Peer-reviewed version available at *Appl. Sci.* **2018**, *8*, 2005; doi:10.3390/app8102005

Article

Real-Time Whole-Body Imitation by Humanoid Robots and Task-Oriented Teleoperation Using an Analytical Mapping Method and Quantitative Evaluation

Zhijun Zhang [1,‡*], Yaru Niu [1,‡*], Ziyi Yan [1,] and Shuyang Lin [1,]

[1] South China University of Technology; auzjzhang@scut.edu.cn, auchrisniu@mail.scut.edu.cn
* Correspondence: auzjzhang@scut.edu.cn, auchrisniu@mail.scut.edu.cn
‡ These authors contributed equally to this work.

Abstract: Due to the limitations on the capabilities of current robots regarding task learning and performance, imitation is an efficient social learning approach that endows a robot with the ability to transmit and reproduce human postures, actions, behaviors, etc., as a human does. Stable whole-body imitation and task-oriented teleoperation via imitation are challenging issues. In this paper, a novel comprehensive and unrestricted real-time whole-body imitation system for humanoid robots is designed and developed. To map human motions to a robot, an analytical method called geometrical analysis based on link vectors and virtual joints (GA-LVVJ) is proposed. In addition, a real-time locomotion method is employed to realize a natural mode of operation. To achieve safe mode switching, a filter strategy is proposed. Then, two quantitative vector-set-based methods of similarity evaluation focusing on the whole body and local links, called the Whole-Body-Focused (WBF) method and the Local-Link-Focused (LLF) method, respectively, are proposed and compared. Two experiments conducted to verify the effectiveness of the proposed methods and system are reported. Specifically, the first experiment validates the good stability and similarity features of our system, and the second experiment verifies the effectiveness with which complicated tasks can be executed. At last, an imitation learning mechanism in which the joint angles of demonstrators are mapped by GA-LVVJ is presented and developed to extend the proposed system.

Keywords: Humanoid robot; whole-body imitation; social learning; motion mapping; teleoperation for tasks; similarity evaluation

1. Introduction

Robots can be used in place of humans to perform remote tasks such as remote meetings/interactions[1,2], telesurgery[3–6], rehabilitation[7], and search and rescue[8]. However, the ability to control a humanoid robot with a high degree of freedom using a traditional controller or through preprogramming based on human experience strongly depends on the operation skills of the users, the programming skills of the developers, and the availability of extensive experience. As an alternative, imitation is an efficient social learning approach that endows a robot with the ability to transmit and reproduce human postures, actions, behaviors, etc., as a human does. The questions of how to achieve accurate, stable, and complete whole-body imitation and how to evaluate the imitation similarity are two basic issues facing this approach.

Generally, robotic imitation consists of three phases, i.e., observation, representation and reproduction [9]. The observation process usually relies on either marker-based or markerless visual capture systems or dynamic capture systems to acquire human motions. In a marker-based visual capture system, markers must be placed on the demonstrators to indicate the positions of their body parts[10–14], whereas a markerless visual capture system can directly extract 2D or 3D information from video frames[15–20]. The imitation system presented in [21] obtained motion data using the Xsens

MVN dynamic capture system, which offers high precision but is inconvenient and expensive. Our real-time imitation system uses the Microsoft Kinect II, which costs much less and is more user-friendly, to capture 3D skeleton points and track the faces of the demonstrators.

In the representation process, the observed actions are mapped to the internal representation of the robot. The traditional approaches for posture mapping are numerical methods of inverse kinematic (IK) computations [10,16,17,21–24]. The numerical IK method is effective for manipulating the end effectors of robots in the work space, but it is a relatively indirect and imprecise means of producing motions in joint space, and it incurs a rather high computational cost. Because the joint angles determine the posture of the robot, a precise mapping in joint space will be beneficial for the imitation similarity. To enable the generation of more human-like actions with less computation, new methods will be necessary.

To this end, several analytical methods have been exploited [13,14,19,20,25–29]. In addition to the positions of the end effectors, other information such as the positions of the joints is obtained in these methods. An inherent advantage of analytical methods with constraints is that the solutions generated using such analytical methods are more direct, accurate and efficient. Consequently, it is easier to obtain accurate joint angles for mapping that correspond to an observed posture and thus to achieve a high imitation similarity. However, most existing geometrical analytical mapping methods used in imitation learning cannot obtain the specific orientations of each "body link" in a demonstrated human posture [13,20,25–28]. In addition, the details of the robot's joint structures, such as mother and child joint features, are not fully considered when calculating the joint angles for mapping, and the imitation similarity of the generated postures suffers as a result. Some researchers have directly used the rotation matrices provided by the Kinect SDK or the Open NI library, which have been proven to be unstable and imprecise [19,25]. Using an approach different from most existing methods, a novel analytical method called GA-LVVJ is proposed in this paper, which can achieve a high imitation similarity with a low computational cost.

Upper-body imitation has been studied for many years [10,12,14,15,19,22,30–36]. Compared with upper-body imitation, whole-body imitation is more comprehensive and complicated. Whole-body imitation usually involves multi-mode lower-body imitation and balance control [13,16–18,20,21,23–26,37]. Kondo et al. proposed a method of joint angle generation using a particle filter, which was based on a designed similarity function [18]. However, the particle filter could not ensure standing stability, and consequently, a robot controlled in this way was limited to a doubly supported posture (standing on two feet). Hwang et al. studied upper-body and lower-body imitation separately to ensure stability[16]. The postures were estimated by means of neural-network-based inverse kinematics and 5 feature points extracted from a stereo-vision system, but only 11 preset stable motions were available for the lower body. Lin et al. modified key postures via a Q-learning process to maintain a robot's balance during imitation[38]. It was necessary to train the balance control process in advance, and only limited numbers of joint angles could be modified independently using several fixed preset values. In [25], the HipRoll joint of the swing leg was locked during the single-support phase, and the unstable imitation of the lower-body motion sometimes led to falling down. Lei et al. presented a method of maintaining whole-body balance when imitating single poses or a pose sequence[26]. However, the landing of the foot when transitioning from the single-support to the double-support phase was not considered. Moreover, walking and hand detection, which are essential for task execution, also were not taken into account. In [23], Ou et al. proposed a real-time imitation system based on the Kinect. The system derived robot joint angles by applying the Levenberg-Marquardt algorithm to the IK optimization problem and achieved both double and single support by means of an "Ankle Strategy" [37] and a "Hip Strategy". Liu et al. also adjusted the ankle joints to maintain balance [13]. Still, walking and hand detection were not considered.

Walking must be considered for true whole-body imitation. Koenemann et al. used the Xsens MVN motion capture system to achieve stable whole-body imitation via the numerical inverse kinematics method [21]. The end effector was retargeted if the posture was not stable. The system enabled the robot to imitate a human gait at a low velocity, but the demonstrator was required to lift his or her leg slowly and high, which is inconvenient and inconsistent with spontaneous human walking motions. Poubel et al. focused on the problem of shifting the support leg and lifting/lowering the foot during motion imitation via task specification [17,24]. These authors avoided using motion primitives, and the robot acted in a freer and more human-like manner when shifting the support leg, stepping and walking. However, the locomotion functionality remained limited because of the low stepping velocity. Using a different approach from these methods, some researchers have controlled the locomotion of a robot by using preprogrammed walking primitives to make it easier for the demonstrator to operate the robot. The system developed in [39] used a depth camera to record the positions of the human demonstrator and calculate the displacements, but it required a Wiimote as ancillary equipment to tell the system when to record the initial and final positions. In [40], a humanoid robot was made to imitate human upper-limb motions and was issued walking, hand closing and opening, and mode switch instructions via speech signals to complete a pen shopping task. Almetwally et al. used additional postures to switch on/off the locomotion function and successfully directed a robot to pick up a ball using upper-limb imitation [29,41]. The system had only one static reference position for locomotion, and the additional postures were not spontaneous human motions; consequently, the operation was relatively time-consuming and inconvenient. In [25], an imitation system was used to issue walking instructions by means of upper-limb postures. The volunteers participating in this research verified the difficulty of this method.

The evaluation of the motion similarity between a human demonstrator and a humanoid robot is another critical issue for both the optimization of multiple solutions and the assessment of the effectiveness of an imitation system. For solution optimization, in [23], the imitation similarity was measured by calculating the distance between the normalized vectors representing the human and the robot, and the corresponding error function was applied in the IK optimization problem. Alissandrakis et al. presented state, action and effect metrics for different correspondence mappings associated with various imitators and imitation tasks [42]. For assessing the effectiveness of an imitation system, Zuher et al. used an evaluation metric called the Mean Opinion Score to quantify the imitation quality [25]. However, this approach was subjective and not sufficiently accurate. In [16,17,21,24], the researchers evaluated the imitation similarity by determining the position errors of the end effectors between the scaled human model and the robot; however, this approach could not completely reflect the similarity of the postures because the end effectors represent only a few points on the entire body. In [26], the author presented a quantitative metric of the similarity between a human pose and a robot pose based on the latent spaces of the human and the robot; the poses were represented by joint angles, which also provide an indirect representation of posture. With regard to metrics that are not specifically aimed at imitation systems, Zhang et al. evaluated the human-like behavior of the Nadine robot by calculating the Pearson product-moment correlation coefficients of key points captured by a depth camera between the humanoid robot and the human demonstrator [43]. Harada et al. constructed a quantitative evaluation method for pose and motion similarity in accordance with human perception[44]. The weights of the body regions were optimized to ensure the highest correlation coefficients between the quantitative similarities and average human intuition. The quantitative evaluation was based on the positions and angles of the body regions, which do not directly represent the posture and do not consider the different effects of different joints on the posture. To improve the evaluation of imitation similarity, two quantitative vector-based similarity evaluation criteria are proposed in this paper.

Overall, the constraints of the previous research on robotic imitation are as follows:

- Most imitation systems that can achieve stable whole-body motions are based on numerical IK methods for posture mapping. However, joint angles generated using numerical methods are not as accurate as those obtained using analytical methods and incur higher computation costs.
- Most analytical methods of joint angle mapping used in imitation systems are simple and rough, and they cannot indicate the orientations of human body links and lack sufficient consideration

of the robot's joint structures. This will affect the accuracy of the joint angles for mapping and limit the imitation similarity.

- Many approaches cannot achieve stable doubly supported and singly supported imitation.
- Most researchers have not considered the imitation of head motions, hand motions or locomotion, which are essential for task execution using an imitation system. Some systems partially accomplish these functions but require additional postures or audio instructions or require ancillary handheld or wearable devices, which are inconvenient.
- Most similarity criteria are based on the positions of key points, i.e., the positions of end effectors or skeleton points, or are based on joint angles and consequently cannot directly reflect the posture similarity.

To solve the problems identified above and improve the performance of real-time imitation systems, a novel stable, complete and unrestricted whole-body imitation system is proposed and developed in this paper. In addition, vector-set-based and multi-focused methods of imitation evaluation are proposed and employed to evaluate our real-time imitation system. The main contributions of this paper can be summarized as follows:

- A novel comprehensive and unrestricted whole-body imitation system is proposed and developed. In addition, an imitation learning algorithm is presented and developed based on it. To the best of our knowledge, it is the most complete and free whole-body imitation system developed to date, enabling the imitation of head motions, arm motions, lower-limb motions, hand motions and locomotion and not requiring any ancillary handheld or wearable devices or any additional audio or gesture-based instructions. The system includes a double-support mode, a single-support mode and a walking mode. The balance is controlled in each mode, and the stability of the system enables the robot to execute some complicated tasks in real time.
- A novel analytical method called GA-LVVJ is proposed for the accurate mapping of human motions to a robot based on the observed human data. Link vectors are constructed according to the captured skeleton points, and the virtual joints are set according to the link vectors and the robot joints. Then, the frame of each human body link is built to indicate its orientation and posture. A structural analysis of child and mother joints is employed for the calculation of the joint angles.
- A real-time locomotion method is proposed for the walking mode. Both the rotations and displacements of the human body are calculated in real time. No ancillary equipment is required to issue instructions for recording the position, and no fixed point is needed to serve as a reference position.
- A filter strategy is proposed and employed to ensure that the robot transitions into the correct motion mode and that its motions are stabilized before the mode changes.
- Two quantitative vector-based similarity evaluation methods are proposed, namely, the WBF method and the LLF method, which focus on the whole-body posture and the local-link posture, respectively. They can provide novel metrics of imitation similarity that consider both whole-body and local-link features.

2. System Framework

The framework of the proposed whole-body imitation system is shown in Figure 1. The whole framework includes four stages, i.e., the observation stage, the motion mapping stage, the mode switch stage and the reproduction stage. In the observation stage, human motion information are captured by the camera. Then in the motion mapping stage, the human skeleton model is constructed, and the human motions are mapped to the robot body by using the proposed joint mapping (through GA-LVVJ), head motion mapping, hand motion mapping and locomotion mapping methods. It is worth pointing out that since the information obtained from the skeleton data is limited and insufficient, auxiliary approaches (in addition to the GA-LVVJ method) for the head motions and hand motions of a humanoid robot are employed. In the mode switch stage, the robot will switch to the double-support mode, the single-support mode or the walking mode according to the state of the human skeleton model. In the reproduction stage, if the robot is in the double-support or the single-support mode, the mapped joint angles will be adjusted lightly through the balance control and then produce stable whole-body imitation motions. If the robot switches to the locomotion mode, the mapped locomotion data will be applied to replicate human rotations and displacements.

