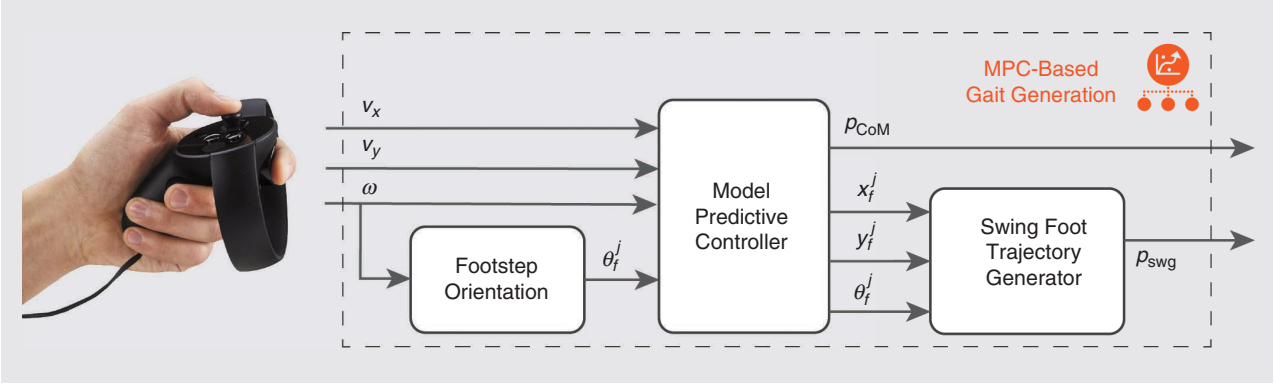


Figure 6. The pipeline of the MPC-based gait generation module.

The constraints enforced in the second QP are

- a balance constraint, which ensures that the ZMP is at all times inside the support polygon
- a kinematic constraint, guaranteeing that the footsteps are placed in a kinematically feasible region and that they avoid self-collisions
- a stability constraint, which makes sure that the generated CoM trajectory does not diverge with respect to the ZMP.

The balance or ZMP constraint at a generic instant of the single support is expressed as

$$R_j^T \begin{pmatrix} x_{\text{ZMP}}^{k+i} - x_f^j \\ y_{\text{ZMP}}^{k+i} - y_f^j \end{pmatrix} \leq \frac{1}{2} \begin{pmatrix} d_{z,x} \\ d_{z,y} \end{pmatrix}, \tag{2}$$

where $d_{z,x}$ and $d_{z,y}$ denote the size of the rectangular support polygon while $(x_{\text{ZMP}}^{k+i}, y_{\text{ZMP}}^{k+i})$ is the ZMP position at the ith prediction instant. R_j^T is the rotation matrix associated with the orientation of the jth predicted footstep within the prediction horizon, which is computed in the "Footstep Orientation" block in Figure 6. The ZMP constraint is enforced at each instant of the prediction horizon, i.e., for $i = 1, \ldots, C$.

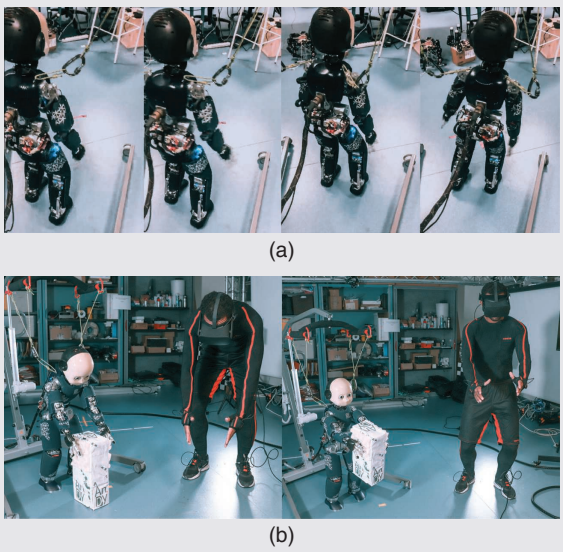


Figure 7. (a) The robot walking in the high-level mode. (b) The robot controlled in the low-level mode with a motion capture suit and a VR headset.