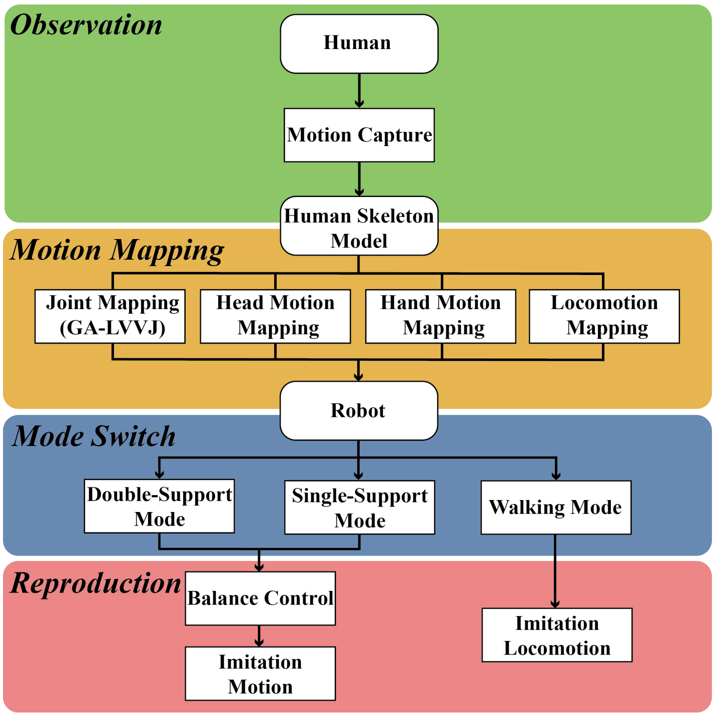


Figure 1. The framework of the proposed whole-body imitation system.

3. Motion Mapping

To realize whole-body imitation, the motions of a human demonstrator need to be mapped to a robot. Specifically, we consider joint mapping (through GA-LVVJ), head motion mapping, hand motion mapping and locomotion mapping in the proposed system. In this section, the methods of motion mapping from the human to the robot are described in detail.

3.1. Motion Loop and Locomotion Loop

Human motion data are consecutively acquired by an RGB-D camera. In our system, a motion loop includes the processing of the motion data from one frame and a series of execution orders in accordance with the corresponding motion data. A locomotion loop includes multiple consecutive motion loops; the number of motion loops included depends on the duration of one motion loop, which is uncertain for each one. At the start of each locomotion loop, the position of the demonstrator is compared with his or her position from the last locomotion loop. If the conditions for locomotion are satisfied, the robot will transition into the walking mode and walk to cover the corresponding displacement and rotation angle.

3.2. Construction of Human Skeleton Model And Joint Mapping: GA-LVVJ

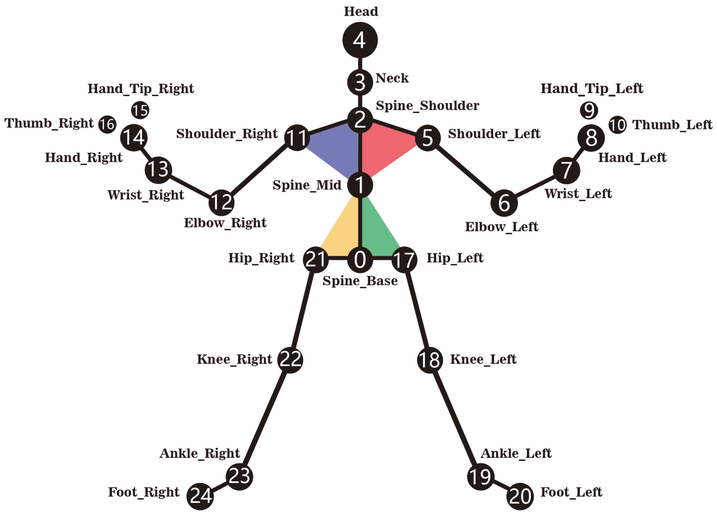


Figure 2. Names and numbers of skeleton points captured by the Kinect II.

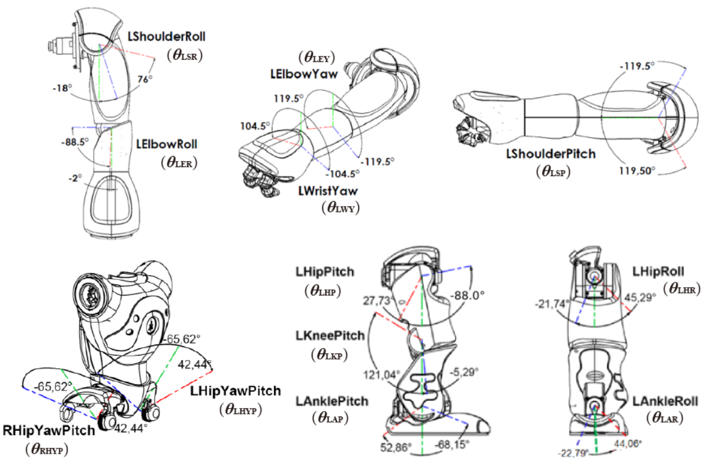


Figure 3. Upper- and lower-limb joints and joint angles of the Nao robot. [45]

The GA-LVVJ method is proposed for mapping human postures to a robot by building the human skeleton model and extracting joint angles. Without loss of generality, GA-LVVJ includes the following steps:

- Construction of human skeleton link vectors in accordance with the captured 3D points;
- Construction of human virtual joints in accordance with the human skeleton link vectors and robot joints;
- Establishment of local link frames for the human skeleton model with reference to the virtual joints;
- Calculation of the joint angles of the human skeleton model using the local link frames and link vectors;
- Application of the joint angles of the human skeleton model to the robot.

In our system, a Kinect II RGB-D camera is used to capture 3D skeleton points of the human body, which include points corresponding to the end effectors and joints. Hence, a complete skeleton model of the human body can be constructed based on these points, the numbers of which are shown in

Figure 2. From these skeleton points, we can obtain the link vectors $\vec{V}_{i-j}$ formed by pairs of skeleton points. The subscript i denotes the number of the skeleton point at which the vector begins, and j denotes the number of the skeleton point to which the vector points. It is worth pointing out that the proposed GA-LVVJ method is a general mapping method which can be applied to most types of humanoid robots. Without loss of generality, the NAO robot is used as an example here. Figure 3 depicts the joints and corresponding joint angles of Nao.

In this paper, if two joints or two links are on the same limb chain, the one closer to the torso is defined as the mother joint or mother link, whereas the other one is defined as the child joint or child link. A link frame is attached to each link and indicates the link's position and orientation, the link frames of Nao are shown in Figure 4. The origin of a link frame is located at the intersection between the link itself and its child link or at the center of mass of a link.

In order to map human postures to the robot through a geometrical analysis of the joint angles, the joint structures and each link frame of the human skeleton model should be established. The link frames of the human skeleton model serve as the references for the calculation of the joint angles for mapping. In order to construct them, virtual joints of the human skeleton model similar to those of the robot are established, as shown in Figure 4, regardless of what the real joint structures of the human body are. The axis of each joint is set collinear with or vertical to the vector of the mother link in the initial posture. Then, the link frames should be set in accordance with the axes of the joints at each intersection between a link and its child link. The origin of a link frame is located at the intersection between the link itself and its child link, as depicted in Figure 4. The postures of the robot and the human skeleton model that are shown in Figure 4 are defined as the initial postures in this paper. The LShoulderPitch and RShoulderPitch angles are equal to $\frac{\pi}{2}$, whereas all other joint angles in the initial posture are zero. These joint angles are defined as the initial joint angles.

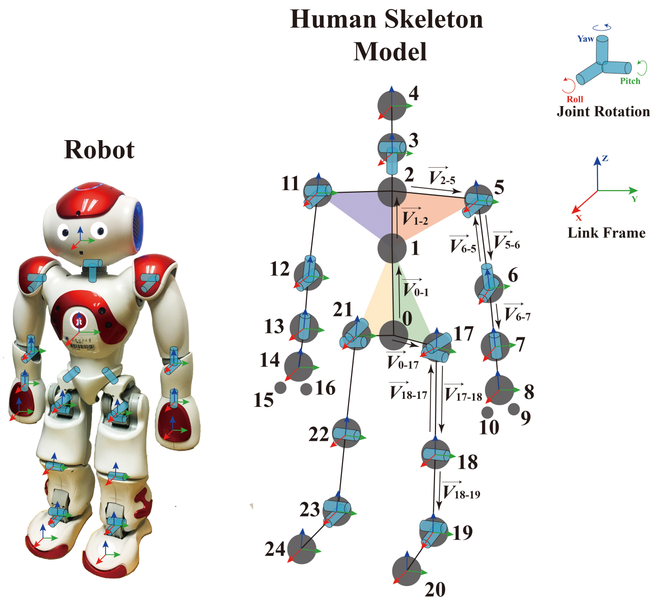


Figure 4. Joint structures and link frames of the robot and the human skeleton model.

To improve the accuracy of the joint mapping, when extracting the angle of the mother joint between two link vectors, the child link vector should be first projected on the plane vertical to the coordinate axis which represents the rotation axis of the mother joint, then this joint angle can be calculated through the angle between the projected vector and another vector on this plane. When extracting the angle of the child joint between two link vectors, the joint angle can be obtained through the angle between the child link vector and the same plane mentioned above. If there is only one joint between two link vectors, it can be regarded as a mother joint.

It is worth noting that since the human torso is not rigid, skeleton points 0, 1, 2, 5, 11, 17 and 21 in the human skeleton model may not all be in the same plane. As shown in Figure 4, to obtain more accurate joint angles, the torso of the human skeleton model is divided into four parts, i.e., the left upper torso, the right upper torso, the left lower torso and the right lower torso. Compared with a single-frame method (such as a method based on a single torso frame [28] or a fixed world frame [13,27]), the proposed method of torso segmentation yields more precise joint angles for the human skeleton model and improves the imitation similarity for local postures.

Next, the process of obtaining several key joint angles for the left side of the body will be presented as an example of GA-LVVJ.

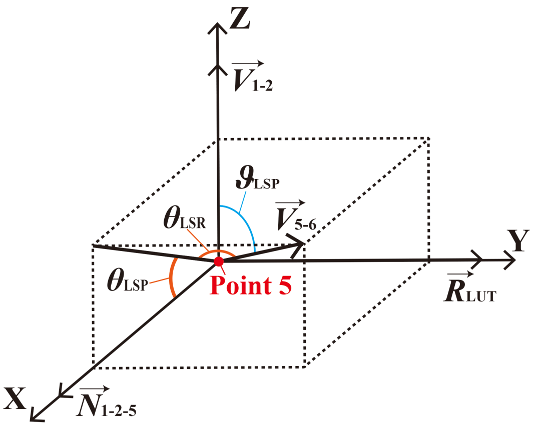


Figure 5. The link frame of the left upper torso of the human skeleton model and the shoulder joint angles for mapping.

(1) Joints of Shoulder_Left: The angles of shoulder joints should be calculated with reference to upper torso links. The axis of LShoulderPitch is collinear with the vector $\vec{V}_{2-5}$. Because LShoulderRoll is the child joint of LShoulderPitch, its axis changes with the rotation of LShoulderPitch and is collinear with the the normal vector to Plane 2-5-6, $\vec{N}_{2-5-6}$, which is calculated as

$$\vec{N}_{2-5-6} = \vec{V}_{2-5} \times \vec{V}_{6-5}. \tag{1}$$

When LShoulderPitch is at its initial angle, the axis of LShoulderRoll is collinear with the normal vector to the left upper torso plane, $\vec{N}_{1-2-5}$, i.e.,

$$\vec{N}_{1-2-5} = \vec{V}_{2-5} \times \vec{V}_{1-2}. \tag{2}$$

Then, $\vec{N}_{1-2-5}$ is translated to the origin of the link frame of the left upper torso, point 5, and the X axis of the frame is set collinear with $\vec{N}_{1-2-5}$, as depicted in Figure 5. Because the axis of LShoulderPitch is vertical to the torso link in Nao but may not be in the human skeleton model, $\vec{V}_{1-2}$, which represents the upper torso link, rather than $\vec{V}_{2-5}$, which is aligned with the axis of LShoulderPitch in the human skeleton model, is translated to point 5 and set collinear with the X axis of the link frame. $\vec{V}_{1-2}$ is collinear with the Z axis. Then, a reference vector $\vec{R}_{LUT}$ is introduced and set collinear with the Y axis to complete a right-handed frame; this vector is obtained as follows:

$$\vec{R}_{LUT} = \vec{V}_{1-2} \times \vec{N}_{1-2-5}, \tag{3}$$

where the subscript LUT is an abbreviation for "left upper torso".

As previously discussed, LShoulderPitch is the mother joint of LShoulderRoll. To achieve a more similar local posture of the left upper arm relative to the left upper torso as the reference, which means that the orientation of the left upper arm link in the frame of the left upper torso link of the robot should be consistent with that in the human skeleton model, the geometrical forms of the angles of these two adjacent joints (i.e., LShoulderPitch and LShoulderRoll) should be distinguished. As illustrated in Figure 5, the angle of LShoulderPitch, θ_{LSP}, should be calculated in accordance with the angle between $\vec{N}_{1-2-5}$ and the projection of $\vec{V}_{5-6}$ onto the X-Z plane, which is also the angle between the X-Y plane and the plane formed by $\vec{R}_{\mathrm{LUT}}$ and $\vec{V}_{5-6}$, i.e.,

$$|\theta_{\mathrm{LSP}}| = \arccos\left(\frac{\vec{V}_{1-2} \cdot (\vec{R}_{\mathrm{LUT}} \times \vec{V}_{5-6})}{|\vec{V}_{1-2}||\vec{R}_{\mathrm{LUT}} \times \vec{V}_{5-6}|}\right). \tag{4}$$

To obtain θ_{LSP}, an intermediate variable ϑ_{LSP} is defined as follows:

$$\vartheta_{\mathrm{LSP}} = \arccos\left(\frac{\vec{V}_{5-6} \cdot \vec{V}_{1-2}}{|\vec{V}_{5-6}||\vec{V}_{1-2}|}\right). \tag{5}$$

Then, θ_{LSP} is obtained as

$$\theta_{\mathrm{LSP}} = \begin{cases} -|\theta_{\mathrm{LSP}}|, \text{ if } \vartheta_{\mathrm{LSP}} \leq \frac{\pi}{2}, \\ |\theta_{\mathrm{LSP}}|, \text{ if } \vartheta_{\mathrm{LSP}} > \frac{\pi}{2}. \end{cases} \tag{6}$$

The angle of LShoulderRoll, θ_{LSR}, should be calculated from the angle between $\vec{V}_{5-6}$ and the X-Z plane, i.e.,

$$\theta_{\mathrm{LSR}} = \frac{\pi}{2} - \arccos\left(\frac{\vec{V}_{5-6} \cdot \vec{R}_{\mathrm{LUT}}}{|\vec{V}_{5-6}||\vec{R}_{\mathrm{LUT}}|}\right). \tag{7}$$

(2) Joints of Elbow_Left: The axis of LElbowYaw is collinear with the vector $\vec{V}_{6-5}$. Because LElbowRoll is the child joint of LElbowYaw, its axis changes with the rotation of LElbowYaw and is collinear with the normal vector to Plane 5-6-7, $\vec{N}_{5-6-7}$, which is calculated as

$$\vec{N}_{5-6-7} = \vec{V}_{6-5} \times \vec{V}_{6-7}. \tag{8}$$

When LElbowYaw is at its initial angle, the axis of LElbowRoll is collinear with $\vec{N}_{\text{2-5-6}}$. As depicted in Figure 6, $\vec{N}_{2-5-6}$ and $\vec{V}_{6-5}$ are translated to the origin of the link frame of the left upper arm, point 6, and the X and Z axes of the frame are set collinear with $\vec{N}_{2-5-6}$ and $\vec{V}_{6-5}$, respectively. Then, a reference vector $\vec{R}_{\mathrm{LUA}}$ is set collinear with the Y axis to complete a right-handed frame; this vector is obtained as follows:

$$\vec{R}_{\mathrm{LUA}} = \vec{V}_{6-5} \times \vec{N}_{2-5-6}, \tag{9}$$

where the subscript LUA is an abbreviation for “left upper arm”.

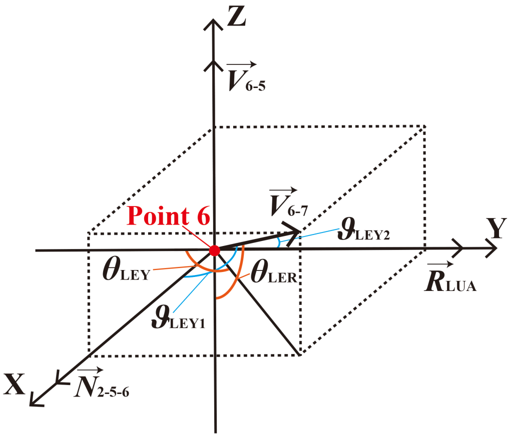


Figure 6. The link frame of the left upper arm of the human skeleton model and the elbow joint angles for mapping.

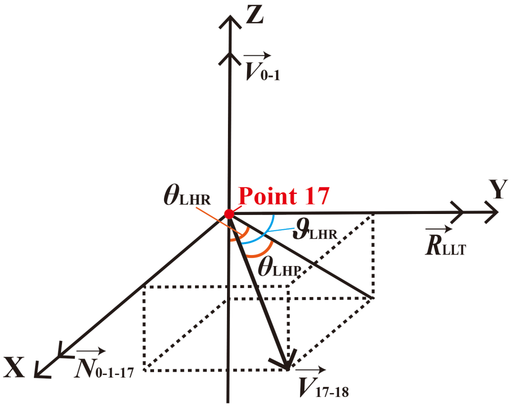


Figure 7. The link frame of the left lower torso of the human skeleton model and the hip joint angles for mapping.

The angles of elbow joints should be calculated in the frames of upper arm links. As shown in Figure 6, the angle of LElbowYaw, θ_{LEY}, should be calculated from the angle between the projection of $\vec{V}_{6-7}$ onto the X-Y plane and the reverse of $\vec{R}_{\text{LUA}}$, which is also the angle between the Y-Z plane and the plane formed by $\vec{V}_{6-5}$ and $\vec{V}_{6-7}$, i.e.,

$$|\theta_{\text{LEY}}| = \arccos\left(\frac{\vec{N}_{2-5-6} \cdot (\vec{V}_{6-5} \times \vec{V}_{6-7})}{|\vec{N}_{2-5-6}||\vec{V}_{6-5} \times \vec{V}_{6-7}|}\right). \tag{10}$$

To obtain θ_{LEY}, intermediate variables ϑ_{LEY1} and ϑ_{LEY2} are needed, i.e.,

$$\vartheta_{\text{LEY1}} = \arccos\left(\frac{\vec{V}_{6-7} \cdot \vec{N}_{2-5-6}}{|\vec{V}_{6-7}||\vec{N}_{2-5-6}|}\right), \tag{11}$$

$$\vartheta_{\text{LEY2}} = \arccos\left(\frac{\vec{V}_{6-7} \cdot \vec{R}_{\text{LUA}}}{|\vec{V}_{6-7}||\vec{R}_{\text{LUA}}|}\right). \tag{12}$$

Then, θ_{LEY} is

$$\theta_{\mathrm{LEY}} = \begin{cases} -|\theta_{\mathrm{LEY}}|, \text{ if } \vartheta_{\mathrm{LEY1}} \leq \frac{\pi}{2}, \\ |\theta_{\mathrm{LEY}}|, \text{ if } \vartheta_{\mathrm{LEY1}} > \frac{\pi}{2} \text{ and } \vartheta_{\mathrm{LEY2}} > \frac{\pi}{2}, \\ |\theta_{\mathrm{LEY}}| - 2\pi, \text{ if } \vartheta_{\mathrm{LEY1}} > \frac{\pi}{2} \text{ and } \vartheta_{\mathrm{LEY2}} \leq \frac{\pi}{2}. \end{cases} \tag{13}$$

The angle of LElbowRoll, θ_{LER}, should be calculated in accordance with the angle between $\vec{V}_{6-7}$ and the X-Y plane plus $\frac{\pi}{2}$, which can also be calculated as

$$\theta_{\mathrm{LER}} = \pi - \arccos\left(\frac{\vec{V}_{6-7} \cdot \vec{V}_{6-5}}{|\vec{V}_{6-7}||\vec{V}_{6-5}|}\right). \tag{14}$$

(3) Joints of Hip_Left: As depicted in Figure 4, LHipRoll is the mother joint of LHipPitch; its axis is collinear with the normal vector of the left lower torso, $\vec{N}_{0-1-17}$, which is calculated as

$$\vec{N}_{0-1-17} = \vec{V}_{0-17} \times \vec{V}_{0-1}. \tag{15}$$

The axis of LHipPitch changes with the rotation of LHipRoll and is collinear with the normal vector to the plane formed by $\vec{N}_{0-1-17}$ and $\vec{V}_{17-18}$. When LHipRoll is at its initial angle, the axis of LHipPitch is collinear with $\vec{V}_{0-17}$.

Then, $\vec{N}_{0-1-17}$ is translated to the origin of the link frame of the left lower torso, point 17, and the X axis of the frame is set collinear with $\vec{N}_{0-1-17}$, as depicted in Figure 7. Because the axis of LHipPitch in the initial posture is vertical to the torso link in Nao but may not be in the human skeleton model, $\vec{V}_{0-1}$, which represents the lower torso, rather than $\vec{V}_{0-17}$, which is aligned with the axis of LHipPitch in the human skeleton model, is translated to point 17 and set collinear with the X axis of the link frame. $\vec{V}_{0-1}$ is collinear with the Z axis. Then, a reference vector $\vec{R}_{\mathrm{LLT}}$ is set collinear with the Y axis to complete a right-handed frame; this vector is obtained as follows:

$$\vec{R}_{\mathrm{LLT}} = \vec{V}_{0-1} \times \vec{N}_{0-1-17}, \tag{16}$$

where the subscript LLT is an abbreviation for "left lower torso".

The angles of hip joints are calculated in the frames of lower torso links. As illustrated in Figure 7, the LHipRoll angle, θ_{LHR}, should be calculated in accordance with the angle between the reverse of $\vec{V}_{0-1}$ and the projection of $\vec{V}_{17-18}$ onto the Y-Z plane, which is also the angle between the X-Z plane and the plane formed by $\vec{N}_{0-1-17}$ and $\vec{V}_{17-18}$, i.e.,

$$|\theta_{\mathrm{LHR}}| = \arccos\left(\frac{\vec{R}_{\mathrm{LLT}} \cdot (\vec{N}_{0-1-17} \times \vec{V}_{17-18})}{|\vec{R}_{\mathrm{LLT}}||\vec{N}_{0-1-17} \times \vec{V}_{17-18}|}\right). \tag{17}$$

To obtain θ_{LHR}, ϑ_{LHR} is introduced:

$$\vartheta_{\mathrm{LHR}} = \arccos\left(\frac{\vec{V}_{17-18} \cdot \vec{R}_{\mathrm{LLT}}}{|\vec{V}_{17-18}||\vec{R}_{\mathrm{LLT}}|}\right). \tag{18}$$

Then, θ_{LHR} is obtained as

$$\theta_{\mathrm{LHR}} = \begin{cases} |\theta_{\mathrm{LHR}}|, \text{ if } \vartheta_{\mathrm{LHR}} \leq \frac{\pi}{2}, \\ -|\theta_{\mathrm{LHR}}|, \text{ if } \vartheta_{\mathrm{LHR}} > \frac{\pi}{2}. \end{cases} \tag{19}$$

The angle of LHipPitch, θ_{LHP}, should be computed from the angle between $\vec{V}_{17-18}$ and the Y-Z plane, i.e.,

$$\theta_{\mathrm{LHP}} = -\frac{\pi}{2} + \arccos\left(\frac{\vec{V}_{17-18} \cdot \vec{N}_{0-1-17}}{|\vec{V}_{17-18}||\vec{N}_{0-1-17}|}\right). \tag{20}$$

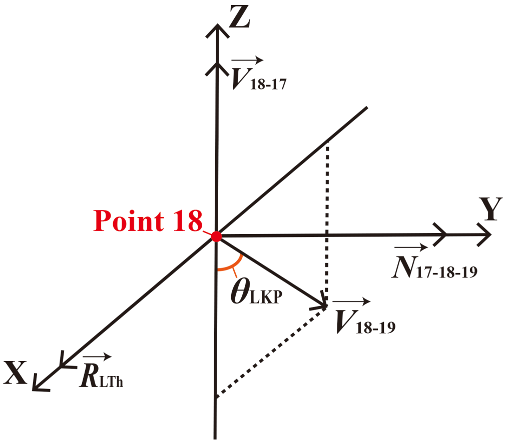


Figure 8. The link frame of the left thigh of the human skeleton model and the knee joint angle for mapping.

(4) Joint of Knee_Left: The axis of LKneePitch is collinear with the normal vector to Plane 17-18-19, $\vec{N}_{17-18-19}$, which is calculated as

$$\vec{N}_{17-18-19} = \vec{V}_{18-19} \times \vec{V}_{18-17}. \tag{21}$$

Then, $\vec{N}_{17-18-19}$ is translated to the origin of the frame of the left thigh, point 18, and set collinear with the Y axis of the frame, as illustrated in Figure 8. $\vec{V}_{18-17}$ is collinear with the Z axis of the frame. A reference vector $\vec{R}_{\text{LTh}}$ that is collinear with the X axis is introduced to complete a right-handed frame; this vector is obtained as follows:

$$\vec{R}_{\text{LTh}} = \vec{N}_{17-18-19} \times \vec{V}_{18-17}, \tag{22}$$

where the subscript LTh is an abbreviation for “left thigh”.