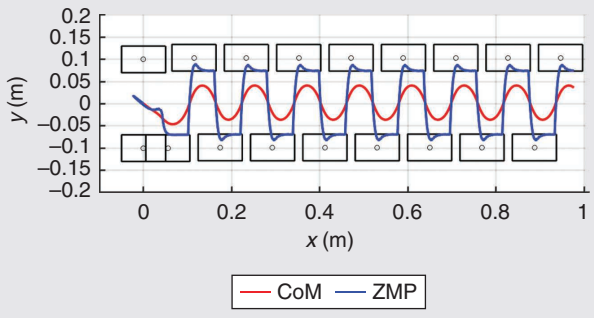


Figure 8. The walking phase: the CoM and ZMP trajectories along the MPC-generated gait.

The kinematic constraint ensures that the footsteps are placed consistently within the robot's capabilities. The constraint is

$$R_{j-1}^T \begin{pmatrix} x_f^j - x_f^{j-1} \\ y_f^j - y_f^{j-1} \end{pmatrix} \leq \pm \begin{pmatrix} 0 \\ \ell \end{pmatrix} + \frac{1}{2} \begin{pmatrix} d_{k,x} \\ d_{k,y} \end{pmatrix}, \tag{3}$$

where $d_{k,x}$ and $d_{k,y}$ represent the size of a rectangular region that is kinematically feasible and avoids self-collisions, and ℓ is the position of its center with respect to the previous footstep. The $\pm$ sign regularly alternates, discriminating between the left and right footsteps.

To understand the necessity of the stability constraint, note that the LIP model (1) and, hence, the prediction model have an unstable eigenvalue; therefore, the generic CoM trajectory will, in general, diverge with respect to the ZMP. There exists, however, a stability condition relating the CoM initial state in t_k to the future ZMP, expressed as

$$x_{\text{CoM}}^k + \frac{\dot{x}_{\text{CoM}}^k}{\eta} = \eta \int_{t_k}^{\infty} e^{-\eta(\tau - t_k)} x_{\text{ZMP}}(\tau)\, d\tau, \tag{4}$$

which ensures that the CoM trajectory remains bounded with respect to the ZMP. A similar expression holds along y.

The stability constraint of the MPC is obtained by computing (4) for a piecewise linear ZMP trajectory. However, note that the integral requires the future ZMP trajectory, which is available only up to the prediction horizon. The remaining part, after t_{k+C}, can be conjectured by using the available information beyond the prediction horizon (e.g., the planned reference velocities). One possibility is to use an infinite replication of the control inputs over the horizon, which is especially appropriate for regular gaits that exhibit a periodic behavior; the resulting stability constraint is

$$\sum_{i=0}^{C-1} e^{-i\eta\delta} \dot{x}_{\text{ZMP}}^{k+i} = \eta \frac{1 - e^{-C\eta\delta}}{1 - e^{-\eta\delta}} \left(x_{\text{CoM}}^k + \frac{\dot{x}_{\text{CoM}}^k}{\eta} - x_{\text{ZMP}}^k \right). \tag{5}$$

The second QP problem can finally be stated as

$$\begin{aligned} \text{(QP2):} \quad \min_{U^k} & \sum_{i=0}^{C-1} \left((\dot{x}_{\text{ZMP}}^k)^2 + (\dot{y}_{\text{ZMP}}^k)^2 \right. \\ & + \beta_x (\dot{x}_{\text{CoM}} - v_x \cos(\omega\delta i) + v_y \sin(\omega\delta i))^2 \\ & \left. + \beta_y (\dot{y}_{\text{CoM}} - v_x \sin(\omega\delta i) - v_y \cos(\omega\delta i))^2 \right), \end{aligned}$$

subject to

- ZMP constraints (2)
- kinematic constraints (3)
- stability constraint (5) for x and y.

In the cost function, β_x and β_y represent the weights associated with the velocity tracking terms.