Knee joint angles are calculated in the frames of thigh links. The angle of LKneePitch, θ_{LKP}, should be computed in accordance with the angle between the projection of $\vec{V}_{18-19}$ onto the X-Z plane and the reverse of $\vec{V}_{18-17}$. Because the left knee joint has only one degree of freedom (DOF), $\vec{V}_{18-19}$ is on the X-Z plane of the frame of the left thigh; consequently, θ_{LKP} can be obtained as follows:

$$\theta_{\text{LKP}} = \pi - \arccos(\frac{\vec{V}_{18-19} \cdot \vec{V}_{18-17}}{|\vec{V}_{18-19}||\vec{V}_{18-17}|}). \tag{23}$$

The joint angles obtained above are considered to be the joint angles of the human skeleton model and will be set as the target joint angles of the robot. If the value of any angle is beyond the limit of the corresponding robot joint, the angle will be assigned a marginal value when it is applied to the robot joint.

3.3. Head Motion Mapping

To enable the robot agent to act more like a human and allow the human demonstrator to control the camera on the robot’s head in real time, a head motion imitation function is incorporated. Due to the fact that the information obtained from the skeleton data is limited and insufficient for the head motions of a humanoid robot, the head motion mapping needs an auxiliary method for the joint mapping through GA-LVVJ. The Kinect SDK provides a face tracking algorithm with which the rotation angles of HeadPitch and HeadYaw can be extracted and applied for head motion imitation.

This function not only enables the system to achieve better performance in terms of imitation similarity but also allows the demonstrator to observe the environment around the robot much more easily when performing a task such as grabbing an object or searching for something, since the demonstrator can see what "the eyes of the robot" can see in real time.

3.4. Hand Motion Mapping

Due to the same reason as the head motion, the hand motion mapping also needs an auxiliary method for the joint mapping through GA-LVVJ, i.e., a gesture recognition algorithm. The hands of the Nao robot have two states, i.e., open and closed. With these two states, the robot can perform tasks such as grasping objects. If the sensor captures a human hand in a closed state, the robot will close its corresponding hand, and if the sensor captures a human hand in an open state, the robot will open its corresponding hand. If the human hand is detected to be neither open nor closed, the state of the robot's hand will remain unchanged.

3.5. Locomotion Mapping

To achieve better and more complete whole-body imitation in addition to manipulation of the robot agent to perform tasks requiring locomotion, the walking mode is used. In the walking mode, the robot can replicate human rotations and displacements with preset gaits. Skeleton data are extracted in each motion loop, and the rotation angle and displacement are calculated within each locomotion loop. Figure 9 depicts the calculation of the rotation angle and displacement between two adjacent locomotion loops.

Because the locomotion of the robot should be controlled in the robot's base frame, of which the X axis points forward, the Y axis points rightward and the Z axis projects vertically between the two feet and points upward, the data captured in the Kinect frame should first be mapped to the human base frame. The human base frame is defined to be the same as the robot's base frame.

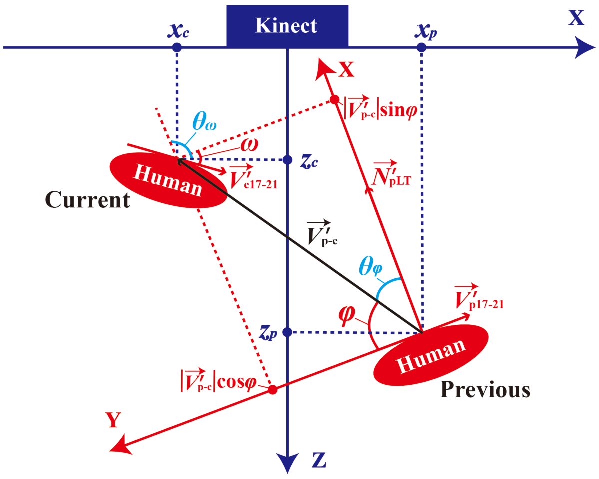


Figure 9. The calculation of the displacement and rotation angle between two adjacent locomotion loops. The blue coordinate axes correspond to the Kinect frame, and the red coordinate axes correspond to the human base frame.

3.5.1. Body Rotation

The vector pointing from Hip_Left (point 17) to Hip_Right (point 21), $\vec{V}_{17-21}$, is introduced as the reference for the rotations of the human body. The corresponding vectors, denoted by $\vec{V}_{\text{c17-21}}$ for the current locomotion loop and $\vec{V}_{\text{p17-21}}$ for the previous one, are obtained for every pair of adjacent locomotion loops. Then, $\vec{V}_{\text{c17-21}}$ and $\vec{V}_{\text{p17-21}}$ are projected onto the horizontal plane (the X-Z plane in the Kinect frame) to eliminate the errors caused by vertical deviations. The projections are denoted by $\vec{V'}_{\text{c17-21}}$ and $\vec{V'}_{\text{p17-21}}$, respectively, and the angle between these two vectors, $|\omega|$, is calculated as

$$|\omega| = \arccos\left(\frac{\vec{V'}_{\text{c17-21}} \cdot \vec{V'}_{\text{p17-21}}}{|\vec{V'}_{\text{c17-21}}||\vec{V'}_{\text{p17-21}}|}\right), \tag{24}$$

where $\vec{V'}_{\text{c17-21}}$ and $\vec{V'}_{\text{p17-21}}$ are collinear with the Y axes of the corresponding human base frames, as shown in Figure 9.

To determine whether the robot should turn clockwise or counterclockwise, the vector representing the direction of the lower torso on the horizontal plane in the previous locomotion loop, $\vec{N'}_{\text{pLT}}$, is introduced as follows:

$$\vec{N'}_{\text{pLT}} = \vec{V}_{\text{up}} \times \vec{V'}_{\text{p17-21}}, \tag{25}$$

where $\vec{V}_{\text{up}}$ is a vertically upward vector. $\vec{N'}_{\text{pLT}}$ is also collinear with the X axis of the previous human base frame, as seen from Figure 9. Then, the angle θ_ω is introduced as follows:

$$\theta_\omega = \arccos\left(\frac{\vec{V'}_{\text{c17-21}} \cdot \vec{N'}_{\text{pLT}}}{|\vec{V'}_{\text{c17-21}}||\vec{N'}_{\text{pLT}}|}\right), \tag{26}$$

Thus, ω can be obtained as follows:

$$\omega = \begin{cases} |\omega|, & \theta_\omega \leq \frac{\pi}{2}, \\ -|\omega|, & \theta_\omega > \frac{\pi}{2}. \end{cases} \tag{27}$$

A positive value means that the human demonstrator is turning counterclockwise, whereas a negative value means that the demonstrator is turning clockwise. If $|\omega|$ meets or exceeds a threshold value ω_0, the robot will rotate with angle ω.

3.5.2. Displacement

The position of the Spine_Base skeleton point (point 0) is regarded as the position of the demonstrator. The vector pointing from the previous position of the Spine_Base point to the current position of Spine_Base is represented by $\vec{V}_{\text{p-c}}$. After the errors caused by vertical deviations of the Spine_Base point have been eliminated, the distance between the positions of the human demonstrator in two adjacent locomotion loops can be obtained as follows:

$$|\vec{V'}_{\text{p-c}}| = \sqrt{(x_{\text{c}} - x_{\text{p}})^2 + (z_{\text{c}} - z_{\text{p}})^2}, \tag{28}$$

where $\vec{V'}_{\text{p-c}}$ is the projection of $\vec{V}_{\text{p-c}}$ onto the horizontal plane, x_{c} is the X coordinate of the Spine_Base point in the Kinect frame in the current locomotion loop, and x_{p} is the X coordinate of the Spine_Base point in the Kinect frame in the previous locomotion loop. Similarly, z_{c} is the Z coordinate of Spine_Base in the Kinect frame in the current locomotion loop, and z_{p} is the previous Z coordinate. Then, we can obtain the angle φ as depicted in Figure 9, using the same approach as for the calculation of ω. First, the angle between $\vec{V'}_{\text{p-c}}$ and $\vec{N'}_{\text{pLT}}$ is introduced as follows:

$$\theta_\varphi = \arccos\left(\frac{\vec{V'}_{\text{p-c}} \cdot \vec{N'}_{\text{pLT}}}{|\vec{V'}_{\text{p-c}}||\vec{N'}_{\text{pLT}}|}\right). \tag{29}$$

Then, φ can be obtained as follows:

$$|\varphi| = \pi - \arccos\left(\frac{\vec{V'}_{\text{p-c}} \cdot \vec{V'}_{\text{p17-21}}}{|\vec{V'}_{\text{p-c}}||\vec{V'}_{\text{p17-21}}|}\right), \tag{30}$$

$$\varphi = \begin{cases} |\varphi|, & \theta_\varphi \leq \frac{\pi}{2} \\ -|\varphi|, & \theta_\varphi > \frac{\pi}{2}. \end{cases} \tag{31}$$

Thus, we can obtain

$$\Delta x = |\vec{V'}_{\text{p-c}}| \sin \varphi, \tag{32}$$

$$\Delta y = |\vec{V'}_{\text{p-c}}| \cos \varphi, \tag{33}$$

where Δx is the displacement along the X axis of the human base frame and Δy is the displacement along the Y axis of the human base frame. For convenience in task execution, we directly apply the displacements of the human demonstrator to the robot.

The displacement of the human's left foot between two adjacent locomotion loops, d_l, and the displacement of the right foot, d_r, are both considered. If both d_l and d_r meet or exceed a threshold value of d_0, then the robot will enter the walking mode to execute locomotion with the corresponding displacement.

3.6. Discussion of Extension to Other Humanoids

The existing humanoid robots are inspired and developed in imitation of human being's body, such as DARwIn [46] and Asimo [47], so their overall configurations are usually similar, especially the joints of shoulders, elbows, hips and knees. These joints mainly decide the postures of a humanoid robot. Since the proposed motion mapping methods are based on human being's motion information and general characteristics of humanoid robots, it can be extended to other types of humanoid robots.

In terms of GA-LVVJ, after the link vectors are constructed, the virtual joints should be always set in accordance with the robot agent. Then the link frames can be defined. When calculating joint angles, if there are no more than two virtual joints between two link vectors, such as the human skeleton models corresponding to Nao and DARwIn, the calculation of this type of humanoid robots is the same as that described in Section III-B. If there are more than two virtual joints between two link vectors, such as the shoulder joints and the hip joints corresponding to Asimo, the DOFs of these joints are redundant, and thus multiple solutions exist. In this case, the unique solution can be obtained if the virtual joints' second child link is considered. For example, the link vector of a lower arm link can help decide the shoulder joint angles, and the link vector of a tibia can help decide the hip joint angles.

For head motion mapping, the algorithm used in our system can extract pitch, roll and yaw angles of the neck, which can meet the needs of most humanoid robots.

For hand motion mapping, if humanoid robots, such as Asimo, have dexterous hands with multiple DOFs, more effective algorithms of gesture recognition are then needed for dexterous hand motions.

With respect to locomotion mapping, it is irrelevant to the body configurations of humanoid robots, so the proposed method can be applied to all the humanoid robots that can walk.

4. Mode Switch

The proposed system includes three modes, i.e., the single-support mode, the double-support mode and the walking mode. To ensure that the robot will switch to the correct mode with only inputs in the form of natural human motions rather than requiring additional postures or other devices and to reduce wobble, a filter strategy is employed. In addition, to ensure the balance and stability of the robot during mode switching, some transition motions are applied.

4.1. Filter Strategy

Our proposed filter strategy has two purposes. First, it is used to ensure that the robot switches to the correct mode through repeated detection and error filtering. Second, the filter strategy is employed to filter unstable motions before the mode changes. Details are presented in the following subsections.

4.2. Double-Support Mode to Single-Support Mode

In the double-support mode, the robot's feet are flat and limited to the horizontal plane, and they can move on the ground. When the human demonstrator lifts one foot such that the height threshold h_0 is reached in three consecutive motion loops, the robot will lean toward the side corresponding to the supporting leg and shift its center of mass over the resulting single-support polygon. During the shifting of the center of mass, the robot's feet are statically fixed on the ground for stability. After the shifting motions are finished, the robot's foot will be lifted, and the imitation motions will be executed. In the single-support phase, the supporting foot is fixed.

When a human in walking with a normal gait, he or she will similarly lift his or her left and right feet to some low height by turns, which will also generate height differences. After all, the lifting of one foot in the single-support mode can be regarded as a short-duration part of the process of stepping in the walking mode. Therefore, to avoid the situation in which the humanoid robot may mistakenly enter the single-support mode when the demonstrator starts to walk, the humanoid robot is constrained to lift its foot only if the height threshold h_0 is reached in at least three consecutive motion loops. In addition, occasional leaping errors caused by the RGB-D camera are prevented from triggering the wrong mode in this way. This approach results in a slight delay, but it can ensure that the robot will not enter the wrong motion mode. Based on the experimental results, h_0 of our system is set to 5 cm, which is much smaller than the height threshold in the work reported in [24].

Although leaping errors do occur, most of the time, the detected height difference is less than a small value h_0' when both feet are on the ground. The execution of lower-limb motions in the double-support mode is filtered to occur only when the height difference is greater than h_0'. In this way, any wobbling of the robot before the lifting of the foot will be alleviated. Such wobbles are expected to be caused by attempts to imitate the motion of the lifted human leg before the height threshold of the single-support mode is reached.