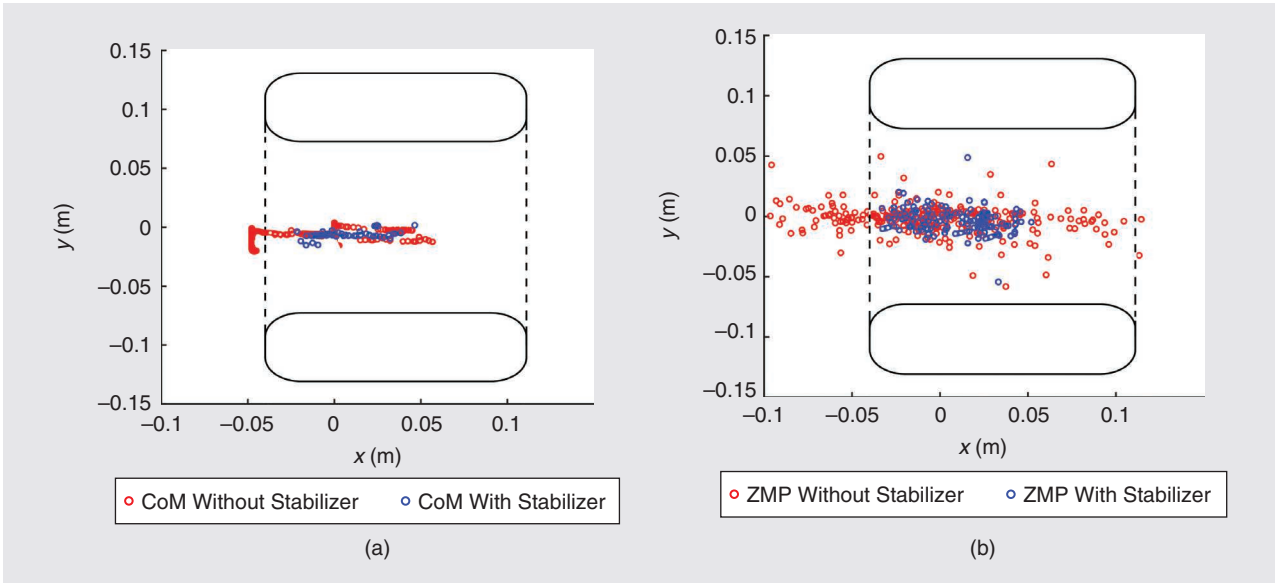


Figure 9. The pickup phase (the feet of the robot are shown in black): (a) the CoM reference position reconstructed from the human reference value (red) and the corresponding stabilized value (blue) and (b) the ZMP of the LIP model associated with the CoM reference without stabilization (red) and with stabilization (blue).

Once both QP problems are solved, the first value of the ZMP derivative $(\dot{x}^k_{\text{ZMP}}, \dot{y}^k_{\text{ZMP}})$ is used to compute the CoM trajectory $\boldsymbol{p}_{\text{CoM}}$ through the prediction model, while the first predicted footstep $(x^1_f, y^1_f, \theta^1_f)$ is employed, utilizing a predefined polynomial shape, to generate a swing foot trajectory $\boldsymbol{p}_{\text{swg}}$ ending at the predicted footstep. Both $\boldsymbol{p}_{\text{CoM}}$ and $\boldsymbol{p}_{\text{swg}}$ are finally tracked by the kinematic controller.

Experiments

This section presents an illustrative experiment performed with a human operator and the iCub humanoid robot. The iCub is only 104 cm high and cannot lift heavy weights, but our methods are not iCub specific and could be easily applied to adult-size humanoids with heavier payloads. In our presented scenario shown in Figure 7, the robot must walk, pick up a box on the floor, and hand it to the worker. The operator is equipped with a wearable motion capture Xsens MVN suit and the VR Oculus Rift system, composed of a headset and a pair of joysticks. The suit provides real-time estimation of the posture, the headset gives visual feedback from the robot cameras, and the joysticks allow the operator to switch between the two different teleoperation modes and guide the robot.

As a first step, the robot should autonomously walk to the box, guided by the operator. After the operator selects the high-level mode on the VR controller, the robot walks while receiving velocity reference commands through the joystick. Figure 8 shows the MPC-generated CoM trajectory (to be sent to the whole-body controller) together with the footsteps. When the robot arrives in front of the box, the operator stops it. Then, still in high-level mode, the operator prepares himself for the pick-up by moving his feet apart and selects a predefined motion for the humanoid to perform the same movement independently.