4.3. Double-support Mode to Walking Mode

When the displacements of both feet or the rotation angle of the human demonstrator reach the threshold d_0 or ω_0 respectively at the start of one locomotion loop, the humanoid robot will enter the walking mode. Several preset walking gaits are specified for use in reaching the required displacement and rotation angle. The displacement threshold must be met by both feet because it may occur that the human demonstrator will reach out with one foot without intending to walk, in which case the robot should remain in the double-support mode. In addition, the threshold should not be set too small to avoid undesired effects of the errors of the RGB-D camera and tiny movements or wobbles of the human demonstrator during the double-support phase.

The displacements of both feet are also detected in every motion loop. If a small displacement d_0' of either foot is detected, the execution of lower-limb imitation in this motion loop will be stopped. The activation will be delayed to the next motion loop in which no such displacement is detected. This approach is used to filter out imitation of a reaching-out leg motion and to alleviate instability when the demonstrator starts to walk while the robot is still in the double-support phase, thus contributing to safe switching into the walking mode.

4.4. Single-support Mode to Double-support Mode

When the human demonstrator is standing on one foot and then places his or her free foot on the ground, if the free foot is detected to be below the height threshold for three consecutive motion loops, then the humanoid robot will begin to transition into the double-support mode. This requirement of three consecutive threshold checks is established to avoid leaping errors. To ensure that the swing foot lands safely, its orientation is first set parallel to the ground, and the leg angles are then changed to place it on the ground. The swing foot is kept parallel to the ground, and the whole-body balancer is also employed during this period.

5. Balance Control

Due to the differences in body configuration and weight distribution between the human demonstrator and the humanoid robot, direct application of the human joint angles as the target joint angles of the robot may cause it to fall down, especially when the robot is in the single-support phase. To maintain balance, the center of mass of the robot must be kept centered over the support polygon. The support polygon here is the convex hull of the footprint. Because of the difference in dynamic features between the robot and the human body, and for simplicity, we discuss only the problem of balance control under the action of gravity, and the imitation process is treated as a quasi-static process; i.e., the state in each moment is considered to be a static state.

The balance control strategy we have used is based on the quadratic programming (QP) method. We suppose that $\mathbf{\Theta}_\mathrm{T}$ denotes the joint angle vector of a target pose for imitation. The joint angles of the target pose are acquired from the pose of the human skeleton model using the GA-LVVJ method and the head motion mapping method. The joint angle vector of a stable pose is denoted as $\mathbf{\Theta}_\mathrm{S}$, which is obtained through the balance control. The stable pose will directly applied to the robot instead of the target pose. To ensure that $\mathbf{\Theta}_\mathrm{S}$ is close to $\mathbf{\Theta}_\mathrm{T}$, the problem can be described as a QP problem on the velocity level, i.e.,

$$\text{min.} \quad \frac{1}{2}\left\|\dot{\mathbf{\Theta}} - \dot{\mathbf{\Theta}}_\mathrm{T}\right\|_2^2 \tag{34}$$

$$\text{s.t.} \quad A^T\dot{\mathbf{\Theta}} = \mathbf{B}, \tag{35}$$

$$C^T\dot{\mathbf{\Theta}} \geq \mathbf{D}, \tag{36}$$

where $\dot{\mathbf{\Theta}} \in \mathbf{R^n}$ denotes the unknown joint angle velocity vector, $\dot{\mathbf{\Theta}}_\mathrm{T} \in \mathbf{R^n}$ denotes the joint angle velocity vector of the target pose acquired from the derivative of $\mathbf{\Theta}_\mathrm{T} \in \mathbf{R^n}$; $A \in R^{n\times l}$ is a matrix which expresses linear equality constraints, $B \in R^l$ is a vector which expresses linear equality constraints; $C \in R^{n\times m}$ is a matrix which expresses linear inequality constraints, and $D \in R^m$ is a vector which expresses linear inequality constraints. n is the dimension of the joint space, l is the dimension of the equality constraints, and m is the dimension of the inequality constraints. The equality constraints are used to keep the feet fixed or in a plane, and the inequality constraints are used to constrain the projection of the center of mass within the corresponding support polygon. On the Nao robot, this QP problem is solved every 20 ms. Once the optimal solution $\dot{\mathbf{\Theta}}_\mathrm{S} \in \mathbf{R^n}$ is got, $\mathbf{\Theta}_\mathrm{S} \in \mathbf{R^n}$ can be obtained from the integral of $\dot{\mathbf{\Theta}}_\mathrm{S}$ and applied to the robot, and thus stable and similar imitation motions can be achieved in real time.

6. Similarity Evaluation

To quantitatively evaluate the robot's motion similarity in the single- or double-support mode, the robot's pose at a given moment is compared with that of the human demonstrator which is represented by the human skeleton model. The vectors representing the robot's pose can be obtained from a forward kinematics calculation using joint angles, whereas the vectors representing the human body are link vectors of the human skeleton model, which are constructed based on the skeleton points captured by the RGB-D camera. The evaluation is inspired by the correlation coefficient between two variable vectors introduced in [48]:

$$r_{ij} = \cos\alpha_{ij}, \tag{37}$$

where r_{ij} denotes the correlation coefficient between vectors i and j and α_{ij} denotes the angle between vectors i and j. Considering the cognitive differences between different situations, two similarity evaluation methods are proposed as follows.

6.1. Whole-Body-Focused (WBF) Method

In the first similarity evaluation method, the similarity is evaluated based on the overall perception of the whole body with the ground as the reference. This method considers the positions and poses of the body links in the world frame.

In practice, the base frame of the robot is rotated and translated to coincide with that of the human for similarity evaluation, which is regarded as the world frame. The vectors which represent the robot's pose are rotated and translated with the base frame of the robot. The correlation coefficient between one robot vector and the corresponding human vector in the world frame, r_w, is

$$r_w = \cos \alpha_w, \tag{38}$$

where $\alpha_w \in [0, \pi]$ is the angle between the two corresponding vectors in the world frame.

In the proposed method, the vectors of the two upper arms, the two lower arms, the two thighs (upper legs), the two tibias (lower legs), the torso and the head are considered. Because the feet have only a slight influence on the similarity and the detection precision for the human's feet is relatively low, the vectors of the two feet are not considered. Therefore, the correlation coefficient between the postures of the robot and the human using the WBF method is

$$R_{\mathrm{W}} = \frac{\sum_{w=1}^{10} r_w}{10} = \frac{\sum_{w=1}^{10} \cos \alpha_w}{10}, \tag{39}$$

where R_{W} is taken to be the similarity index of the WBF evaluation method.

6.2. Local-Link-Focused (LLF) Method

In the first proposed metric, chances are that the posture of a chain of links (such as an arm chain consisting of an upper-arm link and a lower-arm link) of the robot in the world frame look will very similar to that of the human, but the corresponding values of the link correlation coefficients will nevertheless be quite low because the mother link in the robot chain (such as the upper-arm link in an arm chain) may deviate from the direction of the corresponding human link in the world frame. For example, the directions of some robot mother links may be modified considerably to maintain balance, and thus, all of the child links will be affected. Therefore, the local posture similarity cannot be properly assessed when using the WBF method. However, the local motions of certain links will be more important than whole-body motions under some circumstances. For example, human observers will pay more attention to the arms and upper torso of the demonstrator when they wish to learn arm movements in a dance class. Therefore, to enable the similarity metric to reflect the effects of local postures, a link vector of the human or the robot should be transformed into the link frame of its own mother link. When calculating the correlation coefficient of two vectors, the two link frames of the mother links of this pair of corresponding vectors are translated and rotated to coincide with each other and are unified as one frame. Meanwhile, the link vectors are translated and rotated with the link frames of their mother links, respectively. Then the correlation coefficient between one robot vector and the corresponding human vector in the unified frame, r_L, is

$$r_L = \cos \alpha_L, \tag{40}$$

where $\alpha_l \in [0, \pi]$ denotes the angle between two corresponding vectors in their unified frame.

So the correlation coefficient between the postures of the robot and the human using the LLF evaluation method is

$$R_{\mathrm{L}} = \frac{\sum_{l=1}^{10} r_l}{10} = \frac{\sum_{l=1}^{10} \cos\alpha_l}{10}, \tag{41}$$

where R_{L} is also the similarity index of the LLF evaluation method.

In the experimental section, the two metrics proposed here will be applied to evaluate the similarities of whole-body and local-link features, respectively.

7. Experiment

To validate the effectiveness of the proposed imitation system and evaluate the imitation similarity, two motion imitation experiments were conducted. The joint angles of the robot and the human skeleton model which represents the human demonstrator were tracked, and the similarity indexes of both the WBF and LLF evaluation methods were calculated for each motion loop. In addition, the proposed system was applied for teleoperation to perform a task. The task required grabbing and releasing objects in both the double- and single-support modes, and it also required the robot to move to a target location.

7.1. Real-Time Whole-Body Imitation of Singly and Doubly Supported Motions

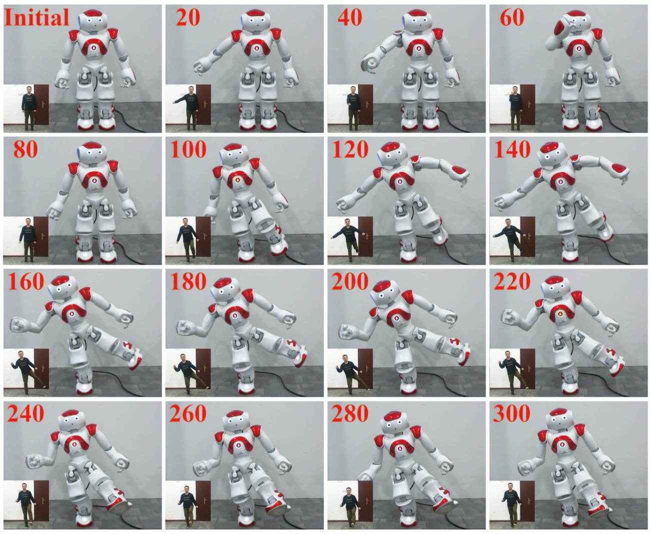


Figure 10. Snapshots of the robot and the human demonstrator in the first experiment.

In this experiment, the demonstrator performed a series of motions that included both doubly supported motions and singly supported motions. Snapshots of this experiment are shown in Figure 10, where the numbers denote the motion loop ordinal number of each snapshot. The double-support phase corresponded to the 1st motion loop through the 85th motion loop. During this period, the main motions of the demonstrator were movements of the right arm. The mode-switch phase corresponded to the 85th motion loop through the 100th motion loop. The demonstrator lifted his left foot, and the robot moved its center of mass to the right side and lifted its left foot. The single-support phase lasted from the 100th motion loop to the end. The demonstrator performed a sequence of singly supported postures, some of which made it difficult for even the human demonstrator to maintain his balance. The results and analysis of this experiment are as follows.

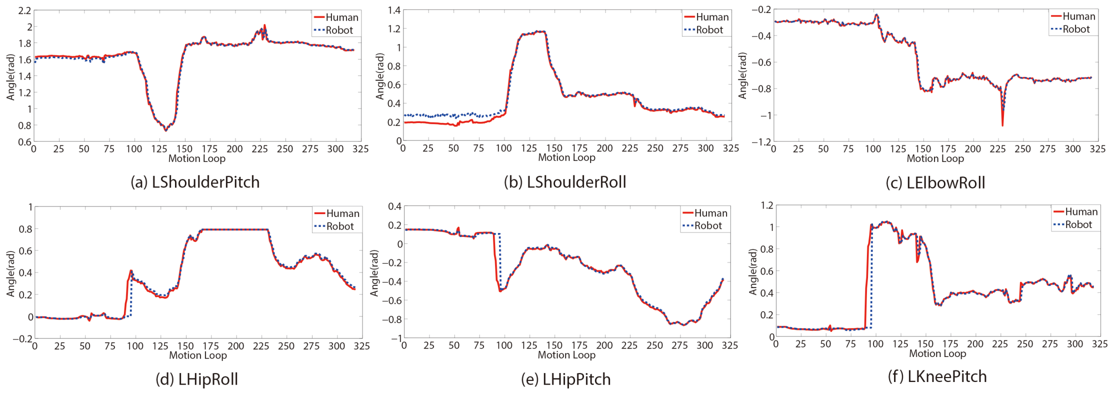
Table 1. Average and Maximum Errors of Key Joint Angles

Joint	Average Error (rad)	Maximum Error (rad)
LShoulderPitch	1.3074×10^{-2}	2.0977×10^{-1}(S)
LShoulderRoll	3.1207×10^{-2}	1.1889×10^{-1}(D)
LElbowRoll	5.2856×10^{-3}	1.7826×10^{-1}(S)
LHipRoll	1.4330×10^{-2}	4.1643×10^{-1}(*)
LHipPitch	1.5433×10^{-2}	6.0684×10^{-1}(*)
LKneePitch	1.9829×10^{-2}	9.3876×10^{-1}(*)
RShoulderPitch	3.7299×10^{-2}	6.2313×10^{-1}(D)
RShoulderRoll	1.7255×10^{-2}	1.3195×10^{-1}(D)
RElbowRoll	4.5075×10^{-2}	4.1345×10^{-1}(D)
RHipRoll	5.2704×10^{-2}	1.3830×10^{-1}(S)
RHipPitch	2.2450×10^{-2}	7.4136×10^{-2}(S)
RKneePitch	4.5989×10^{-2}	1.6979×10^{-1}(S)

7.1.1. Computational Cost

The average computational cost of the posture mapping (calculation of the target joint angles of the robot) was 0.033 ms, much lower than that of typical numerical IK solvers such as [21], which requires 1.04 ms for posture mapping on average.

7.1.2. Error Analysis



(a) LShoulderPitch

(b) LShoulderRoll

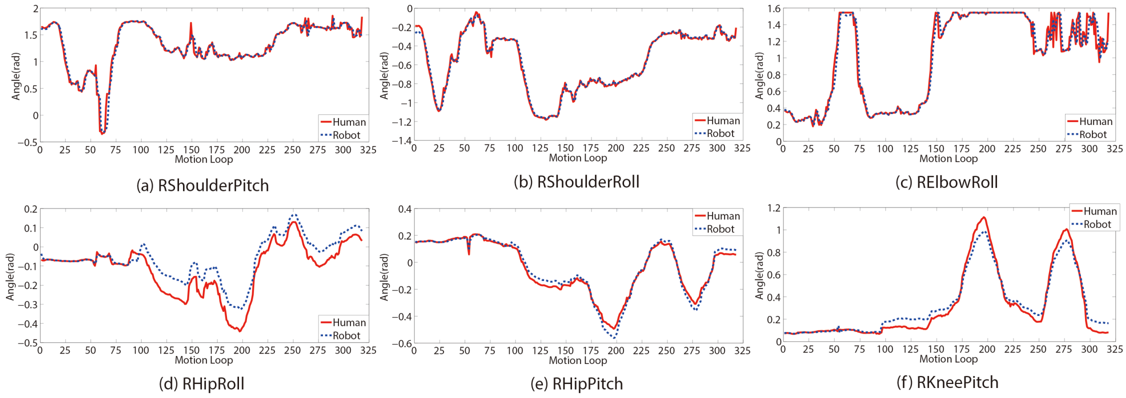
(c) LElbowRoll

(d) LHipRoll

(e) LHipPitch

(f) LKneePitch

Figure 11. Left key joint angle trajectories of the human and the robot in the first experiment.



(a) RShoulderPitch

(b) RShoulderRoll

(c) RElbowRoll

(d) RHipRoll

(e) RHipPitch

(f) RKneePitch

Figure 12. Right key joint angle trajectories of the human and the robot in the first experiment.

The average and maximum errors between key joint angles in the human skeleton model and the robot are listed in Table 1, where the letters "D", "S", and $*$ indicate that the maximum errors occurred in the double-support phase, the single-support phase and the mode-switch phase, respectively. As seen from Table 1, the average errors are very small, and the maximum errors are reasonable, thereby verifying the effectiveness of the proposed real-time imitation system. The observed errors are attributed to the following causes. (1) The delay of the robot's motions will lead to angle deviations. (2) Joint angles will be modified when the original target angles do not satisfy the balance conditions for a stable pose. (3) The robot's joint limits will restrict some postures. (4) During the mode-switch phase (from the double-support mode to the single-support mode), the robot cannot lift its foot immediately after the human's foot reaches the threshold height for stability. (5) Due to the error of the Kinect II, the joint angles extracted from the detected data of the same joint between two motion loops may show a leaping deviation, whereas the robot's joint angles vary continuously.

7.1.3. Consistency of Joint Angle Trajectories

The angle trajectories of several key joints are depicted in Figure 11 and Figure 12. Although there are some small deviations, the trajectory curves show good joint angle consistency between the robot and the human skeleton model. From the 1st motion loop to the 85th motion loop (double-support phase), the uniformity of the joint angle trajectories between the human skeleton model and the robot is good, and the deviations are almost invisible in most trajectories. The deviations of the LShoulderRoll angle are due to the self-collision avoidance of the robot. From the 85th motion loop to the 100th motion loop (mode-switch phase), the center-of-mass transference and the filter strategy introduce an obvious delay in lifting the left foot. This delay is evident in the angle trajectories of LHipRoll, LHipPitch and LKneePitch (Figure 11(d), (e) and (f)). As shown in Table 1, these three joint angles' maximum errors occur during the mode-switch phase and are among the largest of the maximum errors. From the 100th motion loop to the end (single-support phase), because the balance of the singly supported postures is mainly maintained by the right leg, the angles of RHipRoll, RHipPitch, and RKneePitch exhibit obvious deviations, as seen in Figure 12(a), (b) and (c). These joint angles are modified to maintain balance. The maximum errors of these three joint angles appear in the single-support phase, but the values are quite low. From the 255th motion loop to the end, there are obvious fluctuations and errors in the RElbowRoll angle. This is because the right lower arm of the demonstrator obscured the RGB-D camera's view of the right elbow. The other joint angle trajectories during this period show good uniformity.

7.1.4. Similarity Evaluation

The similarity indexes of the WBF method and the LLF method were applied to evaluate our imitation system in real time. The trajectories of both indexes are shown in Figure 13. In the double-support phase, the WBF similarity index is above 0.94, and the LLF index is above 0.98. The variation trends of the two curves are similar, and the value of the LLF index is higher than that of the WBF index. From the 55th motion loop to the 70th motion loop, the similarity index curves show obvious fluctuations because of the slight delay in the movements of the robot's right arm. This slight delay can be observed from the RShoulderPitch angle trajectories (Figure 12(a)). During the mode-switch phase, both similarity index curves dip down and form troughs. However, the index values remain above 0.93. In the single-support phase, the WBF similarity index shows a decline compared with that in the double-support phase, whereas the LLF index returns to a high level. GA-LVVJ considers the local geometrical relations between articulated links and obtains the joint angles of the imitated posture directly and accurately, thereby contributing to the excellent observed performance in terms of local posture similarity.

In the WBF method, in which the ground is regarded as the reference, if the robot is standing on its right foot, then the right tibia is the mother link of the right thigh and the right thigh is the mother link of the torso. In turn, the torso is the mother link of the upper links of the other limbs, which are the mother links of their respective lower links. Thus, the deviations of the mother links will accumulate and affect the positions of their child links in the world frame, even if their local similarity is high. This phenomenon will be more evident during the single-support phase because the links of the supporting leg, which serve as the mother links of the other parts, may deviate considerably to maintain balance. As shown in Figure 14 and Figure 15, in the right-support phase, because the right leg angles are considerably modified to maintain balance, the left-thigh WBF similarity index shows an evident decline in the single-support phase compared with the double-support phase. Because the left tibia is the child link of the left thigh, the left-tibia WBF similarity index shows a sharper decline than that of the left thigh. Furthermore, the decline will be more evident when the single-support posture is more challenging. However, with a lowest value above 0.94, the WBF similarity index still indicates good performance in terms of the imitation similarity as seen from the perspective of the world frame.

In short, this experiment verifies that the joint angle consistency between the human demonstrator and the robot is high and that our system performs well in terms of both the local similarity and the whole-body similarity.

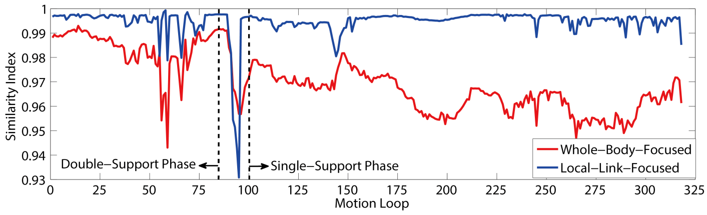


Figure 13. Similarity index trajectories in the first experiment as calculated using the two evaluation methods.

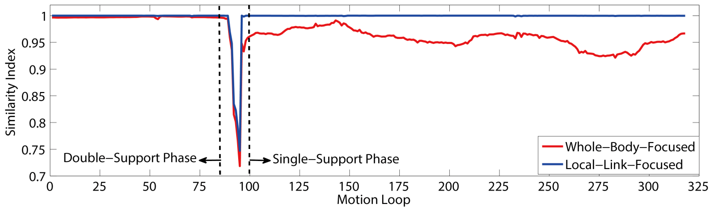


Figure 14. Left-thigh similarity index trajectories in the first experiment as calculated using the two evaluation methods.

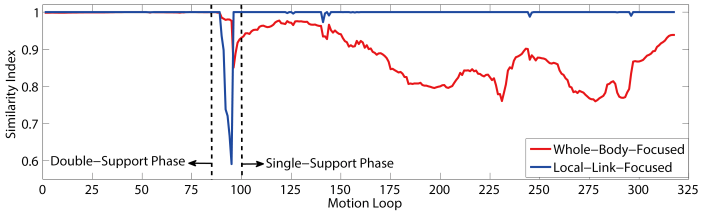


Figure 15. Left-tibia similarity index trajectories in the first experiment as calculated using the two evaluation methods.

7.1.5. Test on Different Human Body Shapes

The system has been tested on five human demonstrators with different body shapes as shown in Figure 16. Still, even difficult single-support motions can be imitated by the robot stably and continuously, which verifies the effectiveness of the proposed methods.

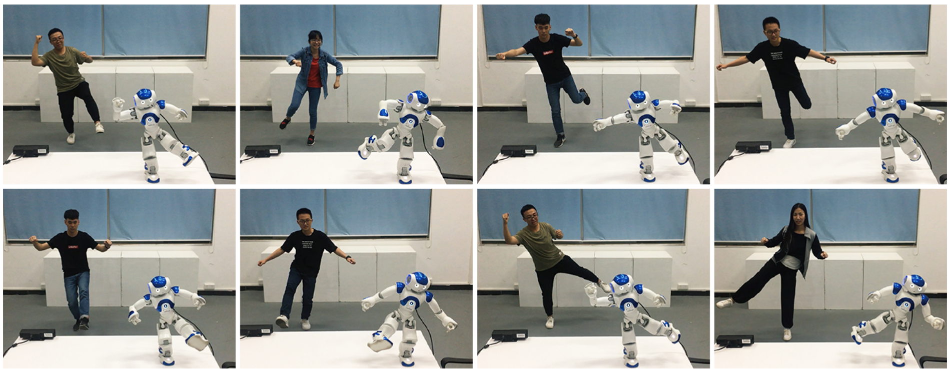


Figure 16. Snapshots of some difficult single-support motions that the robot can perform by imitating human demonstrators with different body shapes.

7.2. *Task-Oriented Teleoperation*

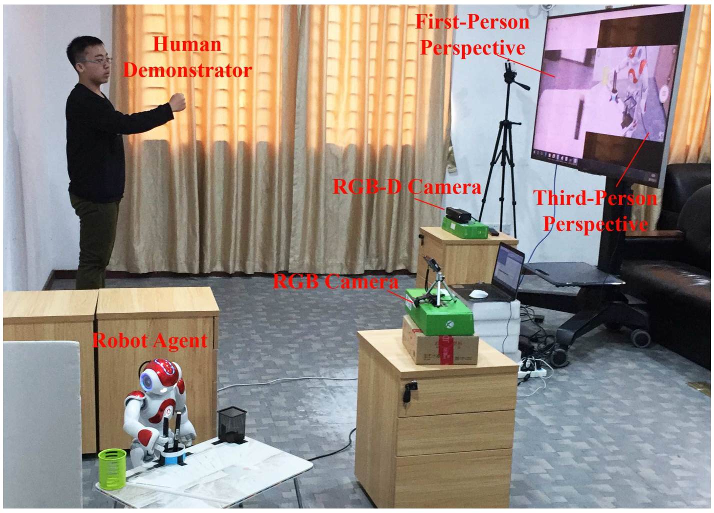


Figure 17. The real-time imitation system used in the task-oriented teleoperation experiment.

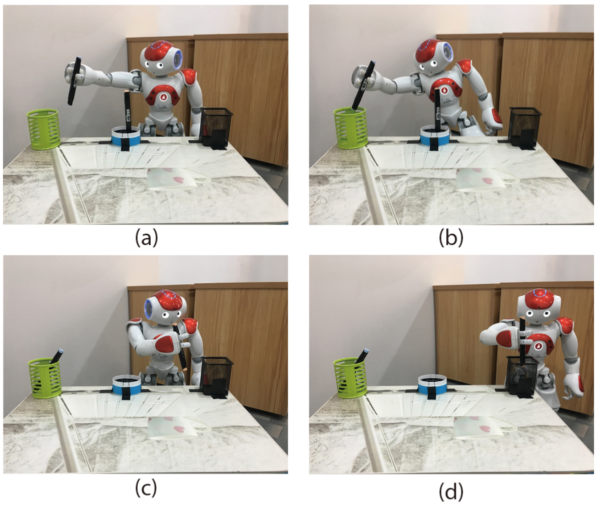


Figure 18. Description of the teleoperation task. (a) shows that the robot cannot reach the green container in the double-support mode. (b) shows that when the robot leans rightward in the single-support mode, it can easily reach the container. (c) shows that the robot cannot reach the black container in the double-support mode. (d) shows that when the robot walks leftward, it can easily reach the container.