Finally, the operator switches to the low-level mode using the VR controller and performs a squat motion to pick up the box. Motion retargeting drives the robot to follow the movements of the operator in real time, successfully handing the box over to the human. Balance is maintained throughout this phase thanks to the stabilization performed on the retargeted human references, as shown in Figure 9. A video clip of this experiment is available online [26].

Conclusions

In this article, we proposed a multimode teleoperation framework for a humanoid robot on loco-manipulation tasks. The first mode is a low-level teleoperation of all the joints of the robot, while the other enables the execution of high-level commands and predesigned motion primitives, which can be useful for locomotion or other specific tasks.

The use case presented in the experiment consists of walking to a box on the ground using the high-level mode and then picking it up by switching to the low-level mode. It is possible to envision several scenarios in which the presented framework might be employed. For example, the operator might switch to the low-level mode to use control panels, open doors, or recover items, all actions that could be necessary during exploration. The robot might also be equipped to execute specific actions necessary for maintenance operations (e.g., tightening screws and assembling parts) using specialized tools activated from the joystick. In the future, we will test these scenarios with adult-size humanoid robots.

A current limitation of this framework that prevents us from addressing more complicated teleoperation scenarios is the absence of haptic feedback. While safe physical interaction with the environment may be ensured by the robot's low-level control, haptic feedback is still critical to enhance the remote-control capabilities of the operator. Typically, haptic feedback is localized in the end effector, where most

of the interaction occurs; in the case of teleoperated humanoids, however, whole-body haptic feedback should be considered, possibly by means of wearable vibrotactile devices.

Acknowledgment

This work was supported by the European Union's Horizon 2020 AnDy project.

References

[1] Occupational accidents in OECD countries. OECD Employment Outlook 1989, ch. 4. Organization for Economic Cooperation and Development, Paris, France. (1989). [Online]. Available: http://www.oecd.org/els/emp/3888265.pdf

[2] P. Hämäläinen, J. Takala, and K. Saarela, "Global estimates of occupational accidents," *Safety Sci.*, vol. 44, no. 2, pp. 137–156, 2006.

[3] J. Takala et al., "Global estimates of the burden of injury and illness at work in 2012," *J. Occup. Environ. Hygiene*, vol. 11, no. 5, pp. 326–337, 2014.

[4] "Census of fatal occupational injuries summary, 2017," U.S. Bureau Labor Statist., Dept. Labor, Washington, D.C., Doc. No. USDL-18-1978, Dec. 18, 2018. [Online]. Available: https://www.bls.gov/news.release/cfoi.nr0.htm

[5] WALK-MAN, "WALK-MAN: Whole-body Adaptive Locomotion and Manipulation." Accessed on: Feb. 22, 2019. [Online]. Available: https://www.walk-man.eu/

[6] COMANOID, "COMANOID: Multi-contact collaborative humanoids in aircraft manufacturing." Accessed on: Feb. 22, 2019. [Online]. Available: http://comanoid.cnrs.fr/

[7] S. Ivaldi et al., "Anticipatory models of human movements and dynamics: The roadmap of the AnDy project," in *Proc. Int. Conf. Digital Human Modeling (DHM)*, 2017, pp. 72–86.

[8] S. Tachi, *Telexistence*, 2nd ed. Singapore: World Scientific, 2015.

[9] C. L. Fernando et al., "Design of TELESAR V for transferring bodily consciousness in telexistence," in *Proc. 2012 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, Vilamoura, 2012, pp. 5112–5118.

[10] N. Y. Lii et al., "Simulating an extraterrestrial environment for robotic space exploration: The METERON SUPVIS-JUSTIN telerobotic experiment and the SOLEX proving ground," in *Proc. 13th Symp. Advanced Space Technologies Robotics and Automation (ASTRA),* European Space Agency/European Space Research and Technology Centre (ESA/ESTEC), Noordwijk, The Netherlands, May 2015. [Online]. Available: https://elib.dlr.de/98352/1/Lii-SupvisJustinSolexAstra2015_v23e.pdf

[11] D. Kim, B.-J. You, and S.-R. Oh, *Whole Body Motion Control Framework for Arbitrarily and Simultaneously Assigned Upper-Body Tasks and Walking Motion*. Berlin: Springer-Verlag, 2013, pp. 87–98.