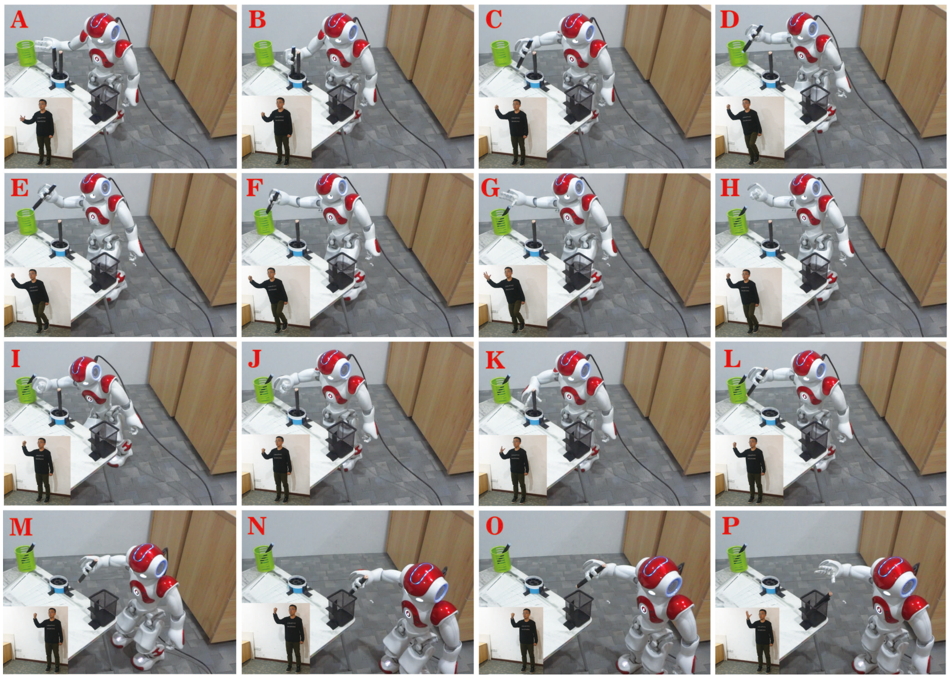


Figure 19. Snapshots of the robot and the human in the task-oriented teleoperation experiment.

In this subsection, we report the use of our system to manipulate the robot to place two pens into two different containers. The experimental setting is shown in Figure 17. The robot was placed at a location in the room where the demonstrator could not see it. A Kinect II RGB-D camera was employed to capture the motions of the human demonstrator. The first-person perspective of the robot was captured by the robot's head-mounted camera, and the third-person perspective of the overall task situation was monitored by an RGB camera; both video feeds were shown on the screen to provide views of the task in real time.

The initial position of the robot dictated that the robot should seize the pens in its right hand. However, the containers were out of range of the robot's right arm when the robot was in the double-support mode, as depicted in Figure 18(a) and (c). The left arm was not used to reach the black container because it was not convenient to transfer the pen from the right hand to the left hand. Thus, two approaches were employed to reach the containers out of range of the robot's arm. To reach the green container, the robot was required to lift its left foot and lean rightward to cover a longer distance, as shown in Figure 18(b). To reach the black container, the robot was required to walk leftward, as shown in Figure 18(d).

In the experiment, the robot imitated a series of motions performed by the human demonstrator and successfully placed the pens into the containers. (A)-(H) in Figure 19 show snapshots of the motions of the human and the robot during the process of placing the blue pen into the right green container. (I)-(P) in Figure 19 show snapshots of the motions of the human and the robot during the process of placing the pink pen into the left black container.

This teleoperation experiment verifies that the proposed imitation system can be applied to execute real-time teleoperation tasks stably and effectively.

8. An Imitation Learning Case

Imitation learning is an efficient learning mechanism for reducing the complexity of search spaces and accelerating learning[49]. As shown in Figure 20, a hello motion is demonstrated by five different demonstrators. In each demonstration the moving arm's joint angles mapped through GA-LVVJ are recorded with their corresponding temporal information. Each dataset of a certain joint is aligned temporarily and scaled to a fixed time step. Then the datasets are encoded by Gaussian Mixture Model (GMM), and a generalized joint angle trajectory is produced through Gaussian Mixture Regression (GMR)[50]. Thus, the robot can learn from multiple demonstrations of the hello motion and reproduce a new one.

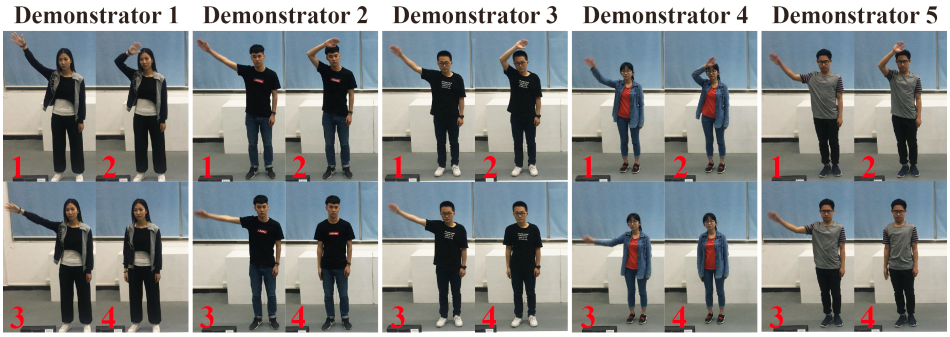


Figure 20. The snapshots of the hello motion performed by 5 Demonstrators.

First, the datasets of different demonstrators are aligned temporarily using Dynamic Time Wrapping (DTW)[51] to improve the coherence of data, then the data of each demonstration is scaled to 100 time steps. The dataset of a joint $\xi = \{\xi_j\}_{j=1}^{N}$ denotes N observations of $\xi_j = \{\xi_{t,j}, \xi_{a,j}\} \in R^2$, where $\xi_{t,j} \in R$ is a temporal value and $\xi_{a,j} \in R$ is a joint angle. Then the dataset is modeled by a GMM of K components, and each data point ξ_j is define by the probability density function

$$p(\xi_j) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\xi_j; \mu_k, \Sigma_k) \tag{42}$$

where π_k denotes the prior probability of component k, and $\mathcal{N}(\xi_j; \mu_k, \Sigma_k)$ denotes the Gaussian distribution whose mean vector and covariance matric are μ_k and Σ_k, respectively. The temporal and angular components of μ_k and Σ_k can be represented separately as

$$\mu_k = \{\mu_{t,k}, \mu_{a,k}\}, \Sigma_k = \begin{pmatrix} \Sigma_{tt,k} & \Sigma_{ta,k} \\ \Sigma_{at,k} & \Sigma_{aa,k} \end{pmatrix}. \tag{43}$$

Then a generalized joint angle trajectory can be obtained by applying GMR as follows. For each component k, the expected distribution of the angular value ξ_a given the temporal value ξ_t is defined by

$$p(\xi_a|\xi_t, k) = \mathcal{N}(\xi_a; \hat{\xi}_{a,k}, \hat{\Sigma}_{aa,k}) \tag{44}$$

$$\hat{\xi}_{a,k} = \mu_{a,k} + \Sigma_{at,k}(\Sigma_{tt,k})^{-1}(\xi_t - \mu_{t,k}) \tag{45}$$

$$\hat{\Sigma}_{aa,k} = \Sigma_{aa,k} - \Sigma_{at,k}(\Sigma_{tt,k})^{-1}\Sigma_{ta,k}. \tag{46}$$

Considering the complete GMM, the expected distribution of ξ_a given ξ_t is defined by

$$p(\xi_a|\xi_t) = \sum_{k=1}^{K} \beta_k \mathcal{N}(\xi_a; \hat{\xi}_{a,k}, \hat{\Sigma}_{aa,k}) \tag{47}$$

where $\beta_k = p(k|\xi_t)$ is the probability of the component k given ξ_t, i.e.,

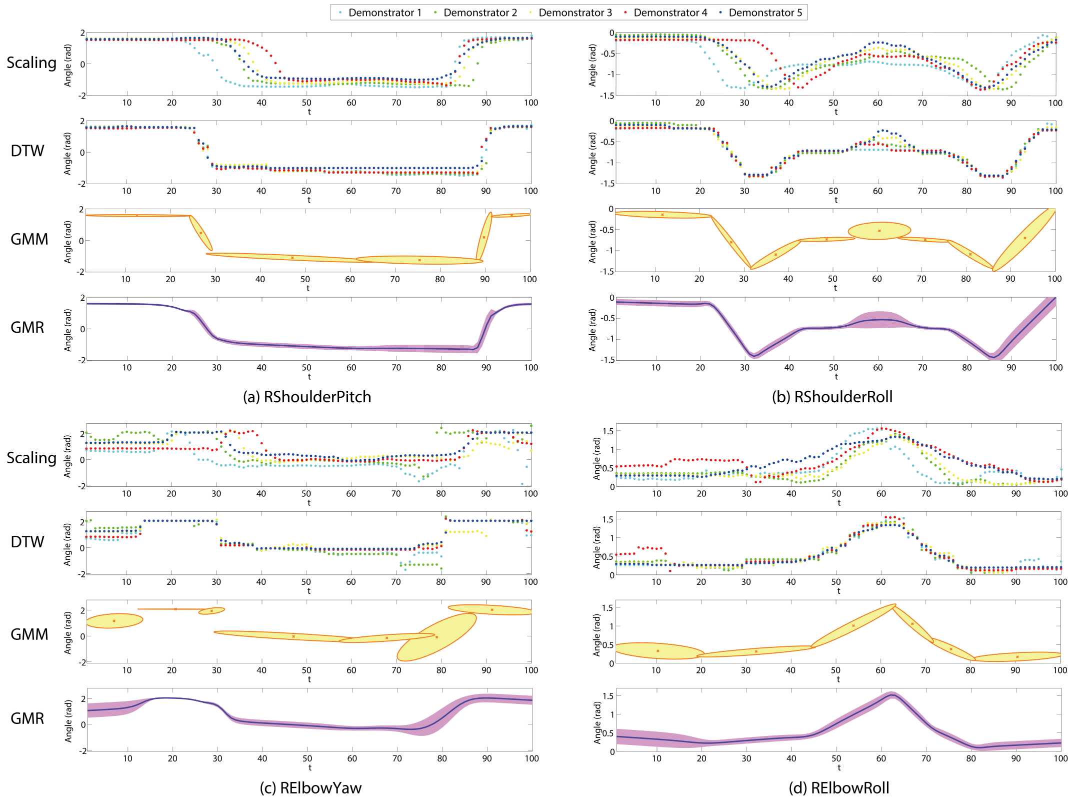
$$\beta_k = \frac{p(k)p(\xi_t|k)}{\sum_{i=1}^{K} p(i)p(\xi_t|i)} = \frac{\pi_k \mathcal{N}(\xi_t; \mu_{t,k}, \Sigma_{tt,k})}{\sum_{i=1}^{K} \pi_i \mathcal{N}(\xi_t; \mu_{t,i}, \Sigma_{tt,i})}. \tag{48}$$

Then the expected distribution of ξ_a given ξ_t is $p(\xi_a|\xi_t) \sim \mathcal{N}(\hat{\xi}_a, \hat{\Sigma}_{aa})$, where the parameters of the Gaussian distribution are

$$\hat{\xi}_a = \sum_{k=1}^{K} \beta_k \hat{\xi}_{a,k}, \hat{\Sigma}_{aa} = \sum_{k=1}^{K} \beta_k^2 \hat{\Sigma}_{aa,k}. \tag{49}$$

Therefore, by evaluating $\{\hat{\xi}_a, \hat{\Sigma}_{aa}\}$, a generalized joint angle trajectory $\hat{\xi} = \{\xi_t, \hat{\xi}_a\}$ and associated covariance matrix can be obtained.

This process is shown in Figure 21. The reproduced hello motion performed by Nao is depicted in Figure 22, which demonstrates the key features of a hello motion effectively.



(a) RShoulderPitch

(b) RShoulderRoll

(c) RElbowYaw

(d) RElbowRoll

Figure 21. The imitation learning process of the hello motion. The first figure of each joint angle (i.e., RShoulderPitch, RShoulderRoll, RElbowYaw or RElbowRoll) depicts the scaling result of the raw datasets. The second figure depicts the DTW and scaling result of the datasets. The third one depicts the GMM encoding result. The forth one depicts the generalized angle trajectory produced by GMR.

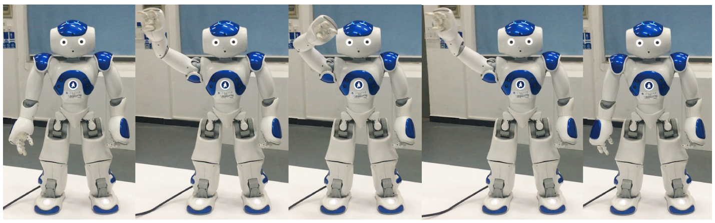


Figure 22. The snapshots of the reproduced hello motion performed by Nao.

9. Conclusion

In this paper, a novel real-time whole-body imitation system for humanoid robots has been proposed. To map the motions of a human demonstrator to a robot, a novel analytical method called GA-LVVJ has been proposed. In addition, a real-time locomotion method has been employed during the walking phase for convenience in completing tasks. To enable accurate and safe mode switching, we have presented a filter strategy to eliminate leaping errors and alleviate wobble. In addition, two quantitative methods of similarity evaluation focusing on the whole body and local links, called the WBF method and the LLF method, respectively, have been proposed and compared. The results of two experiments verify the effectiveness of the proposed imitation system. The first experiment has demonstrated the good stability and high similarity achieved during single-support and double-support imitation. The second experiment has validated the capability to execute tasks via real-time teleoperation using the proposed system. At last, an imitation learning mechanism in which the joint angles of demonstrators are mapped by GA-LVVJ is presented and developed to extend the proposed system.

Author Contributions: Conceptualization, Zhijun Zhang and Yaru Niu; Methodology, Yaru Niu; Software, Yaru Niu and Shuyang Lin; Validation, Zhijun Zhang, Yaru Niu and Ziyi Yan; Formal Analysis, Yaru Niu; Investigation, Yaru Niu; Resources, Zhijun Zhang; Data Curation, Yaru Niu; Writing and Original Draft Preparation, Yaru Niu; Writing and Review & Editing, Zhijun Zhang, Yaru Niu and Ziyi Yan; Visualization, Yaru Niu; Supervision, Zhijun Zhang; Project Administration, Zhijun Zhang; Funding Acquisition, Zhijun Zhang and Yaru Niu.

Funding: This research was funded in part by the National Key R&D Program of China under Grant 2017YFB1002505, National Natural Science Foundation of China under Grants 61603142 and 61633010, Guangdong Foundation for Distinguished Young Scholars under Grant 2017A030306009, The Guangdong Youth Talent Support Program of Scientific and Technological Innovation under Grant 2017TQ04X475, Science and Technology Program of Guangzhou under Grant 201707010225, the Fundamental Research Funds for Central Universities under Grant 2017MS049, Scientific Research Starting Foundation of South China University of Technology, National Key Basic Research Program of China (973 Program) under Grant 2015CB351703, and Natural Science Foundation of Guangdong Province under Grant 2014A030312005.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

GA-LVVJ	Geometrical Analysis Based on Link Vectors and Virtual Joints
WBF	Whole-Body-Focused
LLF	Local-Link-Focused
IK	Inverse kinematic
LUT	Left upper torso
LSP	LShoulderPitch
LSR	LShoulderRoll
LUA	Left upper arm
LEY	LElbowYaw
LER	LElbowRoll
LLT	Left lower torso
LHR	LHipRoll
LHP	LHipPitch
LTh	Left thigh
DOF	Degree of freedom
LKP	LKneePitch
LT	Lower torso
QP	Quadratic programming
GMM	Gaussian Mixture Model
GMR	Gaussian Mixture Regression
DTW	Dynamic Time Wrapping

References

1. Morita, T.; Mase, K.; Hirano, Y.; Kajita, S. Reciprocal attentive communication in remote meeting with a humanoid robot. Proceedings of the 9th international conference on Multimodal interfaces, 2007, pp. 228–235.
2. Qi, R.; Khajepour, A.; Melek, W.W.; Lam, T.L.; Xu, Y. Design, Kinematics, and Control of a Multijoint Soft Inflatable Arm for Human-Safe Interaction. *IEEE Transactions on Robotics* **2017**.
3. Ferraguti, F.; Preda, N.; Manurung, A.; Bonfe, M.; Lambercy, O.; Gassert, R.; Muradore, R.; Fiorini, P.; Secchi, C. An energy tank-based interactive control architecture for autonomous and teleoperated robotic surgery. *IEEE Transactions on Robotics* **2015**, *31*, 1073–1088.
4. Marescaux, J.; Leroy, J.; Rubino, F.; Smith, M.; Vix, M.; Simone, M.; Mutter, D. Transcontinental robot-assisted remote telesurgery: feasibility and potential applications. *Annals of surgery* **2002**, *235*, 487.

5. Meng, C.; Zhang, J.; Liu, D.; Liu, B.; Zhou, F. A remote-controlled vascular interventional robot: system structure and image guidance. *The International Journal of Medical Robotics and Computer Assisted Surgery* **2013**, *9*, 230–239.
6. Butner, S.E.; Ghodoussi, M. Transforming a surgical robot for human telesurgery. *IEEE Transactions on Robotics and Automation* **2003**, *19*, 818–824.
7. Atashzar, S.F.; Polushin, I.G.; Patel, R.V. A Small-Gain Approach for Nonpassive Bilateral Telerobotic Rehabilitation: Stability Analysis and Controller Synthesis. *IEEE Transactions on Robotics* **2017**, *33*, 49–66.
8. Nielsen, C.W.; Goodrich, M.A.; Ricks, R.W. Ecological interfaces for improving mobile robot teleoperation. *IEEE Transactions on Robotics* **2007**, *23*, 927–941.
9. Bakker, P.; Kuniyoshi, Y. Robot see, robot do: An overview of robot imitation. AISB96 Workshop on Learning in Robots and Animals, 1996, pp. 3–11.
10. Riley, M.; Ude, A.; Wade, K.; Atkeson, C.G. Enabling real-time full-body imitation: a natural way of transferring human movement to humanoids. Proceedings of IEEE International Conference on Robotics and Automation (ICRA), 2003, Vol. 2, pp. 2368–2374.
11. Zhao, X.; Huang, Q.; Peng, Z.; Li, K. Kinematics mapping and similarity evaluation of humanoid motion based on human motion capture. Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2004, Vol. 1, pp. 840–845.
12. Gu, Y.; Thobbi, A.; Sheng, W. Human-robot collaborative manipulation through imitation and reinforcement learning. IEEE International Conference on Information and Automation (ICIA), 2011, pp. 151–156.
13. Liu, H.Y.; Wang, W.J.; Wang, R.J.; Tung, C.W.; Wang, P.J.; Chang, I.P. Image recognition and force measurement application in the humanoid robot imitation. *IEEE Transactions on instrumentation and measurement* **2012**, *61*, 149–161.
14. Tomić, M.; Chevallereau, C.; Jovanović, K.; Potkonjak, V.; Rodić, A. Human to humanoid motion conversion for dual-arm manipulation tasks. *Robotica* **2018**, pp. 1–21.
15. Cheng, G.; Kuniyoshi, Y. Real-time mimicking of human body motion by a humanoid robot. Proceedings of the 6th Intelligent Autonomous Systems, 2000, pp. 273–280.
16. Hwang, C.L.; Chen, B.L.; Syu, H.T.; Wang, C.K.; Karkoub, M. Humanoid Robot's Visual Imitation of 3-D Motion of a Human Subject Using Neural-Network-Based Inverse Kinematics. *IEEE Systems Journal* **2016**, *10*, 685–696.
17. Sakka, S.; Poubel, L.P.; Cehajic, D. Tasks prioritization for whole-body realtime imitation of human motion by humanoid robots. Digital Intelligence (DI2014), 2014, pp. 5–p.
18. Kondo, Y.; Yamamoto, S.; Takahashi, Y. Real-time Posture Imitation of Biped Humanoid Robot based on Particle Filter with Simple Joint Control for Standing Stabilization. Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems, 2016, pp. 130–135.
19. Filiatrault, S.; Cretu, A.M. Human arm motion imitation by a humanoid robot. IEEE International Symposium on Robotic and Sensors Environments (ROSE), 2014, pp. 31–36.
20. Chen, J.; Wang, G.; Hu, X.; Shen, J. Lower-body control of humanoid robot NAO via Kinect. *Multimedia Tools and Applications* **2018**, *77*, 10883–10898.
21. Koenemann, J.; Burget, F.; Bennewitz, M. Real-time imitation of human whole-body motions by humanoids. IEEE International Conference on Robotics and Automation (ICRA), 2014, pp. 2806–2812.
22. Avalos, J.; Cortez, S.; Vasquez, K.; Murray, V.; Ramos, O.E. Telepresence using the kinect sensor and the NAO robot. IEEE 7th Latin American Symposium on Circuits & Systems (LASCAS), 2016, pp. 303–306.
23. Ou, Y.; Hu, J.; Wang, Z.; Fu, Y.; Wu, X.; Li, X. A real-time human imitation system using kinect. *International Journal of Social Robotics* **2015**, *7*, 587–600.
24. Poubel, L.P.; Sakka, S.; Ćehajić, D.; Creusot, D. Support changes during online human motion imitation by a humanoid robot using task specification. IEEE International Conference on Robotics and Automation (ICRA), 2014, pp. 1782–1787.
25. Zuher, F.; Romero, R. Recognition of human motions for imitation and control of a humanoid robot. Robotics Symposium and Latin American Robotics Symposium (SBR-LARS), 2012, pp. 190–195.
26. Lei, J.; Song, M.; Li, Z.N.; Chen, C. Whole-body humanoid robot imitation with pose similarity evaluation. *Signal Processing* **2015**, *108*, 136–146.

27. Lee, J.H.; others. Full-body imitation of human motions with kinect and heterogeneous kinematic structure of humanoid robot. IEEE/SICE International Symposium on System Integration (SII), 2012, pp. 93–98.
28. Franz, S.; Nolte-Holube, R.; Wallhoff, F. NAFOME: NAO Follows Me-tracking, reproduction and simulation of human motion. *Jade University of Applied Sciences, Germany* **2013**.
29. Almetwally, I.; Mallem, M. Real-time tele-operation and tele-walking of humanoid robot Nao using Kinect depth camera. 10th IEEE International Conference on Networking, Sensing and Control (ICNSC), 2013, pp. 463–466.
30. Aleotti, J.; Skoglund, A.; Duckett, T. Position teaching of a robot arm by demonstration with a wearable input device. International Conference on Intelligent Manipulation and Grasping (IMG04), 2004, pp. 1–2.
31. Calinon, S.; Billard, A. Incremental learning of gestures by imitation in a humanoid robot. Proceedings of the ACM/IEEE international conference on Human-robot interaction, 2007, pp. 255–262.
32. Asfour, T.; Azad, P.; Gyarfas, F.; Dillmann, R. Imitation learning of dual-arm manipulation tasks in humanoid robots. *International Journal of Humanoid Robotics* **2008**, *5*, 183–202.
33. Liu, X.; Zhou, X.; Liu, C.; Wang, J.; Zhou, X.; Xu, N.; Jiang, A. An interactive training system of motor learning by imitation and speech instructions for children with autism. 9th IEEE International Conference on Human System Interactions (HSI), 2016, pp. 56–61.
34. Boucenna, S.; Anzalone, S.; Tilmont, E.; Cohen, D.; Chetouani, M. Learning of social signatures through imitation game between a robot and a human partner. *IEEE Transactions on Autonomous Mental Development* **2014**, *6*, 213–225.
35. Rozo, L.; Calinon, S.; Caldwell, D.G.; Jimenez, P.; Torras, C. Learning physical collaborative robot behaviors from human demonstrations. *IEEE Transactions on Robotics* **2016**, *32*, 513–527.
36. Alibeigi, M.; Ahmadabadi, M.N.; Araabi, B.N. A fast, robust, and incremental model for learning high-level concepts from human motions by imitation. *IEEE Transactions on Robotics* **2017**, *33*, 153–168.
37. Stephens, B. Integral control of humanoid balance. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2007, pp. 4020–4027.
38. Lin, J.L.; Hwang, K.S. Balancing and Reconstruction of Segmented Postures for Humanoid Robots in Imitation of Motion. *IEEE Access* **2017**, *5*, 17534–17542.
39. Veltrop, T. Control a humanoid robot using the Kinect. Available online: http://bit.ly/RjAVwQ, 2011.
40. Päusch, S.; Hochdorfer, S. Teleoperation of the Robot NAO with a Kinect Camera. Available online: http://www.zafh-servicerobotik.de/, 2012.
41. Suay, B. Humanoid Robot Control and Interaction using Depth Camera. Available online: http://www.youtube.com/watch?v=GdepIXZTJsw, 2011.
42. Alissandrakis, A.; Nehaniv, C.L.; Dautenhahn, K. Action, state and effect metrics for robot imitation. The 15th IEEE International Symposium on Robot and Human Interactive Communication, 2006, pp. 232–237.
43. Zhang, Z.; Beck, A.; Magnenat-Thalmann, N. Human-like behavior generation based on head-arms model for robot tracking external targets and body parts. *IEEE transactions on cybernetics* **2015**, *45*, 1390–1400.
44. Harada, T.; Taoka, S.; Mori, T.; Sato, T. Quantitative evaluation method for pose and motion similarity based on human perception. 4th IEEE/RAS International Conference on Humanoid Robots, 2004, Vol. 1, pp. 494–512.
45. Robotics, A. Nao Joints of Upper and Lower Limbs. Available online: http://doc.aldebaran.com/2-1/family/robots/joints_robot.html, 2018.
46. DARwin-OP. Available online: http://support.robotis.com/en/product/darwin-op.htm, 2018.
47. Honda. ASIMO: The World's Most Advanced Humanoid Robot. Retrieved from http://asimo.honda.com/asimo-specs/, 2018.
48. Lee Rodgers, J.; Nicewander, W.A. Thirteen ways to look at the correlation coefficient. *The American Statistician* **1988**, *42*, 59–66.
49. Billard, A.; Calinon, S.; Dillmann, R.; Schaal, S. Robot programming by demonstration. In *Springer handbook of robotics*; Springer, 2008; pp. 1371–1394.
50. Calinon, S.; Guenter, F.; Billard, A. On learning, representing, and generalizing a task in a humanoid robot. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* **2007**, *37*, 286–298.
51. Chiu, C.Y.; Chao, S.P.; Wu, M.Y.; Yang, S.N.; Lin, H.C. Content-based retrieval for human motion data. *Journal of Visual Communication and Image Representation* **2004**, *15*, 446–466.