[12] K. Hu, C. Ott, and D. Lee, "Online human walking imitation in task and joint space based on quadratic programming," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2014, pp. 3458–3464.

[13] M. Elobaid, Y. Hu, G. Romualdi, S. Dafarra, J. Babic, and D. Pucci, "Telexistence and teleoperation for walking humanoid robots," in *Proc. SAI Intelligent Systems Conf. IntelliSys 2019: Intelligent Systems and Applications,* pp. 1106–1121.

[14] Y. Ishiguro et al., "High speed whole body dynamic motion experiment with real time master-slave humanoid robot system," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2018, pp. 1–7. doi: 10.1109/ICRA.2018.8461207.

[15] S. Kajita, H. Hirukawa, K. Harada, and K. Yokoi, *Introduction to Humanoid Robotics*. Berlin: Springer-Verlag, 2014.

[16] S. Kajita et al., "Biped walking pattern generation by using preview control of zero-moment point," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2003, pp. 1620–1626.

[17] P.-B. Wieber, "Trajectory free linear model predictive control for stable walking in the presence of strong perturbations," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2006, pp. 137–142.

[18] A. Herdt, H. Diedam, P.-B. Wieber, D. Dimitrov, K. Mombaur, and M. Diehl, "Online walking motion generation with automatic footstep placement," *Adv. Robot.*, vol. 24, no. 5-6, pp. 719–737, 2010.

[19] N. Scianca, M. Cognetti, D. De Simone, L. Lanari, and G. Oriolo, "Intrinsically stable MPC for humanoid gait generation," in *Proc. 16th IEEE-RAS Int. Conf. Humanoid Robots*, 2016, pp. 101–108.

[20] A. Aboudonia, N. Scianca, D. De Simone, L. Lanari, and G. Oriolo, "Humanoid gait generation for walk-to locomotion using single-stage MPC," in *Proc. 17th IEEE-RAS Int. Conf. Humanoid Robots*, 2017, pp. 178–183.

[21] A. Zamparelli, N. Scianca, L. Lanari, and G. Oriolo, "Humanoid gait generation on uneven ground using Intrinsically Stable MPC," in *Proc. 12th IFAC Symp. Robot Control*, 2018, pp. 393–398.

[22] L. Penco et al., "Robust real-time whole-body motion retargeting from human to humanoid," in *Proc. IEEE-RAS 18th Int. Conf. Humanoid Robots*, 2018, pp. 425–432.

[23] E. M. Hoffman, A. Rocchi, A. Laurenzi, and N. G. Tsagarakis, "Robot control for dummies: Insights and examples using OpenSoT," in *Proc. IEEE-RAS 17th Int. Conf. Humanoid Robots*, 2017, pp. 736–741.

[24] J. Kober and J. Peters, "Learning motor primitives for robotics," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2009, pp. 2112–2118.

[25] N. Scianca, D. De Simone, L. Lanari, and G. Oriolo, MPC for humanoid gait generation: Stability and feasibility. 2019. [Online]. Available: https://arxiv.org/abs/1901.08505

[26] "RAM videos," YouTube, Sept. 2019. [Online]. Available: https://www.youtube.com/playlist?list=PLaViAl2WLPMfDt-w2s0Qz7yl_e-JEWNSJ

Luigi Penco, National Institute for Research in Computer Science and Automation, Nancy, France. Email: luigi.penco@inria.fr.

Nicola Scianca, Department of Computer, Control, and Management Engineering, Sapienza University of Rome, Italy. Email: scianca@diag.uniroma1.it.

Valerio Modugno, Department of Computer, Control, and Management Engineering, Sapienza University of Rome, Italy. Email: modugno@diag.uniroma1.it.

Leonardo Lanari, Department of Computer, Control, and Management Engineering, Sapienza University of Rome, Italy. Email: lanari@diag.uniroma1.it.

Giuseppe Oriolo, Department of Computer, Control, and Management Engineering, Sapienza University of Rome, Italy. Email: oriolo@diag.uniroma1.it.

Serena Ivaldi, National Institute for Research in Computer Science and Automation, Nancy, France. Email: serena.ivaldi@inria.fr.